

The Pennsylvania State University
The Graduate School
College of Engineering

SEMANTICS AND AESTHETICS INFERENCE FOR IMAGE
SEARCH: STATISTICAL LEARNING APPROACHES

A Dissertation in
Computer Science and Engineering
by
Ritendra Datta

© 2009 Ritendra Datta

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2009

The dissertation of Ritendra Datta was reviewed and approved* by the following:

James Z. Wang
Associate Professor of Information Sciences and Technology
Dissertation Co-Adviser, Co-Chair of Committee

Jia Li
Associate Professor of Statistics
Dissertation Co-Adviser, Co-Chair of Committee

Robert Collins
Associate Professor of Computer Science and Engineering

C. Lee Giles
David Reese Professor of Information Sciences and Technology

David Miller
Professor of Electrical Engineering

Bhuvan Ugaonkar
Assistant Professor of Computer Science and Engineering

Raj Acharya
Professor of Computer Science and Engineering
Head of the Department of Computer Science and Engineering

*Signatures are on file in the Graduate School.

Abstract

The automatic inference of image semantics is an important but highly challenging research problem whose solutions can greatly benefit content-based image search and automatic image annotation. In this thesis, I present algorithms and statistical models for inferring image semantics and aesthetics from visual content, specifically aimed at improving real-world image search. First, a novel approach to automatic image tagging is presented which furthers the state-of-the-art in both speed and accuracy. The direct use of automatically generated tags in real-world image search is then explored, and its efficacy demonstrated experimentally. An assumption which makes most annotation models misrepresent reality is that the state of the world is static, whereas it is fundamentally dynamic. I explore learning algorithms for adapting automatic tagging to different scenario changes. Specifically, a meta-learning model is proposed which can augment a black-box annotation model to help provide adaptability for personalization, time evolution, and contextual changes. Instead of retraining expensive annotation models, adaptability is achieved through efficient incremental learning of only the meta-learning component. Large scale experiments convincingly support this approach. In image search, when semantics alone yields many matches, one way to rank images further is to look beyond semantics and consider visual quality. I explore the topic of data-driven inference of aesthetic quality of images. Owing to minimal prior art, the topic is first explored in detail. Then, methods for extracting a number of high-level visual features, presumed to have correlation with aesthetics, are presented. Through feature selection and machine learning, an aesthetics inference model is trained and found to perform moderately on real-world data. The aesthetics-correlated visual features are then used in the problem of selecting and eliminating images at the high and low extremes of the aesthetics scale respectively, using a novel statistical model. Experimentally, this approach is found to work well in visual quality based filtering. Finally, I explore the use of image search techniques for designing a novel image-based CAPTCHA, a Web security test aimed at distinguishing humans from machines. Assuming image search metrics to be potential attack tools, they are used in the loop to design attack-resistant CAPTCHAs.

Table of Contents

List of Figures	vii
List of Tables	xv
Acknowledgments	xvi
Chapter 1	
Introduction	1
1.1 Semantic and Aesthetic Gaps	2
1.2 Contributions and their Significance	4
1.3 Organization of this Dissertation	9
Chapter 2	
Image Search, Annotation, Aesthetics: State of the Art	11
2.1 Introduction	11
2.2 Image Search Techniques: Addressing the Core Problem	22
2.3 Offshoots: Problems of the New Age	56
2.4 Summary	66
Chapter 3	
Bridging the Semantic Gap:	
Improving Image Search via Automatic Annotation	68
3.1 Model-based Categorization	71
3.2 Annotation and Retrieval	82
3.3 Experimental Validation	84
3.4 Summary	93

Chapter 4	
Bridging Dynamic Semantic Gap:	
Adapting Automatic Image Tagging via Meta-learning	94
4.1 PLMFIT: Principled Meta-learning Framework for Image Tagging . . .	101
4.2 Tagging Improvements with PLMFIT	114
4.3 Experimental Results	125
4.4 Issues and Limitations	135
4.5 Summary	137
Chapter 5	
Beyond Semantics:	
Basics, Inference, and Applications of Aesthetics	138
5.1 Questions of Interest	139
5.2 Technical Solution Approaches	142
5.3 Public Datasets for Empirical Modeling	144
5.4 Visual Features for Photographic Aesthetics	148
5.5 Feature Extraction	153
5.6 Feature Selection, Classification, Regression	162
5.7 Empirical Evaluation of Inference	164
5.8 Image Filtering via Aesthetics Inference	167
5.9 Regression and Classification Models	168
5.10 Selection and Elimination Algorithms	172
5.11 Empirical Evaluation of Search Refinement	173
5.12 Summary	177
Chapter 6	
Exploiting the Semantic Gap:	
Designing CAPTCHAs Using Image Search Metrics	179
6.1 Image Recognizability Under Distortion	183
6.2 Experimental System: IMAGINATION	190
6.3 Comparison with Existing CAPTCHAs	195
6.4 Experimental Results	199
6.5 Summary	208
Chapter 7	
Conclusions and Future Research Directions	210
7.1 Future Research Directions	211
7.2 The Future of Image Search	212

Appendix A	
Trends and Impact of Image Retrieval Research	214
A.1 Publication Trends	214
A.2 Scientific Impact on Other Communities	217
Appendix B	
Orthogonal Partition Generation	220
Bibliography	223

List of Figures

1.1	A high level overview, of contributions made in this dissertation, is shown here. At its core reside the concepts of semantic and aesthetic gaps. A significant contribution of this work involves attempts to ‘bridge’ these gaps in the form of inferring semantics and aesthetics of natural images. This work also addresses the fact that semantic gap is essentially dynamic in the real world. The semantic gap can be exploited for solving an orthogonal problem in Web security, which is another contribution of this dissertation.	3
2.1	A study of post-1995 publications in CBIR. <i>Top</i> : Normalized trends in publications containing phrases “image retrieval” and “support vector” in them. <i>Bottom</i> : Publisher wise break-up of publication count on papers containing “image retrieval” in them.	15
2.2	Our view of the many facets of image retrieval as a field of research.	16
2.3	Real-world use of content-based image retrieval using color, texture, and shape matching. <i>Top</i> : http://airliners.net , is a photo-sharing community with more than a million airplane-related pictures. <i>Bottom</i> : http://riya.com is a collection of several million pictures.	20
2.4	Real-world use of automatic image annotation, http://alipr.com . The screenshot shows a random set of uploaded pictures and the annotations given by ALIPR (shown in blue and gray) and by users (shown in green).	21
2.5	An overview on image signature formulation.	24
2.6	Different types of image similarity measures, their mathematical formulations and techniques for computing them.	34
2.7	Paradigms of clustering methods and their scopes of applications. .	46
3.1	Four common scenarios for real-world image retrieval.	69

3.2	The idea behind the S-C model, shown here on a <i>toy</i> image. We denote the perimeters of each segment by Θ and the border lengths between pairs of segments by Δ . Intuitively, Δ/Θ ratios for the <i>orange, light-blue</i> (sun and sky) and <i>white, light-blue</i> (clouds and sky) pairs equals 1 since sun and cloud perimeters coincide with their borders shared with sky. In general, the ratio has low value when segments are barely touching, and near 1 when a segment is completely contained within another segment.	74
3.3	Steps toward generating the structure-composition model. On the left, we have three training pictures from the ‘bus’ category, their segmented forms, and a matrix representation of their segment adjacency counts. On the right, the corresponding matrix representations over all three training pictures are shown. Finally, these matrices are combine to produce the structure-composition model, shown here schematically as a matrix of Beta parameters and counts.	76
3.4	Sample categories and corresponding structure-composition model representations. <i>Top</i> : Sample training pictures. <i>Middle</i> : Matrices of segment adjacency counts. <i>Bottom</i> : Matrices of mean Δ/Θ ratios. Brightness levels represent relative magnitude of values. . . .	78
3.5	Algorithm for computing S-C features.	80
3.6	Categorization accuracies for the 10-class experiment are shown. Performance of our combined S-C+ C-T model is shown with varying number of mixture components in the C-T model. Previously reported best results shown for comparison.	85
3.7	Sample automatic tagging results on some Yahoo! Flickr pictures taken in Amsterdam, show along with the manual tags.	86
3.8	Precision (left) and recall (right) achieved by re-annotation, with varying noise levels in original tags. Note that the linear correlation of the baseline case to e is intrinsic to the noise simulation.	88
3.9	Precision (left) and recall (right) achieved by re-annotation, varying parameter Z , shown for 5 noise levels.	88
3.10	Precision (left) and recall (right) under scenario 1, compared to baseline.	90
3.11	Precision (left) and recall (right) under scenario 2, compared to baseline.	90
3.12	Precision (left) and recall (right) under scenario 3, compared to baseline.	91
3.13	Retrieval precision for scenario 4 at three noise levels.	92
4.1	High-level overview of our PLMFIT meta-learning framework. . . .	102

4.2	Estimates of $Pr(A_{w_j} G_{w_j} = 1)$ as obtained empirically with images from the Alipr dataset (see Sec. 4.3) for 40 most frequently occurring tags (decreasing frequency from left to right). As can be seen, the black-box is much more precise in predicting tags such as ‘face’ or ‘plant’, compared to ‘texture’ or ‘ice’.	105
4.3	Visualization of ratio $\frac{Pr(G_{w_i}=1 A_{w_j}=1, G_{w_j}=1)}{Pr(G_{w_i}=1 A_{w_j}=0, G_{w_j}=1)}$ as obtained empirically with images from the Alipr dataset (see Sec. 4.3) for 30 most frequently occurring tags. Two interesting cases, that highlight the importance of these terms to meta-learning effectiveness, are marked. For example, the value at location (A) can be read as the ratio of probabilities of ‘water’ being guessed for an image by the black-box given that ‘sky’ is also guessed, correctly versus incorrectly.	107
4.4	Network depicting WordNet-based relations among the 60 most frequently occurring tags in the Alipr dataset (Sec. 4.3). An edge between a pair of words indicates that the relatedness measure LCH [153] exceeds 1.7, roughly the mid-point of its [0.37, 3.58] range of values.	111
4.5	Empirical evidence, based on Alipr dataset (Sec. 4.3) that WordNet can help with inductive transfer. Networking depicting proximity of pre-smoothed estimates of $Pr(G_{w_i}=1 A_{w_j}=a, G_{w_j}=1)$ for pairs of tags among the top 60 most frequently occurring ones. Specifically, there is an edge between a pair of tags w_{j1} and w_{j2} if $\frac{1}{ V } \sum_{i=1}^{ V } Pr(G_{w_i}=1 A_{w_{j1}}=a, G_{w_{j1}}=1) - Pr(G_{w_i}=1 A_{w_{j2}}=a, G_{w_{j2}}=1) $ for $a = 0$ and 1 are both below 0.0025 (chosen to generate a less-cluttered but interesting graph). Compared to Fig. 4.4, while we see many overlaps, there are quite a few differences as well, by virtue of what the relations stand for.	112
4.6	Empirical differences between estimated $Pr(G_{w_i}=1 A_{w_j}=a, G_{w_j}=1)$ when using semantically related words as against unrelated ones, are shown. For each of the 30 most frequent tags w_i in Alipr dataset, we compute $y(w_i)$ (defined in Eq. 4.11 in the text). What we see is that out of 30 tags, 25 and 27 tags for $a = 0$ and $a = 1$ respectively have better estimates of the probability terms when substituted with semantically related terms, as against unrelated ones. This indicates that smoothing with relatedness weights is an attractive strategy.	113
4.7	The 48-dimensional <i>LUV</i> features extracted in PLMFIT.	114

4.8	Estimated values of $\hat{\mu}_{j,d,a}$, model parameters for $Pr(h_1, \dots, h_{48} \mid A_{w_j}=a)$, for the 48-dimensional global image features for two tags with observed differences, are shown. As in the case of ‘space’ and ‘fruit’, if differences are significant for $A_{w_j} = 0$ and 1, then this ratio contributes to the inference. An intuition behind the difference in case of ‘fruit’, for example, is that close-up shots of fruits tend to be brighter and more colorful than is typical.	115
4.9	Time-ordered histograms of occurrence of the top 10 most frequent tags in the Alipr dataset (consisting of 20,000 images), computed over 2,000 image overlapping windows (except last one) with window starting points at 1,000 image intervals. Notice how tag popularity fluctuate over time, e.g., after a point, ‘wildlife’ diminishes in frequency while ‘animal’ gains prominence.	116
4.10	Tagging adaptation over time using a black-box augmented with PLMFIT.	117
4.11	Overview of persistent/transient memory models for tagging adaptation over time.	118
4.12	Distribution over the 50 most frequent tags in the Flickr dataset (see Sec. 4.3) for 18 randomly sampled users. Note how the distribution greatly varies, which reinforces our belief that personalization can give automatic image tagging a significant performance boost. . . .	121
4.13	Motivating the case for personalization with Flickr data: Graph (1)-(3) depict the fraction of all tags covered by the most frequent 5, 10, and 20 tags respectively for each of 100 randomly chosen users (sorted by % covered). The dashed line shows the same for all users pooled together, providing evidence that the tags space is localized for most users. Graph (4) shows the distribution of overlap between tags of 100, 000 image pairs each sampled randomly from (a) within same users, and (b) across users, normalized by the minimum number tags for each pair. While more than 90% of the across-user cases have no tag overlaps, almost 50% of the same-user pairs have some overlap, while 8%+ have maximally identical tags.	122
4.14	Performance (precision and F-score) of adaptation over time for Alipr trace #1.	128
4.15	Performance (precision and F-score) of adaptation over time for Alipr trace #2.	129
4.16	Comparison of precision and F-score for the two memory models of incremental learning, <i>persistent</i> and <i>transient</i>	130
4.17	Comparing F-score and computation time with varying L_{inter}	131

4.18	Sample annotation results found to improve over time with PLM-FIT adaptation.	132
4.19	Graph showing precision, recall, and F-score for a number of settings, including Alipr's tagging of the Flickr images (baseline), PLMFIT adaptation without personalization (seed), and personalization with different numbers of tags predicted. The case of 'User Average' uses the average number r of tags for a given user's training data and predicts top r tags for the test cases.	133
4.20	Graph showing variation of precision and recall with a varying proportion of seed data to user-specific samples, used for training the PLMFIT for personalization.	134
5.1	Three aesthetics inferencing problems of significance.	139
5.2	Distributions of the average photo ratings received.	145
5.3	Distributions of number of photo ratings received per image.	146
5.4	Correlation plot of (average score, number of ratings) pairs.	146
5.5	Distribution of the level of consensus among ratings.	147
5.6	Distribution of emotion votes given to images (Alipr).	147
5.7	Correlation between the aesthetics and originality ratings for 3581 photographs obtained from Photo.net.	149
5.8	Aesthetics scores can be significantly influenced by the semantics. Loneliness is depicted using a person in this frame, though the area occupied by the person is very small. Average aesthetics score for these images are 6.0 out of 7 (left) and 6.61 out of 7 (right).	151
5.9	The proposed <i>colorfulness</i> measure, f_2 . The two photographs on the <i>left</i> have high values while the two on the <i>right</i> have low values.	155
5.10	(a) The <i>rule of thirds</i> in photography: Imaginary lines cut the image horizontally and vertically each into three parts. Intersection points are chosen to place important parts of the composition instead of the center. (b)-(d) Daubechies wavelet transform. <i>Left</i> : Original image. <i>Middle</i> : Three-level transform, levels separated by borders. <i>Right</i> : Arrangement of three bands LH, HL and HH of the coefficients.	157
5.11	The HSV Color Space.	159
5.12	Aesthetics ratings are often higher for images with low depth of field. (a) 6.37 out of 7 (b) 6.25 out of 7	160
5.13	Demonstrating the <i>shape convexity</i> feature. <i>Left</i> : Original photograph. <i>Middle</i> : Three largest non-background segments shown in original color. <i>Right</i> : Exclusive regions of the <i>convex hull</i> generated for each segment are shown in white. The proportion of white regions determine the convexity value.	162

5.14	<i>Left:</i> Variation of 5 – CV SVM accuracy with the minimum number of unique ratings per picture. <i>Right:</i> Variation of 5 – CV SVM accuracy with inter-class gap δ	164
5.15	The CART tree obtained on the 56 visual features (partial view). .	166
5.16	Example images from Photo.net where the consensus aesthetics score ≥ 6 (above), and ≤ 4 (below), on 1 – 7.	168
5.17	Distributions of no. of ratings (left) and scores (right) in Photo.net dataset.	174
5.18	Precision in selecting high-quality images, shown here for three selection set sizes, $T = 10, 20$, and 30 . <i>Bottom-right:</i> Impact of using weighted model estimation vs. their unweighted counterparts, with <i>HIGH</i> fixed and T varying.	175
5.19	A sample instance of $T = 10$ images selected by our approach, for <i>HIGH</i> = 5.5. The actual consensus scores are shown in red, indicating an 80% precision in this case.	176
5.20	<i>Above:</i> Precision in eliminating low-quality images, shown here for two set sizes, namely $T = 25$ and 50 . <i>Below:</i> The corresponding errors, made by eliminating high-quality images in the process. . . .	177
6.1	Sample CAPTCHAs proposed or in real-world use. (a)-(b) Text-based CAPTCHAs in public use. (c) Image-based CAPTCHA proposed by CMU’s Captcha Project. User is asked to choose an appropriate label from a list. (d) Asirra [76] presents pictures of cats and dogs and asks users to select all the cats.	180
6.2	Architecture of the IMAGINATION system. The circled ‘R’ components represent randomizations.	190
6.3	Screenshot of the Click step of authentication in the IMAGINATION system. The tiled image is randomly partitioned orthogonally and dithered using different color sets, to make it harder for automated identification of the image boundaries. The user must click near the center of one of the images to get past this step. . . .	193
6.4	Two screenshots of the Annotate step in the IMAGINATION system, where a distorted image is presented, and the user must select an appropriate label from a list of choices.	193
6.5	Plot of distribution of distances of user clicks from the nearest image centers within the composite images. This distribution (26, 152 points), plotted in base-10 log scale, shows that a majority of the clicks are within a short distance of a center.	200

6.6	Variation of success rates by human (above) and automated randomly sampled clicks (below) with varying tolerance radius r . This graph helps to choose r given a desired trade-off between human ease of use and insulation from attacks.	201
6.7	Variation of average machine recognizability with change in luminance scaling factor. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$	203
6.8	Variation of average machine recognizability with change in density of noisy lines added, represented in pixels specifying the gap between consecutive lines. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$	204
6.9	Variation of average machine recognizability with change in quantization level, specified in terms of the number of color clusters generated and (centroids) used for mapping. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$, and we show the cut/rescale case here.	206
6.10	Variation of average machine recognizability with change in dithering level, specified in terms of the number of colors available for dithering each partition. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$, and we show the cut/rescale case here.	207
6.11	Overall variation of human recognizability with dithering parameter DITHERPAR and noise density parameter DENSEPAR, taken across all four composite distortion methods.	208
6.12	Overall variation of human recognizability with the image concept, taken across all four composite distortions and their parameter values; 15 most frequently sampled concepts are shown here.	208

A.1	Conference-wise and journal-wise publication statistics on topics closely related to image retrieval, year 2000 onwards. <i>Top</i> : Publication counts. <i>Bottom</i> : Total citations.	215
A.2	Publication statistics on sub-topics of image retrieval, 2000 onwards. <i>Top</i> : Publication Counts. <i>Bottom</i> : Total citations. <i>Abbreviations</i> : <i>Feature</i> - Feature Extraction, <i>R.F.</i> - Relevance Feedback, <i>Similar</i> - Image similarity measures, <i>Region</i> - Region based approaches, <i>App.</i> - Applications, <i>Prob.</i> - Probabilistic approaches, <i>Speed</i> - Speed and other performance enhancements.	216
A.3	Normalized trends in publications containing ‘image retrieval’ and corresponding phrases, as indexed by Google Scholar. Counts are normalized by the number of papers having ‘image retrieval’ for the particular year.	218
A.4	[Acronyms: MM := Multimedia, IR := Information Retrieval, DL := Digital Libraries/ World Wide Web, HCI := Human-Computer Interaction, LN := Language Processing, AI := Artificial Intelligence, and CV := Computer Vision]. Directed graphs representing inter-field impact induced by CBIR-related publications. An edge $a \rightarrow b$ implies publications at venue/journal concerning field b , having content concerning field a . We show oppositely directed edges between pairs of nodes, wherever significant, in the left and right graphs. <i>Top</i> : Edge thicknesses represent (relative) publication count . <i>Bottom</i> : Edge thicknesses represent (relative) citations as reported by Google Scholar.	219
B.1	Steps to orthogonal partition generation, to create 8 rectangular sub-regions for image tiling.	220

List of Tables

2.1	Popular distances measures used for image similarity computation. .	41
2.2	Comparison of three learning techniques in context of image retrieval.	44
2.3	A qualitative requirement analysis of various CBIR offshoots.	65
4.1	Contextual adaptation performance on 10,000 Corel images	126
4.2	Contextual adaptation performance on 16,000 Alipr images	127
4.3	Personalization performance on 162,650 Flickr images (300 users)	132
5.1	Datasets available for emotion/aesthetics learning.	144
6.1	Some features and distortions that affect their extraction	187
6.2	Qualitative comparison of our system with other CAPTCHAs . . .	197
6.3	Four Distortions that are part of the IMAGINATION System . . .	205

Acknowledgments

First and foremost, I would like to thank my advisors, Professors James Wang and Jia Li, for their continued guidance and encouragement, to help me become forward-thinking and self-sufficient. Owing to their training and support, my introduction to the world of research was much accelerated. The guidance from them, received through courses, meetings, and regular discussion sessions, has been most valuable to me. The mentoring I have received from them have spanned well beyond academic research. Thanks to them, I believe I am professionally groomed better today than I was in 2004, and I now have a better understanding of my role in the grand scheme of things.

I would also like to thank Professors Robert Collins, Lee Giles, David Miller, and Bhuvan Urgaonkar for agreeing to serve on my dissertation committee, taking time out of their busy schedules, to provide valuable feedback. My interactions with them through courses, research, or general discussions have all been helpful in shaping my thesis, my attitude toward research, and my goals for the future.

In the course of my doctoral study, I have had the opportunity to collaborate with brilliant fellow students Amitayu Das, Marc Friedenberg, Weina Ge, Dhiraj Joshi, Razvan Orendovici, Ashish Parulekar, Neela Sawant, Walter Weiss, and Ziming Zhuang, among others. My summer research collaborations with Jianying Hu and Bonnie Ray (IBM Research), Marshall Bern (Xerox PARC), and Charles Schafer (Google) were great learning experiences which helped give new perspectives to my core dissertation work. I would like to thank them for their contributions, mentoring, valuable discussions, and continued collaboration.

I am thankful to the US National Science Foundation for supporting my dissertation work, under grant numbers 0347148 and 0219272.

Last, but in no way least, I would like to thank my parents, sister Bidisha, brother-in-law Amit, aunt Bu, and Manaswini, for their continued support and faith in my abilities. Their unquestioned and uninterrupted encouragement over the years has been a constant source of sustenance. I am forever indebted to my late grandparents for some of the most memorable moments of my life.

Dedication

To Ma and Baba,
Manikuntala and Rabindra Datta,
who have contributed to my work like nobody else.

Introduction

The word ‘semantics’, in the context of this dissertation, refers to the linguistic interpretation of media such as images and video. The human mind, through years of real-world experience and interactions, usually finds it natural to interpret images or summarize video clips. However, the aggregate size of the image and video collections of the world has been growing at a phenomenal rate [155, 174], making it infeasible to manually interpret every entity. For example, it is reported in [174] that the annual production of images is ~ 80 billion, and home videos is ~ 1.4 billion. Furthermore, Web portals such as Flickr, Facebook, and YouTube have made it easier than ever before to upload personal collections to public repositories, spearheading an explosion of user-generated content.

The scale of these media collections poses new problems for information retrieval. A key problem is to be able to organize and search them based on semantics. This is relatively easy in the presence of reliable, manually generated semantics data. The present day image search engines such as those owned by Yahoo! and Google, rely mainly on the text in the vicinity of the images in order to infer semantics. Much of these images come without explicit semantic tags, and those inferred from surrounding text are often unreliable. Therein lies the need to automatically infer semantics from visual content, to either facilitate semantics-based search directly, or to generate meaningful tags. In order to be useful in the real-world, it is preferable for automatic inference to be scalable and accurate. Additional desiderata of future information retrieval systems include the ability to recognize and leverage information deeper than semantics, such as emotional

response. For example, given the ever-expanding size of image repositories, many images share very similar semantics. A retrieval system for images can potentially enhance the user experience by re-ranking images based on inferred aesthetic value.

1.1 Semantic and Aesthetic Gaps

The technical challenges involved with semantics recognition have been formalized lucidly in relevant literature. An important formalization is the concept of *semantic gap* proposed by Smeulders et al. [242], which is defined as follows:

The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.

In simpler terms, it usually implies the inability of current technology to completely understand the semantics of multimedia objects in general, and images in particular. Similarly, the technical challenge involved with inferring aesthetic quality from visual content of images can be formalized by the concept of *aesthetic gap* which is introduced in this dissertation, and defined analogously as follows:

The aesthetic gap is the lack of coincidence between the information that one can extract from low-level visual data (i.e., pixels in digital images) and the interpretation of emotions that the visual data may arouse in a particular user in a given situation.

Again, in simpler terms, the aesthetic gap refers to the inability of current technology to infer the aesthetic quality of an image as perceived by an individual or collectively by a group of individuals. A high level conceptual view of these gaps is presented in Fig. 1.1.

The ability to overcome these technical challenges, partially or completely, is often referred to as the ‘bridging’ of these gaps. There have been many attempts in the past [58, 242] to bridge the semantic gap, but little to no formal attempt at bridging the aesthetic gap has been made. The latter is arguably a harder technical challenge, starting with the lack of a concrete definition of what constitutes *ground-truth* aesthetic quality of images. A major focus of this dissertation is on ways to bridge the semantic and aesthetic gaps.

If we revisited the definition of semantic gap, we would realize that in real-world contexts, it is in fact *dynamic* in nature. The *information that one can extract from the visual data* for a one-time trained image recognition model does not change, but on the other hand, the *interpretation that the same data have for a user in a given situation* changes across users as well as situations. Hence in real-world implementations, the algorithms really should attempt to bridge the *dynamic semantic gap*, a concept not explored in the existing literature. While similar arguments apply to the aesthetic gap as well, this dissertation focuses on the dynamic nature only of the semantic gap.

1.2 Contributions and their Significance

The contributions made in this dissertation can be summarized at a high level in terms of ‘bridging of gaps’ (see Fig. 1.1), or they can be broken down into a detailed set of accomplishments. I begin with a high level perspective and then go on to spell out the details. Very broadly, my contributions can be summarized by the following points:

1. Pushing the frontier in bridging the semantic gap;
2. Formalizing the concept of aesthetic gap and proposing to bridge it; and
3. Exploiting the semantic gap to solve a problem in information security.

The complete elimination of the semantic or aesthetic gaps through technological advancements is very ambitious, and very unlikely to happen in the near future. Instead, my claim in this work is to have narrowed these gaps, which translates to performance improvements in relevant real-world applications.

At a more detailed level, in this dissertation I present algorithms and statistical models for inferring image semantics and aesthetics from visual content, specifically aimed at improving image search and tagging. Content-based image search and automatic tagging are real-world applications, and the bridging of the semantic and aesthetic gaps are merely ways to improve the user experience in these applications. I now present details of the specific contributions made in this dissertation.

A note on terminology:

Throughout this dissertation, I use certain sets of phrases or terms interchangeably. This is partly because there is a lack of consistency in terminology within the research community as well, since they refer to the same concepts. In particular, I consider the sets of phrases {‘image search’, ‘image retrieval’, ‘content-based image retrieval’, ‘CBIR’}, then {‘image annotation’, ‘image tagging’}, and finally {‘aesthetic value’, ‘visual quality’}, to be equivalence sets in terms of semantics.

1.2.1 A Thorough Study of State-of-the-art

To start with, so as to put the contributions of this work into perspective, I present a thorough survey on relevant research topics, in particular the recent research efforts at bridging the semantic gap for image categorization, retrieval, automatic image tagging, and related new-age problems such as aesthetics inference. We recap progress made in the previous decade, cover progress being made in the current decade (in great technical detail), make conjectures about emerging directions, and discuss the spawning of new fields of research as a result of these efforts. The recent publication trends in this area of research, and their impact on related fields of study, are presented as well.

Significance:

- The thorough survey of the state-of-the-art in image retrieval and automatic annotation presented here is more comprehensive and up-to-date than other existing surveys on the topic. It not only helped me decide on topics worthy of exploration and to make novel contributions, but it will also likely serve as an importance reference for the research community.

1.2.2 Improved Bridging of the Semantic Gap

The main contributions of this dissertation start with an effort to improve the state-of-the-art in bridging the semantic gap for the purpose of image search. In particular, an approach to automatic tagging of images is presented which furthers the state-of-the-art in both speed and accuracy. This involves a novel structure-composition model which helps perform image categorization better than previously proposed algorithms, which is a key step in our automatic tagging approach. The direct use of automatically generated tags in image search under various real-world scenarios is then explored. Referred to as the ‘bridging’ of the annotation-retrieval gap, its efficacy is shown through extensive experiments.

Significance:

- The algorithm for automatic tagging presented here is an improvement in training time, inference time, and in the accuracy of tagging, as compared to current approaches. It brings automatic image tagging a step closer to real-world implementation, where efficiency and performance are both critical.
- The proposed structure-composition model appears to model challenging image categories well, which leads to improved categorization performance over existing approaches. This model may therefore find application in other image recognition problems as well.
- The use of automatic annotation directly for image search has not been explored prior to this work, an approach which shows considerable promise and produces some surprising non-intuitive results. This gives direct evidence that content recognition can help provide access to unlabeled images without changing query modality, sticking to the keyword search paradigm.

1.2.3 Bridging the Dynamic Semantic Gap

One issue with using automatic tagging for real-world image search is that most existing models assume that ground-truth image tags come from a fixed vocabulary, are absolute over time, and are universally acceptable by various people, which make the models misrepresent the dynamic nature of the real world. The association between images and their semantic tags changes with context, over time, and across people, which makes the semantic gap itself dynamic. Thus, algorithms for bridging the semantic gap should adapt to changes, but this idea has not been pursued extensively in the past. I explore learning algorithms for adapting automatic image annotation to such dynamism. A meta-learning model called PLMFIT is proposed, which can augment a black-box annotation model to help provide the requisite adaptability to handle contextual changes, time evolution, and personalization. Instead of re-training expensive image tagging models, efficient adaptability is achieved through incremental learning of only the lightweight meta-learning component. Strongly positive empirical results suggest that meta-learning is very promising in bringing adaptability to image tagging.

Significance:

- The idea of using a meta-learning layer to improve performance of an annotation system is novel and empirically found to produce very significant performance improvements. By itself, even without the dynamism in tagging environment, *inductive transfer* helps PLMFIT in pushing the state-of-the-art in automatic tagging performance.
- Our model is tested successfully on real-world data in which image tagging trends are found to change over time, and users are found to vary in their tagging preferences. This makes it the first annotation model to perform effective adaptation over time and personalization on real users.
- The efficiency and performance achieved through the meta-learning layer makes it realistic to implement computationally intensive annotation systems (e.g., Alipr [166]) within public photo-sharing environments like Flickr.

1.2.4 Formalizing and Bridging the Aesthetic Gap

In image search, when semantics alone yields many relevant matches, ordering them by visual quality can be beneficial. I explore the topic of data-driven inference of visual quality or ‘aesthetic value’ of images. Given the highly subjective nature of this problem, I focus specifically on building data-driven models for aesthetics inference, posing it as a machine learning problem. Owing to minimal prior art, the topic is first explored in great detail, presenting definitions, scope, problems, of interest, and datasets available for training. Then, methods for extracting a number of high-level visual features, presumed to have correlation with aesthetics, are presented. Through feature selection and machine learning, an aesthetics inference model is trained and found to perform moderately on real-world data. The aesthetics-correlated visual features are then used in the problem of selecting and eliminating images at the high and low extremes of the aesthetics scale respectively, using a novel statistical model. Experimentally, this approach is found to work well in tasks such as visual quality filtering of image search results.

Significance:

- The section on fundamentals presents concrete definitions, problems of interests, and pointers to datasets which can be used for empirical validation. This will serve as a key reference for this emerging area of study.
- Ours is among the first formal studies on computational inference of aesthetics. Despite aesthetics being considered very challenging to model, our experiments yield moderate results, giving hope for further advances.
- Even though the direct inference of aesthetic value of images produces only moderate results, when applied to some realistic related problems, very encouraging results are generated. This implies that the inference model need not be perfect in order to be applicable to the real world.

1.2.5 Exploiting the Semantic Gap for Enhanced Security

As mention before, based on past literature and current progress, it is safe to assume that the semantic gap will not be completely eliminated in the near future. This fact can be exploited for the purpose of enhancing system security. Specifically, image recognition problems can be used as CAPTCHAs, tests to distinguish humans from machines to alleviate various Web security problems. I explore the use of image search techniques for designing a novel image-based CAPTCHA. Such a test helps prevent automatic brute-force network attacks by forcing human intervention. Image recognition based CAPTCHAs have emerged as attractive new alternatives to the (presently dominant) text-based systems. While these are perceived to be harder to defeat than text recognition tests, there is still the risk of defeat by image analysis techniques. Assuming image search metrics to be potential attack tools, I use them in the loop to design an attack-resistant CAPTCHA system. In other words, the goal is to design image recognition CAPTCHAs which face low risk of attack from the existing techniques which attempt to bridge the semantic gap.

Significance:

- An understanding of (a) the semantic gap for images and (b) approaches to bridging it, positions us well strategically for designing the IMAGINATION system, an attack resistant CAPTCHA system. The basic design principle presented here can be used to extend it further as and when new innovation in bridging the semantic gap comes along. A public demonstration of this system is currently available at <http://alipr.com/captcha>.
- The study also helps to reveal (a) robustness of current image search metrics to certain artificial distortions, and (b) limitations in dealing with certain other kinds of distortions. This may potentially help design improved, more robust image search metrics in the future.

1.3 Organization of this Dissertation

To start with, so as to put the contributions into perspective, I present a thorough survey on relevant research topics, in Chapter 2. The topics covered include image search, automatic image tagging, image aesthetics inference, and related new-age applications. The main contributions of this dissertation start with Chapter 3. In this chapter, a novel approach for automatic tagging of images is presented which furthers the state-of-the-art in both speed and accuracy. The direct use of automatically generated tags in image search under various real-world scenarios is then explored. In Chapter 4, I explore learning algorithms for adapting automatic image annotation to different scenario changes. A meta-learning model called PLMFIT is proposed, which can augment a black-box annotation model to help provide the requisite adaptability to handle contextual changes, time evolution, and personalization. In image search, when semantics alone yields many relevant matches, ordering them by visual quality can be beneficial. I explore the topic of data-driven inference of visual quality or ‘aesthetic value’ of images in Chapter 5, covering the fundamentals, an approach to inference, and an application of the inference model. Finally, in Chapter 6, I explore the use of image search techniques for designing a novel image-based CAPTCHA, a Web security test aimed at distinguishing humans and machines. I conclude in Chapter 7 with

summarizing remarks, a discussion on directions that the presented research topics can take hereon, and a general discussion on the future of image search.

In addition to this, Appendix A presents publication trends and impact of image search research, in connection with the survey in Chapter 2. Appendix B presents details and proof on the orthogonal partition generator component of the IMAGINATION system described in Chapter 6.

Chapter 2

Image Search, Annotation, Aesthetics: State of the Art

We have witnessed great interest and a wealth of promise in content-based image retrieval as an emerging technology. While the last decade laid foundation to such promise, it also paved the way for a large number of new techniques and systems, got many new people involved, and triggered stronger association of weakly related fields. In this chapter, we survey key theoretical and empirical contributions in the current decade related to image retrieval and automatic image annotation, and discuss the spawning of related sub-fields in the process, such as image aesthetics inference and image-based security systems. We also discuss significant challenges involved in the adaptation of existing image retrieval techniques to build systems that can be useful in the real-world. In retrospect of what has been achieved so far, we also conjecture what the future may hold for image retrieval research.

2.1 Introduction

What Niels Henrik David Bohr exactly meant when he said “Never express yourself more clearly than you are able to think” is anybody’s guess. In light of the current discussion, one thought that this well-known quote evokes is that of subtle irony; there are times and situations when we imagine what we desire, but are unable to express this desire in precise wording. Take, for instance, a desire to find the perfect portrait from a collection. Any attempt to express what makes a portrait

‘perfect’ may end up undervaluing the beauty of imagination. In some sense, it may be easier to find such a picture by looking through the collection and making unconscious ‘matches’ with the one drawn by imagination, than to use textual descriptions that fail to capture the very essence of perfection. One way to appreciate the importance of visual interpretation of picture content for indexing and retrieval is this.

Our motivation to organize things is inherent. Over many years we learned that this is a key to progress without the loss of what we already possess. For centuries, text in different languages has been set to order for efficient retrieval, be it manually in the ancient *Bibliothèque*, or automatically as in the modern digital libraries. But when it comes to organizing pictures, man has traditionally outperformed machines for most tasks. One reason which causes this distinction is that text is man’s creation, while typical images are a mere replica of what man has seen since birth, concrete descriptions of which are relatively elusive. Add to this the theory that the human vision system has evolved genetically over many centuries. Naturally, the interpretation of what we see is hard to characterize, and even harder to teach a machine. Yet, over the past decade, ambitious attempts have been made to make computers learn to understand, index and annotate pictures representing a wide range of concepts, with much progress.

Content-based image retrieval (CBIR), as we see it today, is any technology that in principle helps organize digital picture archives by their visual content. By this definition, anything ranging from an image similarity function to a robust image annotation engine falls under the purview of CBIR. This characterization of CBIR as a field of study places it at a unique juncture within the scientific community. While we witness continued effort in solving the fundamental open problem of robust image understanding, we also see people from different fields, e.g., computer vision, machine learning, information retrieval, human-computer interaction, database systems, Web and data mining, information theory, statistics, and psychology contributing and becoming part of the CBIR community [276]. Moreover, a lateral bridging of gaps between some of these research communities is being gradually brought about as a by-product of such contributions, the impact of which can potentially go beyond CBIR. Again, what we see today as a few cross-field publications may very well spring into new fields of study in the future.

Amidst such marriages of fields, it is important to recognize the shortcomings of CBIR as a real-world technology. One problem with all current approaches is the reliance on visual similarity for judging semantic similarity, which may be problematic due to the *semantic gap* [242] between low-level content and higher-level concepts. While this intrinsic difficulty in solving the core problem cannot be denied, we believe that the current state-of-the-art in CBIR holds enough promise and maturity to be useful for real-world applications, if aggressive attempts are made. For example, Google and Yahoo! are household names today, primarily due to the benefits reaped through their use, despite the fact that robust text understanding is still an open problem. Online photo-sharing has become extremely popular with Flickr [88] which hosts hundreds of millions of pictures with diverse content. The video sharing and distribution forum YouTube has also brought in a new revolution in multimedia usage. Of late, there is renewed interest in the media about potential real-world applications of CBIR and image analysis technologies [233, 68, 48]. We envision that image retrieval will enjoy a success story in the coming years. We also sense a paradigm shift in the goals of the next-generation CBIR researchers. The need of the hour is to establish how this technology can reach out to the common man the way text-retrieval techniques have. Methods for visual similarity, or even semantic similarity (if ever perfected), will remain techniques for building systems. What the average end-user can hope to gain from using such a system is a different question altogether. For some applications, visual similarity may in fact be more critical than semantic similarity. For others, visual similarity may have little significance. Under what scenarios a typical user feels the need for a CBIR system, what the user sets out to achieve with the system, and how she expects the system to aid in this process, are some of the key questions that need to be answered in order to produce a successful system design. Unfortunately, user studies of this nature have been scarce so far.

Comprehensive surveys exist on the topic of CBIR [1, 224, 242, 247], all of which deal primarily with work prior to the year 2000. Surveys also exist on closely related topics such as relevance feedback [320], high-dimensional indexing of multimedia data [19], face recognition [313] (useful for face based image retrieval), applications of CBIR to medicine [198], and applications to art and cultural imaging [39]. Multimedia information retrieval, as a broader research area covering video, audio,

image, and text analysis has been extensively surveyed [235, 157]. In our current survey, we restrict the discussion to image-related research only.

One of the reasons for writing this survey is that CBIR, as a field, has grown tremendously after the year 2000 in terms of the people involved and the papers published. Lateral growth has also occurred in terms of the associated research questions addressed, spanning various fields. To validate the hypothesis about growth in publications, we conducted a simple exercise. We searched for publications containing the phrases “Image Retrieval” using Google Scholar [100] and the digital libraries of ACM, IEEE and Springer, within each year from 1995 to 2005. In order to account for (a) the growth of research in computer science as a whole and (b) Google’s yearly variations in indexing publications, the Google Scholar results were normalized using the publication count for the word “computer” for that year. A plot on another young and fast-growing field within pattern recognition, support vector machines (SVM), was generated in a similar manner for comparison. The results can be seen in Fig. 2.1. Not surprisingly, the graph indicates similar growth patterns for both fields, although SVM has had faster growth. These trends indicate, given the implicit assumptions, a roughly exponential growth in interest in image retrieval and closely related topics. We also observe particularly strong growth over the last five years, spanning new techniques, support systems, and application domains.

In this chapter, we comprehensively survey, analyze, and quantify current progress and future prospects of image retrieval. A possible organization of the various facets of image retrieval as a field is shown in Fig. 2.2. Note that the treatment is limited to progress mainly in the current decade, and only includes work that involves visual analysis in part or full. For the purpose of completeness, and better readability for the uninitiated, we have introduced key contributions of the earlier years in Sec. 2.1.1. Image retrieval purely on the basis of textual meta-data, Web link structures, or linguistic tags is excluded.

2.1.1 The Early Years

The years 1994-2000 can be thought of as the initial phase of research and development on image retrieval by content. The progress made during this phase was

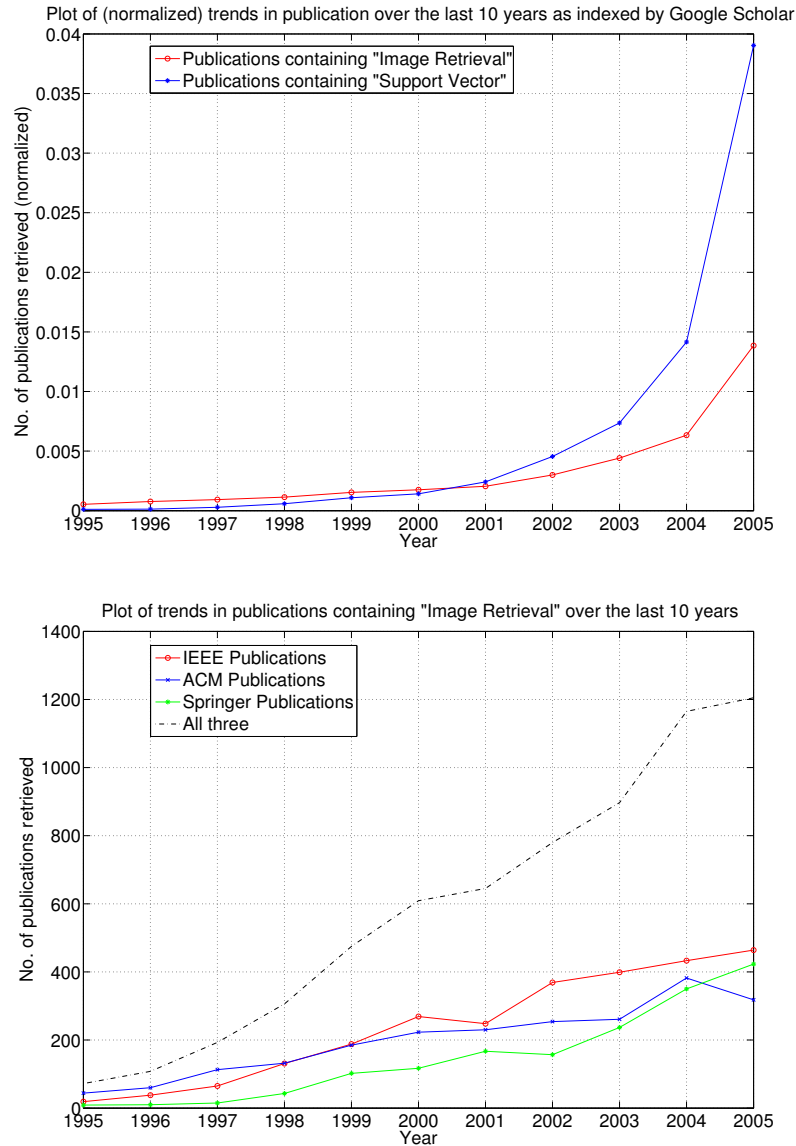


Figure 2.1. A study of post-1995 publications in CBIR. *Top:* Normalized trends in publications containing phrases “image retrieval” and “support vector” in them. *Bottom:* Publisher wise break-up of publication count on papers containing “image retrieval” in them.

lucidly summarized at a high-level in [242], which has had a clear influence on progress made in the current decade, and will undoubtedly continue to influence future work. Therefore, it is pertinent that we provide a brief summary of the ideas, influences, and trends of the early years (a large part of which originate in that survey) before describing the same for the new age. In order to do so, we

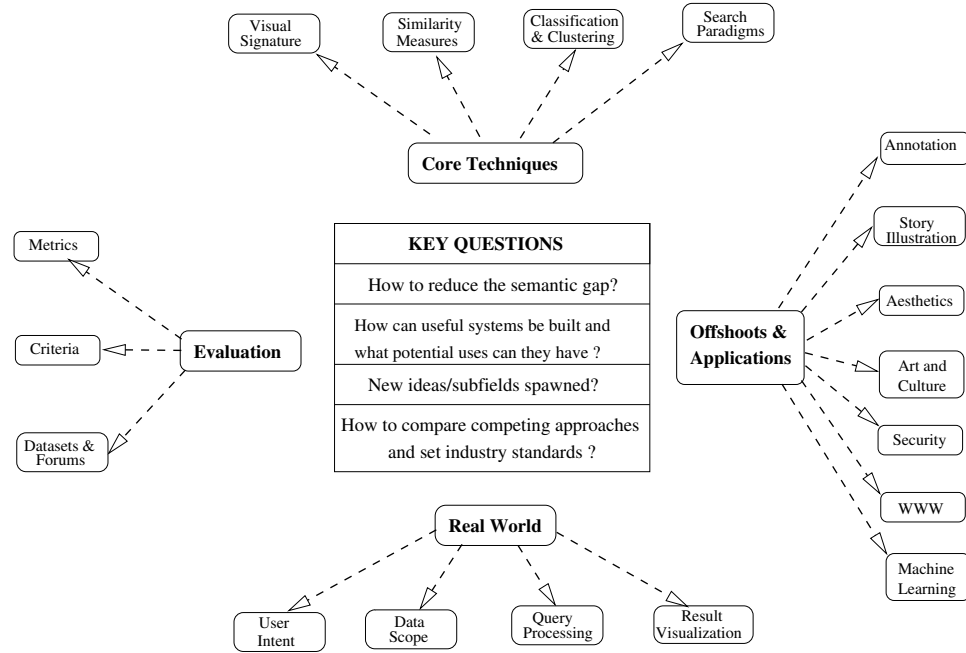


Figure 2.2. Our view of the many facets of image retrieval as a field of research.

first quote the various *gaps* introduced there that define and motivate most of the related problems:

- The *sensory gap* is the gap between the object in the world and the information in a (computational) description derived from a recording of that scene.
- The *semantic gap* is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.

While the former makes recognition from image content challenging due to limitations in recording, the latter brings in the issue of a user’s interpretations of pictures and how it is inherently difficult for visual content to capture them. We continue briefly summarizing key contributions of the early years that deal with one or more of these gaps.

In [242], the domains for image search were classified as *narrow* and *broad*, and to date this remains an extremely important distinction for the purpose of system design. As mentioned, narrow image domains usually have limited variability

and better-defined visual characteristics (e.g., aviation related pictures [2]), which makes content-based image search a tad bit easier to formulate. On the other hand, broad domains tend to have high variability and unpredictability for the same underlying semantic concepts (e.g., Web Images), which makes generalization that much more challenging. As recently noted in [121], narrow and broad domains pose a problem in image search evaluation as well, and appropriate modifications must be made to standard evaluation metrics for consistency. The survey also lists three broad categories of image search, (1) *search by association*, where there is no clear intent at a picture, but instead the search proceeds by iteratively refined browsing, (2) *aimed search*, where a specific picture is sought, and (3) *category search*, where a single picture representative of a semantic class is sought, for example, to illustrate a paragraph of text, as introduced in [50]. Also discussed are different kinds of domain knowledge that can help reduce the sensory gap in image search. Notable among them are concepts of syntactic similarity, perceptual similarity, and topological similarity. The overall goal therefore remains to bridge the semantic and sensorial gaps using the available visual features of images and relevant domain knowledge, to support the varied search categories, ultimately to satiate the user.

In the survey, extraction of visual content from images is split into two parts, namely image processing and feature construction. The question to ask here is what features to extract that will help perform meaningful retrieval. In this context, search has been described as a specification of *minimal invariant conditions* that model the user intent, geared at reducing the sensory gap due to accidental distortions, clutter, occlusion, etc. Key contributions in color, texture, and shape abstraction have then been discussed. Among the earliest use of color histograms for image indexing was that in [249]. Subsequently, feature extraction in systems such as QBIC [87], Pictoseek [94], and VisualSEEK [245] are notable. Innovations in color constancy, the ability to perceive the same color amidst environmental changes, were made by including specular reflection and shape into consideration [85]. In [120] color correlograms were proposed as enhancements to histograms, that take into consideration spatial distribution of colors as well. Gabor filters were successfully used for local shape extraction geared toward matching and retrieval in [184]. Daubechies' wavelet transforms were used for texture feature

extraction in the WBIIS system [279]. Viewpoint and occlusion invariant local features for image retrieval [231] received significant attention as a means to bridge the sensorial gap. Work on local patch-based salient features [262] found prominence in areas such as image retrieval and stereo matching. Perceptual grouping of images, important as it is for identifying objects in pictures, is also a very challenging problem. It has been categorized in the survey as strong/weak segmentation (data-driven grouping), partitioning (data-independent grouping, e.g., fixed image blocks), and sign location (grouping based on a fixed template). Significant progress had been made in field of image segmentation, e.g., [322], where snake and region growing ideas were combined within a principled framework, and [237], where spectral graph partitioning was employed for this purpose. From segments come shape and shape matching needs. In [65], elastic matching of images was successfully applied to sketch-based image retrieval. Image representation by multi-scale contour models were studied in [192]. The use of graphs to represent spatial relationships between objects, specifically geared toward medical imaging, was explored in [212]. In [244], 2D-strings [36] were employed for characterizing spatial relationships among regions. A method for automatic feature selection was proposed in [250]. In [242], the topic of visual content description was concluded with a discussion on the advantages and problems of image segmentation, along with approaches that can avoid strong segmentation but still characterize image structure well enough for image retrieval. In the current decade, many region-based methods for image retrieval have been proposed that do not depend on strong segmentation. We discuss these and other new innovations in feature extraction in Sec. 2.2.1.

Once image features were extracted, the question remained as to how they could be indexed and matched against each other for retrieval. These methods essentially aimed to reduce the semantic gap as much as possible, sometimes reducing the sensorial gap as well in the process. In [242], similarity measures were grouped as feature-based matching (e.g., [249]), object silhouette based matching (e.g., [65]), structural feature matching (hierarchically ordered sets of features, e.g., [288]), salient feature matching (e.g., geometric hashing [289]), matching at the semantic level (e.g., [78]), and learning based approaches for similarity matching (e.g., [297] and [285]). Closely tied to the similarity measures are how they

emulate the user needs, and more practically, how they can be modified stepwise with feedback from the user. In this respect, a major advance made in the user interaction technology for image retrieval was relevance feedback (RF). Important early work that introduced RF into the image retrieval domain included [223], which was implemented in their MARS system [225]. Methods for visualization of image query results were explored, for example, in [87, 35]. Content-based image retrieval systems that gained prominence in this era were, e.g., IBM QBIC [87], VIRAGE [105], and NEC AMORE [197] in the commercial domain, and MIT Photobook [209], Columbia VisualSEEK and WebSEEK [245], UCSB NeTra [176], and Stanford WBIIS [279] in the academic domain. In [242], practical issues such as system implementation and architecture, their limitations and how to overcome them, the user in the loop, intuitive result visualization, and system evaluation were discussed, and suggestions were made. Innovations of the new age based on these suggestions and otherwise are covered extensively in Sec. 2.2.

2.1.2 Real-world Image Search Systems

Not many image retrieval systems are deployed for public usage, save for Google Images or Yahoo! Images (which are based primarily on surrounding meta-data such as filenames and HTML text). Recently, a public domain search engine *Riya* (Fig. 2.3) has been developed which incorporates image retrieval and face recognition for searching pictures of people and products on the Web. It is also interesting to note that CBIR technology is being applied to domains as diverse as family album management, Botany, Astronomy, Mineralogy, and Remote sensing [309, 284, 52, 207, 232]. A publicly available similarity search tool [278] is being used for an on-line database of over 800,000 airline-related images [2, 240] (Fig. 2.3), the integration of similarity search functionality to a large collection of art and cultural images [95], and the incorporation of image similarity to a massive picture archive [251] of the renowned travel photographer Q.-T. Luong.

Automatic linguistic indexing of pictures - real-time (ALIPR), an automatic image annotation system [166] has been recently made public for people to try and have their pictures annotated. As mentioned earlier, presence of reliable tags with pictures are necessary for text-based image retrieval. As part of ALIPR search

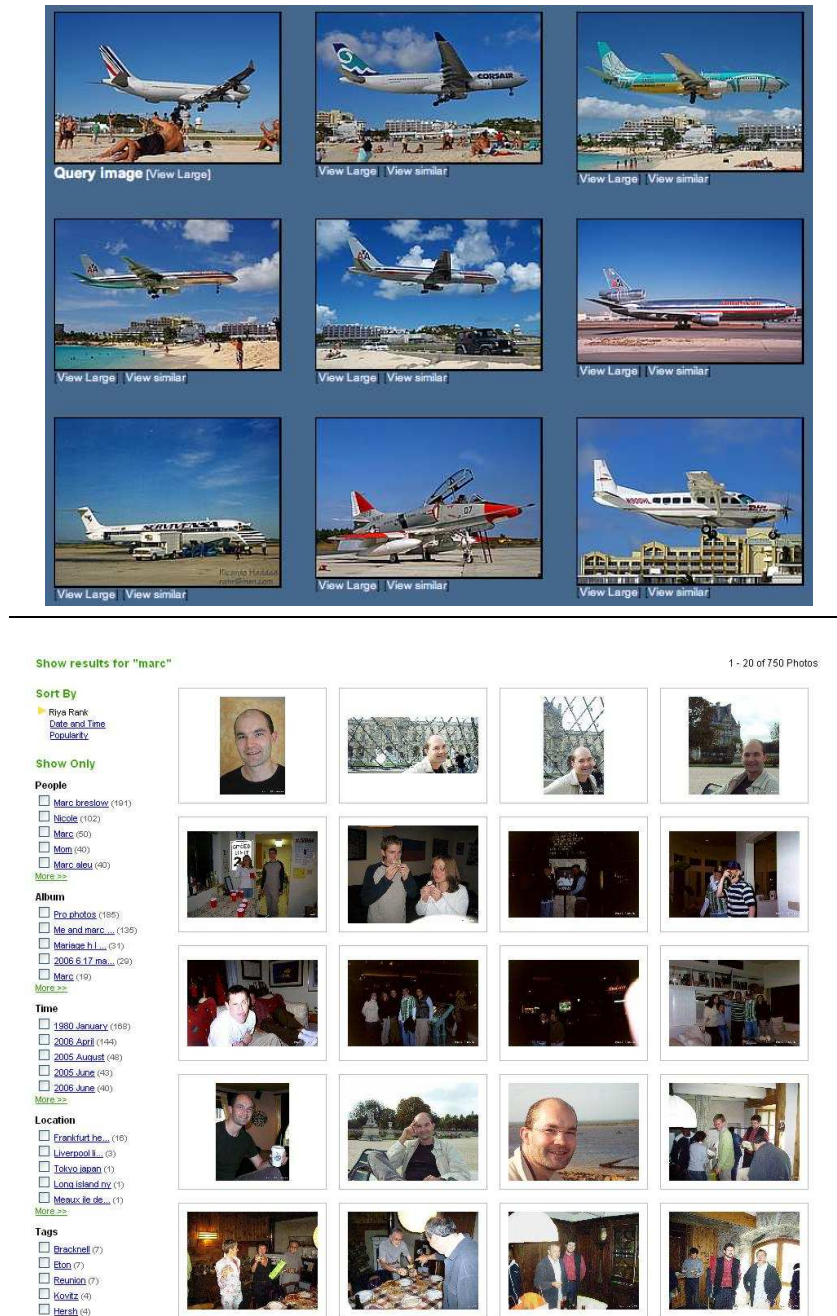


Figure 2.3. Real-world use of content-based image retrieval using color, texture, and shape matching. *Top:* <http://airliners.net>, is a photo-sharing community with more than a million airplane-related pictures. *Bottom:* <http://riya.com> is a collection of several million pictures.

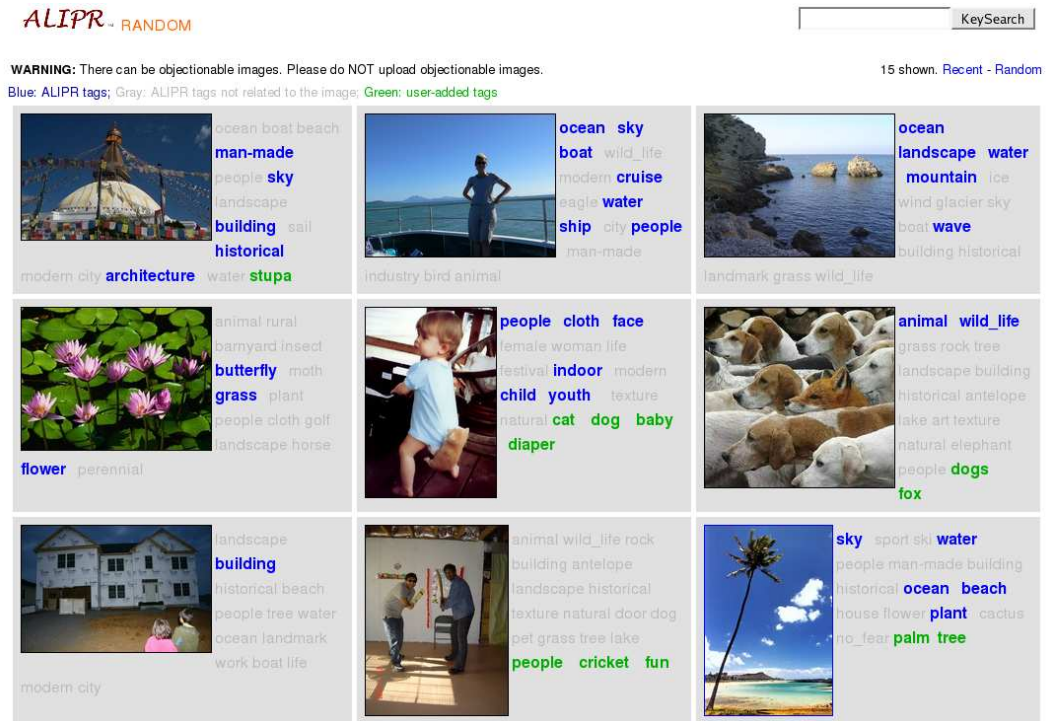


Figure 2.4. Real-world use of automatic image annotation, <http://alipr.com>. The screenshot shows a random set of uploaded pictures and the annotations given by ALIPR (shown in blue and gray) and by users (shown in green).

engine, an effort to automatically validate computer generated tags with human given annotation is being made to build a very large collection of searchable images (Fig. 2.4). Another work-in-progress is a Web image search system [136] that exploits visual features and textual meta-data using state-of-the-art algorithms, for a comprehensive search experience.

Discussion

Image analysis and retrieval systems have received widespread public and media interest of late [233, 68, 48]. It is reasonable to hope that in the near future, the technology will diversify to many other domains. We believe that the future of real-world image retrieval lies in exploiting both text-based and content-based search technologies. While the former is considered more reliable from a user view point, there is immense potential to combine the two to build robust image search engines that make the ‘hidden’ part of the Web images accessible.

2.2 Image Search Techniques: Addressing the Core Problem

Despite the effort made in the early years of image retrieval research (Sec. 2.1.1), we do not yet have a universally acceptable algorithmic means of characterizing human vision, more specifically in the context of interpreting images. Hence, it is not surprising to see continued effort in this direction, either building up on prior work, or exploring novel directions. Considerations for successful deployment of CBIR in the real-world are reflected by the research focus in this area.

By the nature of its task, the CBIR technology boils down to two intrinsic problems: (a) how to mathematically describe an image, and (b) how to assess the similarity between a pair of images based on their abstracted descriptions. The first issue arises because the original representation of an image, which is an array of pixel values, corresponds poorly to our visual response, let alone semantic understanding of the image. We refer to the mathematical description of an image for retrieval purposes as its *signature*. From the design perspective, the extraction of signatures and the calculation of image similarity cannot be cleanly separated. The formulation of signatures determines to a large extent the realm for definitions of similarity measures. On the other hand, intuitions are often the early motivating factors for designing similarity measures in a certain way, which in turn puts requirements on the construction of signatures.

In comparison with pre-2000 work in CBIR, a remarkable difference of recent years has been the increased diversity of image signatures. Advances have been made in both the derivation of new features, e.g., shape, and the construction of signatures based on these features, with the latter type of progress being more pronounced. The richness in the mathematical formulation of signatures grows together with the invention of new methods for measuring similarity. In the rest of this section, we will first address the extraction of image signatures, and then the methods for computing image similarity based on the signatures. In terms of methodology development, a strong trend which has emerged in recent years is the employment of statistical and machine learning techniques in various aspects of the CBIR technology. Automatic learning, mainly clustering and classification, is used to form either fixed or adaptive signatures, to tune similarity measures, and even

to serve as the technical core of certain searching schemes, e.g., relevance feedback. We thus not only discuss the influence of learning while addressing fundamentals issues of retrieval but also devote a subsection on clustering and classification, presented in the context of CBIR. Finally, we review different paradigms of searching with emphasis on relevance feedback. An actively pursued direction in image retrieval is to engage human in the searching process, i.e., to include *human in the loop*. Although in the very early days of CBIR, several systems were designed with detailed user preference specifications, the philosophy of engaging users in recent work has evolved toward more interactive and iterative schemes by leveraging learning techniques. As a result, the overhead for a user in specifying what she is looking for at the beginning of a search is much reduced.

2.2.1 Extraction of Visual Signature

Most CBIR systems perform feature extraction as a pre-processing step. Once obtained, visual features act as inputs to subsequent image analysis tasks such as similarity estimation, concept detection, or annotation. Figure 2.5 illustrates the procedure of generating image signatures and the main research problems involved. Following the order typical in feature extraction and processing, we present below the prominent recent innovations in visual signature extraction. The current decade has seen great interest in region-based visual signatures, for which segmentation is the quintessential first step. While we begin discussion with recent progress in image segmentation, we will see in the subsequent section how there is significant interest in segmentation-free techniques to feature extraction and signature construction.

Image Segmentation

To acquire a region-based signature, a key step is to segment images. Reliable segmentation is especially critical for characterizing shapes within images, without which the shape estimates are largely meaningless. We described above a widely used segmentation approach based on k -means clustering. This basic approach enjoys a speed advantage, but is not as refined as some recently developed methods. One of the most important new advances in segmentation employs the Normalized

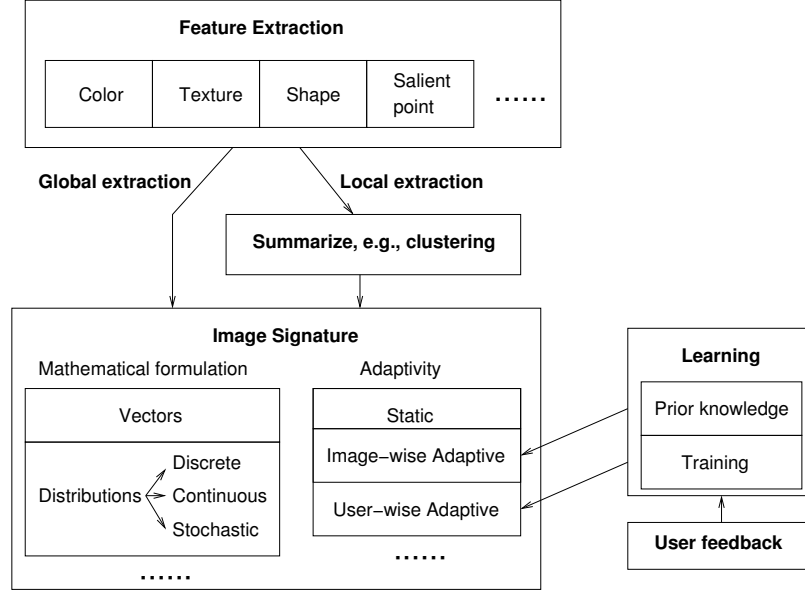


Figure 2.5. An overview on image signature formulation.

Cuts criterion [237]. The problem of image segmentation is mapped to a weighted graph partitioning problem where the vertex set of the graph is composed of image pixels and edge weights represent some perceptual similarity between pixel pairs. The normalized cut segmentation method in [237] is also extended to textured image segmentation by using cues of contour and texture differences [178], and to incorporate known partial grouping priors by solving a constrained optimization problem [303]. The latter has potential for incorporating real-world application-specific priors, e.g., location and size cues of organs in pathological images.

Searching of medical image collections has been an increasingly important research problem of late, due to the high-throughput, high-resolution, and high-dimensional imaging modalities introduced. In this domain, 3D brain magnetic resonance (MR) images have been segmented using Hidden Markov Random Fields and the Expectation-Maximization (EM) algorithm [312], and the spectral clustering approach has found some success in segmenting vertebral bodies from sagittal MR images [26]. Among other recent approaches proposed are segmentation based on the mean shift procedure [49], multi-resolution segmentation of low depth of field images [277], a Bayesian framework based segmentation involving the Markov chain Monte Carlo technique [260], and an EM algorithm based segmentation using a Gaussian mixture model [31], forming *blobs* suitable for image querying and re-

trieval. A sequential segmentation approach that starts with texture features and refines segmentation using color features is explored in [40]. An unsupervised approach for segmentation of images containing homogeneous color/texture regions has been proposed in [66].

While there is no denying that achieving good segmentation is a major step toward image understanding, some issues plaguing current techniques are computational complexity, reliability of good segmentation, and acceptable segmentation quality assessment methods. In the case of image retrieval, some of the ways of getting around this problem have been to reduce dependence on reliable segmentation [31], to involve every generated segment of an image in the matching process to obtain *soft* similarity measures [278], or to characterize spatial arrangement of color and texture using block-based 2-D multi-resolution hidden Markov models (MHMM) [161, 163]. Another alternative is to use perceptual grouping principles to hierarchically extract image structures [122]. In [55], probabilistic modeling of class-wise color segment interactions has been employed for the purpose of image categorization and retrieval, to reduce sensitivity to segmentation.

Major Types of Features

A feature is defined to capture a certain visual property of an image, either globally for the entire image, or locally for a small group of pixels. Most commonly used features include those reflecting color, texture, shape, and salient points in an image. In global extraction, features are computed to capture overall characteristics of an image. For instance, in a color layout approach, an image is divided into a small number of sub-images and the average color components, e.g., red, green, and blue, are computed for every sub-image. The overall image is thus represented by a vector of color components where a particular dimension of the vector corresponds to a certain sub-image location. The advantage of global extraction is the high speed for both extracting features and computing similarity. However, as evidenced by the rare use of color layout in recent work, global features are often too rigid to represent an image. Specifically, they can be over sensitive to location and hence fail to identify important visual characteristics. To increase the robustness to spatial transformation, the second approach to form signatures is by local extraction and an extra step of feature summarization.

In local feature extraction, a set of features are computed for every pixel using its neighborhood, e.g., average color values across a small block centered around the pixel. To reduce computation, an image may be divided into small non-overlapping blocks, and features are computed individually for every block. The features are still local because of the small block size, but the amount of computation is only a fraction of that for obtaining features around every pixel. Let the feature vectors extracted at block or pixel location (i, j) be $x_{i,j}$, $1 \leq i \leq m$, $1 \leq j \leq n$, where the image size $m \times n$ can vary. To achieve a global description of an image, various ways of summarizing the data set $\{x_{i,j}, 1 \leq i \leq m, 1 \leq j \leq n\}$ have been explored, leading to different types of signatures. A common theme of summarization is to derive a distribution for $x_{i,j}$ based on the data set.

Exploration of color features was active in the nascency of CBIR, with emphasis on exploiting color spaces (e.g., LUV) that seem to coincide better with human vision than the basic RGB color space. In recent years, research on color features has focused more on the summarization of colors in an image, that is, the construction of signatures out of colors. A set of color and texture descriptors tested for inclusion in the MPEG-7 standard, and well suited to natural images and video, is described in [183]. These include histogram-based descriptors, spatial color descriptors and texture descriptors suited for retrieval.

Texture features are intended to capture the granularity and repetitive patterns of surfaces within in a picture. For instance, grass land, brick walls, teddy bears, and flower petals differ in texture by smoothness as well as patterns. Their role in domain-specific image retrieval, such as in aerial imagery and medical imaging, is particularly vital due to their close relation to underlying semantics in these cases. Texture features have been studied for long in image processing, computer vision, and computer graphics [109], such as multi-orientation filter banks [179] and wavelet transforms [263]. In image processing, a popular way to form texture features is by using the coefficients of a certain transform on the original pixel values or more sophisticatedly, statistics computed from those coefficients. Examples of texture features using the *wavelet transform* and the discrete cosine transform can be found in [69, 162]. In computer vision and graphics, advances have been made in fields such as texture synthesis, where Markov statistical descriptors based on pairs of wavelet coefficients at adjacent location/orientation/scale in the images

are used [216]. Among the earliest work on the use of texture features for image retrieval are [184]. Texture descriptors, apt for inclusion in the MPEG-7, were broadly discussed in [183]. Such descriptors encode significant, general visual characteristics into standard numerical formats, that can be used for various higher-level tasks. A *thesaurus* for texture, geared toward aerial image retrieval, has been proposed in [175]. The texture extraction part of this thesaurus building process involves the application of a bank of Gabor filters [125] to the images, to encode statistics of the filtered outputs as texture features. Advances in textured region descriptors have been made, such as affine and photometric transformation invariant features that are also robust to the shape of the region in question [229]. While the target application is the more traditional stereo matching, it has been shown to have potential for textured image matching and segmentation as well. Advances in affine-invariant texture feature extraction, designed for texture recognition, have been made in [189], with the use of interest point detection for sparsity. Texture features at a point in the image are meaningful only as a function of its neighborhood, and the (effective) size of this neighborhood can be thought of as a *scale* at which these features are computed. Because a choice of scale is critical to the meaningfulness of such features, it has been explored as an automatic scale selection problem in [31], specifically to aid image retrieval.

Shape is a key attribute of segmented image regions, and its efficient and robust representation plays an important role in retrieval. Synonymous with shape representation is the way such representations are matched with each other. In general, over the years we have seen a shift from global shape representations, e.g., in [87], to more local descriptors, e.g., in [187, 13, 211], due to the typical modeling limitations. Representation of shape using discrete curve evolution to simplify contours is discussed in [150]. This contour simplification helps remove noisy and irrelevant shape features from consideration. A new shape descriptor for similarity matching, referred to as *shape context*, is proposed which is fairly compact yet robust to a number of geometric transformations [11]. In [13], curves are represented by a set of segments or *tokens*, whose feature representations (curvature and orientation) are arranged into a metric tree [47] for efficient shape matching and shape-based image retrieval. A dynamic programming (DP) approach to shape matching is proposed in [211], where shapes are approximated as sequences

of concave and convex segments. One problem with this approach is that computation of Fourier descriptors and moments is slow, although pre-computation may help produce real-time results. Continuing with Fourier descriptors, exploitation of both the amplitude and phase, and the use of Dynamic Time Warping (DTW) distance instead of Euclidean distance is shown to be an accurate shape matching technique in [10]. The rotational and starting point invariance otherwise obtained by discarding the phase information is maintained here by adding compensation terms to the original phase, thus allowing its exploitation for better discrimination.

Closely associated are approaches that model spatial relations among local image entities for retrieval. Much of the approaches to spatial modeling and matching have been influenced by earlier work on *iconic indexing* [36, 37] based on the theory of symbolic projections. Here, images are represented based on orthogonal projections of constituent entities, by encoding the corresponding bi-directional arrangement on the two axes as a *2D string* of entities and relationships. This way, image matching is effectively converted from a spatial matching problem to a one-dimensional matching one. Many variants of the 2D string model have been proposed since. In recent years, extensions such as 2D Be-string [283] have been proposed, where the symbolic encoding has been extended to represent entity locations more precisely, and avoid cutting entities along their bounding rectangles for improved complexity. Another work on iconic indexing can be found in [210], where a symbolic representation of real images, termed *virtual image* is proposed, consisting of entities and the binary spatial relations among them. Compared to traditional iconic representations and their variants, this approach allows more explicit scene representation and more efficient retrieval, once again without requiring the entities to be cut. In [14], a novel alternative to the previously discussed class of spatial models, *weighted walkthroughs*, is proposed. This representation allows quantitative comparison of entities, by incorporating the spatial relationships among each pair of pixels from the two entities. These quantitative relations allow images to be represented by attributed relational graphs (ARG), which essentially makes the retrieval problem one of graph comparison, resulting in improved retrieval performance. This idea has been extended to spatial modeling of 3D objects, in [12]. Other image models that capture spatial arrangements between local features such as interest points, are discussed in the following paragraph.

Features based on local invariants such as *corner points* or *interest points*, traditionally used for stereo matching, are being used in image retrieval as well. Scale and affine invariant interest points that can deal with significant affine transformations and illumination changes have been shown effective for image retrieval [189]. In similar lines, wavelet-based *salient points* have been used for retrieval [255]. In more recent work, the earth mover’s distance [221] has been used for matching locally invariant features in [103], for the purpose of image matching. The significance of such special points lies in their compact representation of important image regions, leading to efficient indexing and good discriminative power, especially in object-based retrieval. In this domain, there has been a paradigm shift from global feature representations to local descriptors, as evidenced by a large number of recent publications. Typically, object categories or visual classes are represented by a combination of local descriptors and their spatial distributions, sometimes referred to collectively as part-based models. Variations usually arise out of the ‘prior’ on the geometry imposed on the spatial relationship between the local parts, with extremes being fully independent (bag of features, each representing a part or region), and fully connected (constellation model, [83]). A fully connected model essentially limits the number of parts that can be modeled, since the algorithm complexity grows exponentially with it. As a compromise, sparser topologies have been proposed, such as the star topology [84], a hierarchy, with the lowest levels corresponding to local features [20], and a geometry where local features are spatially dependent on their nearest neighbors [28]. Model learning and categorization performance achieved in [83] has been improved upon, particularly in learning time, using contextual information and *boosting*, in [6, 4]. A recent work [306] uses segmentation to reduce the number of salient points for enhanced object representation. A discussion on the pros and cons of different types of color interest points used in image retrieval can be found in [102], while a comparative performance evaluation of the various proposed interest point detectors is reported in [188]. The application of salient point detection for related feature extraction has also been explored. For example, interest point detectors have been employed for sparse texture representation, for the purpose of texture recognition, in [152].

Construction of Signatures from Features

In Fig. 2.5, according to mathematical formulations, we summarize the types of signatures roughly into vectors and distributions. As will be discussed in details below, histograms and region-based signatures can both be regarded as sets of weighted vectors, and when the weights sum up to one, these sets are equivalent to discrete distributions (discrete in the sense that the support is finite). Our discussion will focus on region-based signature and its mathematical connection with histograms because it is the most exploited type of image signature. We note however, that distributions extracted from a collection of local feature vectors can be of other forms, for instance, a continuous density function [69], or even a spatial stochastic model [164]. A continuous density in general is more precise to describe a collection of local feature vectors than a discrete distribution with finitely many support vectors. A stochastic model moves beyond a continuous density by taking into account spatial dependence among local feature vectors. For special kinds of images, we may need these sophisticated statistical models to characterize them. For instance, in [164], it is noted that spatial relationship among pixels is crucial for capturing Chinese ink painting styles. On the other hand, more sophisticated statistical models are computationally costly and less intuitive.

In earlier work, histogram was a widely used form of distribution. Suppose the feature vectors are denoted by $x_{i,j} \in \mathcal{R}^d$, the d -dimensional Euclidean space. To form a basic histogram, \mathcal{R}^d is divided into fixed bins and the percentage of $x_{i,j}$'s falling into each bin is calculated. Suppose there are k bins. A histogram can then be treated as a k -dimensional vector $(f_1, f_2, \dots, f_k)^t$, where f_l is the frequency of the l -th bin. Improvements over the basic histogram signature have been actively pursued. In [107], a multi-resolution histogram, together with its associated image matching algorithm, is shown to be effective in retrieving textured images. Computation of histograms at multiple resolutions continues to have the simplicity and efficiency of ordinary histograms, but it additionally captures spatial variations across images. In [128], Gaussian mixture vector quantization (GMVQ) is used to extract color histograms and shown to yield better retrieval than uniform quantization and vector quantization with squared error.

The disadvantages of treating histograms simply as vectors of frequencies are noted in [221]. The main issue is that the vector representation ignores the location

of bins used to generate the histogram. For measuring the closeness of distributions, the locations of histogram bins are vital. The Earth Movers Distance (EMD) is proposed in [221] to take into consideration bin locations. When EMD is used, histogram is mathematically a collection of feature vector and frequency pairs: $\{(z_1, f_1), (z_2, f_2), \dots, (z_k, f_k)\}$, where $z_l \in \mathcal{R}^d$ is the center or location of the l -th bin. It is shown in [156] that EMD, when applied to probability frequencies, is equivalent to the Mallows Distance proposed in the early 1970's [180], which is a true metric for general probability measures. A histogram is a special distribution in the sense that it is discrete, i.e., it takes only countably many different values (for practical interest, finitely many). Moreover, histograms for different images are usually derived using a fixed set of bins.

Once the histogram is viewed as $\{(z_1, f_1), (z_2, f_2), \dots, (z_k, f_k)\}$, a weighted set of vectors, a natural question to raise is why we have to employ a fixed set of bins located at z_1, \dots, z_k . A direct extension from histogram is to adaptively generate z_l and f_l together and also let the number of bins k depend on the image being handled. This is essentially the widely used region-based signature, as used in [67, 278]. Consider the data set $\{x_{i,j}, 1 \leq i, 1 \leq j\}$. Applying a clustering procedure, e.g., k -means, to the data set groups the feature vectors $x_{i,j}$ into \tilde{k} clusters such that the feature vectors in the same clusters tend to be tightly packed. Let the mean of $x_{i,j}$'s in the same cluster l be z'_l . We thus have acquired a summary of the data set: $\{(z'_1, f'_1), \dots, (z'_{k'}, f'_{k'})\}$, where f'_l is the percentage of $x_{i,j}$'s grouped into cluster l . The collection of pixels (i, j) for which $x_{i,j}$'s are in the same cluster forms a relatively homogeneous region because the common cluster forces closeness between the visual features in $x_{i,j}$'s. This is why clustering of local feature vectors is a widely used method to segment images, and also why we call the signature $\{(z'_1, f'_1), \dots, (z'_{k'}, f'_{k'})\}$ region-based.

With fixed bins, histograms of image feature vectors tend to be sparse in multi-dimensional space. In comparison, the region-based signature provides more compact description of images because it allows the representative vectors z'_l to adapt to images. In [67, 278], it is argued that region-based signature is more efficient computationally for retrieval, and it also gets around drawbacks associated with earlier propositions such as dimension reduction and color moment descriptors. Strictly speaking, a region-based signature is not merely a dynamic histogram represen-

tation, and despite the mathematical connections made above, is not necessarily motivated by the intention of generalizing histograms. The motivation for using region-based signature, as argued in [278], is that a relatively homogeneous region of color and texture is likely to correspond to an object in an image. Therefore, by extracting regions, we obtain, in a crude way, a collection of objects, and with objects in an image listed, it is easier to engage intuitions for defining similarity measures. Moreover, although we have z'_l , the mean of $x_{i,j}$'s in region l as a natural result of clustering, the description of the region can be expanded to include features not contained in z'_l , for instance, shape, which can only be meaningfully computed after the region has been formed.

Adaptive Image Signature

It is quite intuitive that the same set of visual features may not work equally well to characterize, say, computer graphics and photographs. To address this issue, learning methods have been used to tune signatures either based on images alone or by learning on-the-fly from user feedback. In Fig. 2.5, we categorize image signatures according to their adaptivity into static, image-wise adaptive, and user-wise adaptive. Static signatures are generated in a uniform manner for all the images.

Image-wise adaptive signatures vary according to the classification of images. The term semantic-sensitive coined in [278] reflects such a mechanism to adjust signatures, and is a major trait of the SIMPLIcity system in comparison to the predecessors. Specifically, images are classified into several types first, and then signatures are formed from different features for these types. Despite the appeal of semantic-sensitive retrieval as a general framework, the classification conducted in SIMPLIcity only involves a small number of pre-selected image types (graph vs. photograph, textured vs. non-textured). The classification method relies on prior knowledge rather than training, and hence is not set up for extension. More recently, semantic-sensitive features are also employed in a physics-motivated approach [205], where images are distinguished as either photo-realistic rendering or photograph.

Care must be taken to ensure that the added robustness provided by heterogeneous feature representation does not compromise on the efficiency of indexing and

retrieval. When a large number of image features are available, one way to improve generalization and efficiency is to work with a feature subset or impose different weights on the features. To avoid a combinatorial search, an automatic feature subset selection algorithm for SVMs is proposed in [286]. Some of the other recent, more generic feature selection propositions involve boosting [256], evolutionary searching [144], Bayes classification error [29], and feature dependency/similarity measures [191]. An alternative way of obtaining feature weights based on user logs has been explored in [199]. A survey and performance comparison of some recent algorithms on the topic can be found in [106].

Discussion

The various methods for visual signature extraction come with their share of advantages and limitations. While global features give the “big picture”, local features represent the details. Therefore, depending on the scale of the key content or pattern, an appropriate representation should be chosen. In this sense, hybrid representations may sometimes be more attractive but this may come at additional complexity. While segmentation is intended to recognize objects in a scene, precise segmentation still remains an open problem. Therefore, alternative approaches to characterize structure may be more suitable. However, such a representation may lose the charm of clear interpretability. Among different approaches to segmentation, there is often a trade-off between quality and complexity, which might lead to a difference in eventual search performance and speed. Hence, a choice on the image signature to be used should depend on the desirability of the system.

In contrast with the early years (Sec. 2.1.1), we have witnessed a major shift from global feature representations for images such as color histograms and global shape descriptors to local features and descriptors, such as salient points, region-based features, spatial model features, and robust local shape characterizations. It is not hard to imagine that this shift was triggered by a realization that the image domain was too deep for global features to reduce the semantic gap. Local features often correspond with more meaningful image components such as rigid objects and entities, which make association of semantics with image portions straightforward. The future in image feature or signature representation resides both in theory and practise. Many years of research has made it clear that emulating human

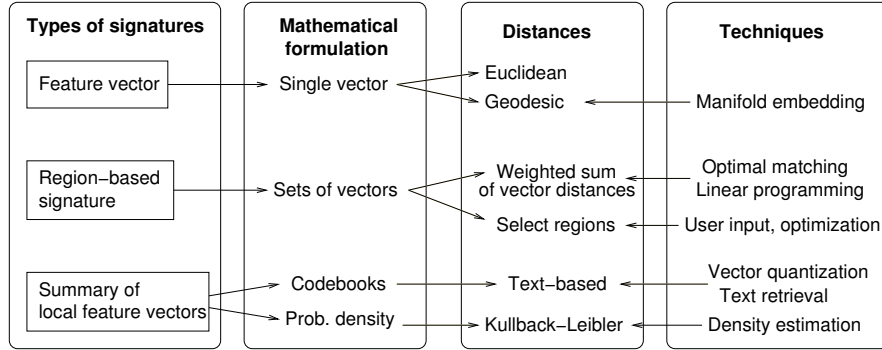


Figure 2.6. Different types of image similarity measures, their mathematical formulations and techniques for computing them.

vision is very challenging, but instead, practical approaches can help build useful systems. While the endeavor to characterize vision will likely continue, particularly in the core field of computer vision, practical approaches, e.g., fusion of local and global representations for top-down as well as a bottom-up representations, will potentially improve retrieval performance and user satisfaction in such systems. The availability of three dimensional image data and stereo image data, whenever obtainable, should be exploited to extract features more coherent with the human vision system. In summary, reducing the sensorial gap in tandem with the semantic gap should continue be a goal for the future.

2.2.2 Image Similarity using Visual Signature

Once a decision on the choice of image signatures is made, how to use them for accurate image retrieval is the next concern. There has been a large number of fundamentally different frameworks proposed in the recent years. Some of the key motivating factors behind the design of the proposed image similarity measures can be summarized as follows:

- agreement with semantics
- robustness to noise (invariant to perturbations)
- computational efficiency (ability to work real-time and in large-scale)
- invariance to background (allowing region-based querying)

- local linearity (i.e., following triangle inequality in a neighborhood)

The various techniques can be grouped according to their design philosophies, as follows:

- treating features as vectors, non-vector representations, or ensembles
- using region-based similarity, global similarity, or a combination of both
- computing similarities over linear space or non-linear manifold
- role played by image segments in similarity computation
- stochastic, fuzzy, or deterministic similarity measures
- use of supervised, semi-supervised, or unsupervised learning

Leaving out those discussed in [242], here we focus on some of the more recent approaches to image similarity computation.

Figure 2.6 shows the basic types of signatures, distances (‘dissimilarity measures’) exploited, and underlying techniques needed to calculate these distances. For each type of signatures, we also elucidate on its mathematical representation, which to a large extent determines the choice of distances and the employment of related methodologies. We will start discussion on the region-based signature since its widespread use occurred in the current decade. The technical emphasis on region-based signature is the definition of distance between sets of vectors, which is not as obvious as defining distance between single vectors. Research on this problem is further enriched by the effort to optimally choose a subset of regions pertaining to users’ interests and by that to increase robustness against inaccurate segmentation. Although global feature vectors had already been extensively used in the early years of CBIR, advances were achieved in recent years by introducing state-of-the-art learning techniques, e.g., manifold embedding. Research efforts have been made to search for nonlinear manifolds in which the geodesic distances may correspond better to human perception. Instead of describing an image by a set of segmented regions, summaries of local feature vectors such as codebook and probability density functions have been used as signatures. Codebooks are generated by vector quantization, and the codewords are sometimes treated symbolically

with text retrieval techniques applied to them. An effective way to obtain a density estimation is by fitting a Gaussian mixture model [110], and the Kullback-Leibler distance is often used to measure the disparity between distributions.

First consider an image signature in the form of a weighted set of feature vectors $\{(z_1, p_1), (z_2, p_2), \dots, (z_n, p_n)\}$, where z_i 's are the feature vectors and p_i 's are the corresponding weights assigned to them. The region-based signature discussed above bears such a form, so a histogram can be represented in this way. Let us denote two signatures by $I_m = \{(z_1^{(m)}, p_1^{(m)}), (z_2^{(m)}, p_2^{(m)}), \dots, (z_{n_m}^{(m)}, p_{n_m}^{(m)})\}$, $m = 1, 2$. A natural approach to defining a region-based similarity measure is to match $z_i^{(1)}$'s with $z_j^{(2)}$'s and then combine the distances between these vectors as a distance between sets of vectors.

One approach to matching [278] is by assigning a weight to every pair $z_i^{(1)}$ and $z_j^{(2)}$, $1 \leq i \leq n_1$, $1 \leq j \leq n_2$, and the weight $s_{i,j}$ indicates the significance of associating $z_i^{(1)}$ with $z_j^{(2)}$. One motivation for the soft matching is to reduce the effect of inaccurate segmentation on retrieval. The weights are subject to constraints, the most common ones being $\sum_i s_{i,j} = p_j^{(2)}$ and $\sum_j s_{i,j} = p_i^{(1)}$. Once the weights are determined, the distance between I_1 and I_2 is aggregated from the pair-wise distances between individual vectors:

$$D(I_1, I_2) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} s_{i,j} d(z_i^{(1)}, z_j^{(2)}), \quad (2.1)$$

where the vector distance $d(\cdot, \cdot)$ can be defined in diverse ways depending on the system. Other matching methods include the Hausdorff distance, where every $z_i^{(1)}$ is matched to its closest vector in I_2 , say $z_{i'}^{(2)}$, and the distance between I_1 and I_2 is the maximum among all $d(z_i^{(1)}, z_{i'}^{(2)})$. The Hausdorff distance, which is used for image retrieval in [145], is symmetrized by computing additionally the distance with the role of I_1 and I_2 reversed and choosing the larger one of the two distances:

$$D_H(I_1, I_2) = \max \left(\max_i \min_j d(z_i^{(1)}, z_j^{(2)}), \max_j \min_i d(z_j^{(2)}, z_i^{(1)}) \right). \quad (2.2)$$

One heuristic to decide the matching weights $s_{i,j}$ for the pair $(z_i^{(1)}, z_j^{(2)})$ is to seek $s_{i,j}$'s such that $D(I_1, I_2)$ in (2.1) is minimized subject to certain constraints on $s_{i,j}$. Suppose $\sum_i p_i^{(1)} = 1$ and $\sum_j p_j^{(2)} = 1$. This can always be made true

by normalization as long as there is no attempt to assign one image an overall higher significance than the other. In practice, $p_i^{(1)}$'s (or $p_j^{(2)}$'s) often correspond to probabilities and automatically yield unit sum. Since $p_i^{(1)}$ indicates the significance of region $z_i^{(1)}$ and $\sum_j s_{i,j}$ reflects the total influence of $z_i^{(1)}$ in the calculation of $D(I_1, I_2)$, it is natural to require $\sum_j s_{i,j} = p_i^{(1)}$, for all i , and similarly $\sum_i s_{i,j} = p_j^{(2)}$, for all j . Additionally, we have the basic requirement $s_{i,j} \geq 0$ for all i, j . The definition of the distance is thus

$$D(I_1, I_2) = \min_{s_{i,j}} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} s_{i,j} d(z_i^{(1)}, z_j^{(2)}), \quad (2.3)$$

subject to $\sum_j s_{i,j} = p_i^{(1)}$, for all i , $\sum_i s_{i,j} = p_j^{(2)}$, for all j , and $s_{i,j} \geq 0$ for all i, j . This distance is precisely the Mallows distance in the case of discrete distributions [180].

The Earth Mover's Distance [221] (EMD) proposed early in the decade represents another soft matching scheme for signatures in the form of sets of vectors. The measure treated the problem of image matching as one of "moving" components of the color histograms of images from one to the other, with minimum effort, synonymous with moving earth piles to fill holes. When p_i and p'_j are probabilities, EMD is equivalent to the Mallows distance. Another useful matching based distance is the IRM (integrated region matching) distance [165]. The IRM distance uses the most similar highest priority (MSHP) principle to match regions. The weights $s_{i,j}$ are subject to the same constraints as in the Mallows distance, but $D(I_1, I_2)$ is not computed by minimization. Instead, the MSHP criterion entails that a pair of regions across two images with the smallest distance among all the region pairs ought to be given the highest priority in matching, that is, to be assigned with a maximum valid weight $s_{i,j}$. The matching is conducted recursively until all the region weights are consumed, i.e., $\sum_j s_{i,j} = p_i^{(1)}$ and $\sum_i s_{i,j} = p_j^{(2)}$ have been achieved for all i and j . IRM is significantly faster to compute than the Mallows distance and has been found comparable in terms of retrieval results.

Improvements over the basic matching idea have been made from different perspectives. These include tuning features according to image types, choosing region weights in more sophisticated ways, improving robustness against inaccurate segmentation, and speeding up retrieval. In the SIMPLIcity system [278], a pre-

liminary categorization (e.g., graph vs. photograph, textured vs. non-textured) is applied to images and different sets of features are used for each category. Region based image retrieval, under the assumption of a hidden semantic concept underlying image generation, is explored in [311]. Here, a uniform, sparse region-based *visual dictionary* is obtained using self-organizing map (SOM) based quantization, and images/regions are assumed to be *generated* probabilistically, conditional on hidden or latent variables that reflect on their underlying semantics. A framework for region-based image retrieval, with particular focus on efficiency, is proposed in [132]. Here, vector quantization (VQ) is employed to build a region codebook from training images, each entry sparsely or compactly represented, with distinct advantages of efficiency and effectiveness in each case. To further speed up retrieval, a tree-structured clustering is applied to images to narrow down the search range [72]. The system first uses a vector signature to decide which cluster an image belongs to, and then uses the region-based signature and the IRM distance to compare the query with images in the chosen cluster.

A variation of IRM is attempted in [41] to employ fuzziness to account for inaccurate segmentation to a greater extent. A new representation for object retrieval in cluttered images, without relying on accurate segmentation is proposed in [6]. Here, image model learning and categorization is improved upon using contextual information and boosting algorithms. A windowed search over location and scale is shown more effective in object-based image retrieval than methods based on inaccurate segmentation [119]. A hybrid approach involves the use of rectangular blocks for coarse foreground/background segmentation on the user's query region-of-interest (ROI), followed by a database search using only the foreground regions [54].

Without user input, image similarity measures usually attempt to take all the regions in an image into consideration. This may not be the best practice when users' interest is more specifically indicated than an example query image. For instance, if the query is a sketch drawn by a user, it may be meaningless to let the left out areas in the sketch affect image comparison. It can be more desirable to match the sketch to only a relevant subset of regions automatically determined by the retrieval system, as explored in [145].

Even if the user starts searching with an example query image, it is sometimes assumed that he or she is willing to specify a portion of the image as of interest. This argument has led to the concept of region-based querying. The Blobworld system [31], instead of performing image to image matching, lets users select one or more homogeneous color-texture segments or *blobs*, as region(s) of interest. For example, if one or more segmented blobs identified by the user roughly correspond to a typical “tiger”, then her search becomes equivalent to searching for the “tiger” object within images. For this purpose, the pictures are segmented into blobs using the E-M algorithm, and each blob b_i is represented as a color-texture feature vector \mathbf{v}_i . Given a query blob b_i , and every blob b_j in the database, the most similar blob has score

$$\mu_i = \max_j \exp \left(\frac{(\mathbf{v}_i - \mathbf{v}_j)^T \Sigma (\mathbf{v}_i - \mathbf{v}_j)}{2} \right), \quad (2.4)$$

where matrix Σ corresponds to user-adjustable weights on specific color and texture features. The similarity measure is further extended to handle compound queries using fuzzy logic. While this method can lead to more precise formulation of user queries, and can help users understand the computer’s responses better, it also requires greater involvement from and dependence on them. For finding images containing scaled or translated versions of query objects, retrieval can also be performed without any explicit involvement of the user [203].

As discussed previously, regions are obtained by segmenting images using local feature vectors. Roughly speaking, region-based signatures can be regarded as a result of summarizing these feature vectors. Along the line of using a summary of local feature vectors as the signature, there are other approaches explored. For instance, in [122], primitive image features are hierarchically and perceptually grouped and their inter-relationships are used to characterize structure [122]. Another approach is the use of vector quantization (VQ) on image blocks to generate *codebooks* for representation and retrieval, taking inspiration from data compression and text-based strategies [321]. For textured images, segmentation is not critical. Instead, distributions of the feature vectors are estimated and used as signatures. Methods for texture retrieval using the Kullback-Leibler (K-L) divergence have been proposed in [69, 185]. The K-L divergence, also known as the *relative entropy*, is an asymmetric information theoretic measure of difference between two

distributions $f(\cdot)$ and $g(\cdot)$, defined as

$$K(f, g) = \int_{-\infty}^{+\infty} f(x) \log \frac{f(x)}{g(x)} dx, \quad K(f, g) = \sum_x f(x) \log \frac{f(x)}{g(x)} \quad (2.5)$$

in the continuous and discrete cases respectively. Fractal block code based image histograms have been shown effective in retrieval on texture databases [215]. The use of the MPEG-7 content descriptors to train self-organizing maps (SOM) for image retrieval is explored in [149].

When images are represented as single vectors, many authors note the apparent difficulty in measuring perceptual image distance by metrics in any given *linear* feature space. One approach to tackle this issue is to search for a non-linear manifold in which the image vectors lie, and to replace the Euclidean distance by the geodesic distance. The assumption here is that visual perception corresponds better with this non-linear subspace than the original linear space. Computation of similarity may then be more appropriate if performed non-linearly along the manifold. This idea is explored and applied to image similarity and ranking in [114, 269, 115, 112, 316]. Typical methods for learning underlying manifolds, which essentially amount to non-linear dimension reduction, are Locally-linear Embedding (LLE), Isomap, and multi-dimensional scaling (MDS) [64].

The different distance measures discussed so far have their own advantages and disadvantages. While simple methods lead to very efficient computation, which in turn make image ranking scalable - a quality that greatly benefits real-world applications, they often are not effective enough to be useful. Depending on the specific application and on the image signatures constructed, a very important step in the design of an image retrieval system is the choice of distance measure. Factors that differ across various distance measures include type of input, method of computation, computational complexity, and whether the measure is a metric or not. In table 2.1, we summarize the distance measures according to these factors, for ease of comparison.

In the previous subsection, we discussed tuning image signatures by categorizing images or by learning from user preferences. A tightly related issue is to tune image similarity measures. It is in fact impossible to completely set apart the two types of adaptivity since tuning signatures ultimately results in the change of sim-

Table 2.1. Popular distances measures used for image similarity computation.

Distance Measure	Input	Computation	Complexity	Metric
Euclidean (L^2 norm)	$\vec{X}_a, \vec{X}_b \in \mathbb{R}^n$ (vectors)	$\vec{X}_a \cdot \vec{X}_b$	$\Theta(n)$	Yes
Weighted Euclidean	$\vec{X}_a, \vec{X}_b \in \mathbb{R}^n$ $W \in \mathbb{R}^n$ (vec. + wts.)	$\vec{X}_a^T [W] \vec{X}_b$ [.] \leftarrow diagonalize	$\Theta(n)$	Yes
Hausdorff	Vector sets: $\{\vec{X}_a^{(1)}, \dots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \dots, \vec{X}_b^{(q)}\}$	See Eqn. 2.2	$\Theta(pqn)$ ($d(\cdot, \cdot) \leftarrow L^2$ norm)	Yes
Mallows	Vector sets: $\{\vec{X}_a^{(1)}, \dots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \dots, \vec{X}_b^{(q)}\}$ Signific.: S	See Eqn. 2.3	$\Theta(pqn)$ + variable part	Yes
IRM	Vector sets: $\{\vec{X}_a^{(1)}, \dots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \dots, \vec{X}_b^{(q)}\}$ Signific.: S	See Eqn. 2.3	$\Theta(pqn)$ + variable part	No
K-L divergence	$\vec{F}, \vec{G} \in \mathbb{R}^m$ (histograms)	$\sum_x F(x) \log \frac{F(x)}{G(x)}$	$\Theta(m)$	No

ilarity. Referring a tuning method in one way or the other is often merely a matter of whichever is easier to understand. Automatic learning of image similarity measures with the help of contextual information has been explored in [292]. In the case that a valid pairwise image similarity metric exists despite the absence of an explicit vectored representation in some metric space, *anchoring* can be used for ranking images [204]. Anchoring involves choosing a set of representative *vantage* images, and using the similarity measure to map an image into a vector. Suppose there exists a valid metric $d(F_i, F_j)$ between each image pair, and a chosen set of K vantage images $\{A_1, \dots, A_K\}$. A *vantage space transformation* $V : F \rightarrow \mathcal{R}^K$ then maps each image F_i in the database to a vectored representation $V(F_i)$ as follows:

$$V(F_i) = \langle d(F_i, A_1), \dots, d(F_i, A_K) \rangle. \quad (2.6)$$

With the resultant vector embedding, and after similarly mapping a query image in the same space, standard ranking methods may be applied for retrieval. When

images are represented as ensembles of feature vectors, or underlying distributions of the low-level features, visual similarity can be ascertained by means of non-parametric tests such as Wald-Wolfowitz [253] and K-L divergence [69]. When images are conceived as bags of feature vectors corresponding to regions, multiple-instance learning (MIL) can be used for similarity computation [310].

A number of probabilistic frameworks for CBIR have been proposed in the last few years [130, 268]. The idea in [268] is to integrate feature selection, feature representation, and similarity measure into a combined Bayesian formulation, with the objective of minimizing the probability of retrieval error. One problem with this approach is the computational complexity involved in estimating probabilistic similarity measures. The complexity is reduced in [266] using VQ to approximately model the probability distribution of the image features.

Discussion

As shown in Fig. 2.6, similarity computation can be performed with feature vectors, region-based signatures, or summarized local features. The main advantage of single vector representing an image is that algebraic and geometric operations can be performed efficiently and in a principled fashion. However, many such representations lack the necessary detail to represent complex image semantics. For example, a picture of two cups on a plate by the window sill cannot easily be mapped to a finite vector representation, simply because the space of component semantics is extremely large, in practice. Instead, if a concatenation of region descriptors is used to represent a picture, it is more feasible to map component semantics (e.g., cup, window) to the image regions. On the other hand, extracting semantically coherent regions is in itself very challenging. Probabilistic representations can potentially provide an alternative, allowing rich descriptions with limited parametrization.

The early years (Sec. 2.1.1) showed us the benefits as well as the limitations of feature vector representations. They also paved the way for the new breed of region-based methods, which have now become more standard than ever before. The idea of region-based image querying also gained prominence in the last few years. Many new salient feature based spatial models were introduced, particularly for recognizing objects within images, building up mostly on pre-2000 work. The

idea that image similarity is better characterized by geodesic distances over a non-linear manifold embedded in the feature space has improved upon earlier notions of a linear embedding of images. A number of systems have also been introduced for public usage in the recent years. The future of image similarity measures lie in many different avenues. The subjectivity in similarity needs to be incorporated more rigorously into image similarity measures, to achieve what can be called *personalized* image search. This can also potentially incorporate ideas beyond the semantics, such as aesthetics and personal preferences in style and content. Extensions of the idea of non-linear image manifolds to incorporate the whole spectrum of natural images, and to allow adaptability for personalization, are avenues to look at. While development of useful systems remains critical, the ever-eluding problem of reducing the semantic gap needs concerted attention.

2.2.3 Clustering and Classification

Over the years it has been observed that it is too ambitious to expect a single similarity measure to produce robust perceptually meaningful ranking of images. As an alternative, attempts have been made to augment the effort with learning-based techniques. In table 2.2, for both clustering and classification, we summarize the augmentations to traditional image similarity based retrieval, the specific techniques exploited, and the limitations respectively.

Image classification or categorization has often been treated as a pre-processing step for speeding up image retrieval in large databases and improving accuracy, or performing automatic image annotation. Similarly, in the absence of labeled data, unsupervised clustering has often been found to be useful for retrieval speedup as well as improved result visualization. While image clustering inherently depends on a similarity measure, image categorization has been performed by varied methods that neither require nor make use of similarity metrics. Image categorization is often followed by a step of similarity measurement, restricted to those images in a large database that belong to the same visual class as predicted for the query. In such cases, the retrieval process is intertwined, whereby categorization and similarity matching steps together form the retrieval process. Similar arguments hold for clustering as well.

Table 2.2. Comparison of three learning techniques in context of image retrieval.

Augmentation (User Involvement)	Purpose	Techniques	Drawbacks
<i>Clustering</i> (minimal)	Meaningful result visualization, faster retrieval, efficient storage	Side-information, kernel mapping, k -means, hierarchical, metric learning [42] [110] [234] [292]	Same low-level features, poor user adaptability
<i>Classification</i> (requires prior training data, not interactive)	Pre-processing, fast/accurate retrieval, automatic organization	SVM, MIL, statistical models, Bayesian classifiers, k -NN, trees [310] [110] [208]	Training introduces bias, many classes unseen
<i>Relevance Feedback</i> (significant, interactive)	Capture user and query specific semantics, refine rank accordingly	Feature re-weighting, region weighting, active learning, memory/mental retrieval, boosting [110] [223] [123] [79]	Same low level features, increased user involvement

In the recent years, considerable progress has been made in clustering and classification, with tremendously diverse target applications. It is not our intention here to provide a general review of these technologies. We refer to [110] for basic principles and a more comprehensive review. We will restrict ourselves to new methods and applications appeared in image retrieval and closely related topics.

Unsupervised clustering techniques are a natural fit when handling large, unstructured image repositories such as the Web. Figure 2.7 summarizes clustering techniques according to the principles of clustering and shows the applicability of different methods when the mathematical representation of learning instances varies. Again, we divide the instances to be clustered into three types: vectors, sets of vectors, and stochastic processes (including distributions), which are consistent with the categorization of image signatures discussed in the previous subsection. From the perspective of application, clustering specifically for Web images has received particular attention from the multimedia community, where meta-data is often available for exploitation in addition to visual features [280, 91, 24].

Clustering methods fall roughly into three types: pair-wise distance based, optimization of an overall clustering quality measure, and statistical modeling. The pair-wise distance based methods, e.g., linkage clustering and spectral graph partitioning, are of general applicability since the mathematical representation of the instances becomes irrelevant. They are particularly appealing in image retrieval because image signatures often have complex formulation. One disadvantage, however, is the high computational cost because we need to compute an order of n^2 pair-wise distances, where n is the size of the data set. In [315], a locality preserving spectral clustering technique is employed for image clustering in a way that unseen images can be placed into clusters more easily than with traditional methods. In CBIR systems which retrieve images ranked by relevance to the query image only, similarity information among the retrieved images is not considered. In this respect, [43] proposes the use of a new spectral clustering [237] based approach to incorporate such information into the retrieval process. In particular, clusters are dynamically generated, tailored specifically to the query image each time, to improve retrieval performance.

Clustering based on the optimization of an overall measure of the clustering quality is a fundamental approach explored since the early days of pattern recognition. The immensely popular method, k -means clustering, is one example. In k -means, the merit of a clustering result is measured by the sum of within-cluster distances between every vector and its cluster centroid. This criterion ensures that clusters generated are tight, a heuristic generally accepted. Here, if the number of clusters is not specified, a simple method to determine this number is to gradually increase it until the average distance between a vector and its cluster centroid is below a given threshold. A more sophisticated way to determine the number of clusters is the competitive agglomeration algorithm, with application to image clustering [228]. In [101], an unsupervised clustering approach for images has been proposed using the Information Bottleneck (IB) principle. The proposed method works for discrete (histograms) as well as continuous (Gaussian mixture) image representations. Clustering based on the IB principle [257] can be summarized as follows: given two variables A (which we try to compress/cluster) and B (which contains relevant information), and their joint distribution $Pr(A, B)$, we seek to perform soft partitioning of A by a probabilistic mapping V , i.e., $Pr(V|A)$, in a

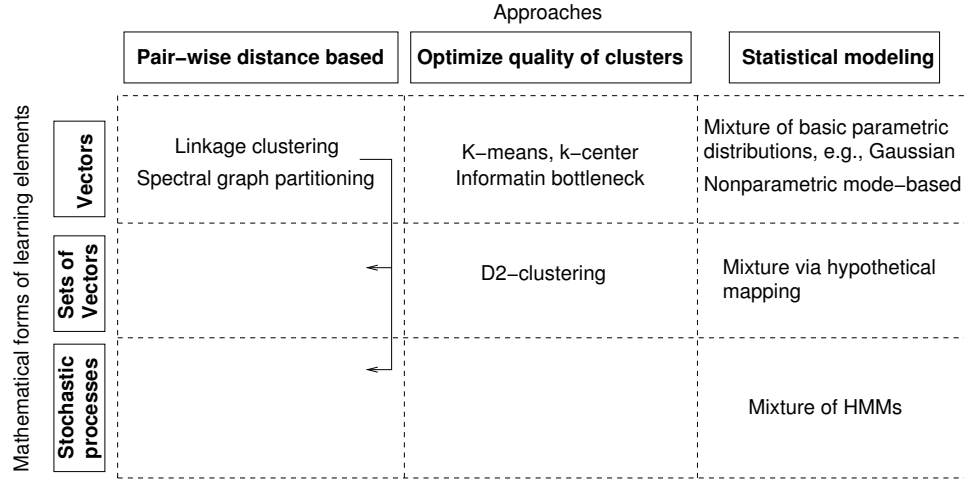


Figure 2.7. Paradigms of clustering methods and their scopes of applications.

way that the mutual information among A and V is minimized, while the relevant information among B and V is maximized.

In k -means clustering, a centroid vector is computed for every cluster. This centroid vector is chosen to minimize the sum of within-cluster distances. When the Euclidean distance is used, it can easily be shown that the centroid ought to be the average of the vectors in a cluster. For non-vector data, the determination of the centroid can be challenging. The extension of k -means to instances represented by sets of weighted vectors is made in [166], namely, the D2-clustering algorithm. The Mallows distance is used for region-based image signatures represented as sets of weighted arbitrary vectors. When the weights assigned to the vectors are probabilities, this representation is essentially a discrete distribution. The centroid for every cluster is also a discrete distribution, for which both the probabilities and the vectors in the support domain need to be solved. Although D2-clustering share the same intrinsic criterion of clustering as k -means, computationally, it is much more complex due to the complexity of the instances themselves. Large-scale linear programming is used for the optimization in D2-clustering. Another algorithm for clustering sets of vectors is developed using the IRM distance [160]. As compared with D2-clustering, this algorithm is similar in principle and significantly faster, but it has weaker optimization properties.

Statistical modeling is another important paradigm of clustering. The general idea is to treat every cluster as a pattern characterized by a relatively restrictive

distribution, and the overall data set is thus a mixture of these distributions. For continuous vector data, the most used distribution of individual vectors is the Gaussian distribution. By fitting a mixture of Gaussians to a data set, usually by the EM algorithm [186], we estimate the means and covariance matrices of the Gaussian components, which correspond to the center locations and shapes of clusters. One advantage of the mixture modeling approach is that it not only provides a partition of data but also yields an estimated density, which sometimes is itself desired [69]. The component in a mixture model is not always a multivariate distribution. For instance, in [164], the objects to be clustered are large areas of images, and every cluster is characterized by a 2-D MHMM. As long as a probability measure can be set up to describe a cluster, the mixture modeling approach applies seamlessly. When it is difficult to form a probability measure in a certain space, a mixture model can be established by clustering the data and mapping each cluster to a distance-preserving Euclidean space [166]. In this case, the mixture model is not used to yield clustering but to better represent a data set and eventually result in better classification.

Image categorization (classification) is advantageous when the image database is well-specified, and labeled training samples are available. Domain-specific collections such as medical image databases, remotely sensed imagery, and art and cultural image databases are examples where categorization can be beneficial. Classification is typically applied for either automatic annotation, or for organizing unseen images into broad categories for the purpose of retrieval. Here we discuss the latter. Classification methods can be divided into two major branches: discriminative modeling and generative modeling approaches. In discriminative modeling, classification boundaries or posterior probabilities of classes are estimated directly, e.g., SVM and decision trees. In generative modeling, the density of data within each class is estimated and the Bayes formula is then used to compute the posterior. Discriminative modeling approaches are more direct at optimizing classification boundaries. On the other hand, the generative modeling approaches are easier to incorporate prior knowledge and can be used more conveniently when there are many classes.

Bayesian classification is used for the purpose of image retrieval in [264]. A textured/non-textured and graph/photograph classification is applied as a pre-

processing to image retrieval in [278]. Supervised classification based on SVMs has been applied to images in [96]. A more recent work describes an efficient method for processing multimedia queries in an SVM based supervised learning framework [208]. SVMs have also been used in an MIL framework in [42]. In the MIL framework, a set of say l training images for learning an image category are conceived as labeled bags $\{(B_1, y_1), \dots, (B_l, y_l)\}$, where each bag B_i is a collection of instances $v_{ij} \in \mathbf{R}^m$. Each instance v_{ij} corresponds to a segmented region j of a training image i , and $y_i \in \{-1, +1\}$ indicating negative or positive example with respect to the category in question. The idea is to map these bags into a new feature space where SVMs can be trained for classification. Image classification based on a generative model for the purpose of retrieval is explored in [55].

Discussion

Clustering is a hard problem with two unknowns, i.e., the number of clusters, and the clusters themselves. In image retrieval, clustering helps in visualization and retrieval efficiency. The usual problems of clustering based applications appear here as well, whereby the clusters may not be representative enough or accurate for visualization. While supervised classification is more systematic, the availability of comprehensive training data is often scarce. In particular, the veracity of “ground truth” in image data itself is a subjective question.

Clustering and classification for the purpose of image retrieval received relatively less attention in the early years. The spotlight was on feature extraction and similarity computation. With the need for practical systems that scale well to billions of images and millions of users, practical hacks such as pre-clustering and fast classification have become critical. The popularization of new information-theoretic clustering methods and classification methods such as SVM and Boosting, have led to their extensive use in the image retrieval domain as well. New generative models such as Latent Dirichlet Allocation (LDA) and 2D-MHMM have made their way into image modeling and annotation. The future, in our opinion, lies in supervised and unsupervised generative models for characterizing the various facets of images and meta-data. There is often a lot of structured and unstructured data available with the images that can be potentially exploited through joint modeling, clustering, and classification. It is difficult to guess how much these methods

can help bridge the semantic or sensorial gap, but one thing is for sure: system implementations can greatly benefit in various ways from the efficiency that these learning-based methods can produce.

2.2.4 Relevance Feedback based Search Paradigms

The approach to search has an undeniable tie with the underlying core technology because it defines the goals and the means to achieve them. One way to look at the types of search is the modality (e.g., query by keyword/keyphrase, by example images, or a combination of both). Other ways to characterize search is by the nature and level of human and system interaction involved, and the user intent. In this section, we concentrate on the latter categorization, exploring the different search paradigms that affect how humans interact and systems interpret/respond.

Relevance feedback (RF) is a query modification technique which attempts to capture the user’s precise needs through iterative feedback and query refinement. It can be thought of as an alternative search paradigm, complementing other paradigms such as keyword based search. Ever since its inception in the CBIR community [223], a great deal of interest has been generated. In the absence of a reliable framework for modeling high-level image semantics and subjectivity of perception, the user’s feedback provides a way to learn case-specific query semantics. While a comprehensive review can be found in [320], here we present a short overview of recent work in RF, and the various ways these advances can be categorized. We group them here based on the nature of the advancements made, resulting in (possibly overlapping) sets of techniques that have pushed the frontiers in a common domain, which include (a) learning-based advancements, (b) feedback specification novelties, (c) user-driven methods, (d) probabilistic methods, (e) region-based methods, and (f) other advancements.

Learning-based Advancements

Based on the user’s relevant feedback, learning based approaches are typically used to appropriately modify the feature set or the similarity measure. However, in practise, a user’s RF results in only a small number of labeled images pertaining to each high-level concept. This, along with other unique challenges pertinent

to RF have generated interest in novel machine learning techniques to solve the problem, such as *one-class* learning, *active* learning, and *manifold* learning. To circumvent the problem of learning from small training sets, a discriminant-EM algorithm is proposed to make use of unlabeled images in the database for selecting more discriminating features [297]. On the other hand, it is often the case that the positive examples received due to feedback are more consistently located in the feature space than negative examples, which may consist of any irrelevant image. This leads to a natural formulation of *one-class* SVM for learning relevant regions in the feature space from feedback [44]. Let $\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$, $\mathbf{v}_i \in \mathbf{R}^d$ be a set of n positive training samples. The idea is to find a mapping $\Phi(\mathbf{v}_i)$ such that most samples are tightly contained in a hyper-sphere of radius R in the mapped space subject to regularization. The primal form of the objective function is given by

$$\min_{R, e, c} \left(R^2 + \frac{1}{kn} \sum_i e_i \right) \text{ subject to } \|\Phi(\mathbf{v}_i) - c\|^2 \leq R^2 + e_i, e_i \geq 0, i \in \{1, \dots, n\}. \quad (2.7)$$

Here, c is the hyper-sphere center in the mapped space, and $k \in [0, 1]$ is a constant that controls the trade-off between radius of the sphere and number of samples it can hold. Among other techniques, a principled approach to optimal learning from RF is explored in [222]. We can also view RF as an *active learning* process, where the learner chooses an appropriate subset for feedback from the user in each round based on her previous rounds of feedback, instead of choosing a random subset. Active learning using SVMs was introduced into RF in [259]. Extensions to active learning have also been proposed [97, 113]. In [115], it is conceived that image features reside on a manifold embedded in the Euclidean feature space. Under this assumption, relevant images to the query provided by RF, along with their nearest neighbors, are used to construct a sub-graph over the images. The geodesic distances, i.e., the shortest path on the graph between pairs of vertices representing image pairs, are then used to rank images for retrieval.

Feedback Specification Novelties

Traditionally, RF has engaged the user in multiple rounds of feedback, each round consisting of one set each of positive and negative examples in relation to the

intended query. However, recent work has introduced other paradigms of query specification that have been found to be either more intuitive, or more effective. Feedback based directly on image semantics characterized by manually defined image labels, and appropriately termed *semantic feedback*, is proposed in [300]. A well-known issue with feedback solicitation is that multiple rounds of feedback test the user’s patience. To circumvent this problem, user logs on earlier feedback can be used in query refinement, thus reducing the user engagement in RF, as shown in [118]. Innovation has also come in the form of the nature by which feedback is specified by the user. In [143], the notion of a multi-point query, where multiple image examples may be used as query and in intermediate RF step, is introduced. At each round of the RF, clusters of images found relevant based on the previous feedback step are computed, whose representatives form the input for the next round of RF. It is well known that there is generally an asymmetry between the sets of positive and negative image examples presented by the user. In order to address this asymmetry during RF when treating it as a two-class problem, a biased discriminant analysis based approach has been proposed in [318]. While most algorithms treat RF as a two-class problem, it is often intuitive to consider multiple groups of images as relevant or irrelevant [117, 201, 317]. For example, a user looking for *cars* can highlight groups of *blue* and *red* cars as relevant, since it may not be possible to represent the concept *car* uniformly in a visual feature space. Another novelty in feedback specification is the use of multi-level relevance scores, to indicate varying degrees of relevance [293].

User-driven Methods

While much of the past attempt at RF has focused on the machine’s ability to learn from the user feedback, the user’s point of view in providing the feedback has largely been taken for granted. Of late, there has been some interest in designing RF paradigms aimed to help users. In some new developments, there have been attempts at tailoring the search experience by providing the user with cues and hints for more specific query formulation [123, 200]. While the approach may still involve RF from the system point of view, it is argued that the human memory can benefit from cues provided, for better query formulation. A similar search paradigm proposed in [79, 80] models successive user response using a Bayesian,

information-theoretic framework. The goal is to ‘learn’ a distribution over the image database representing the mental image of the user and use this distribution for retrieval. Another well-known issue with human being in the loop is that multiple rounds of feedback are often bothersome for the user, which have been alleviated in [118] by making use of logs that contain earlier feedback given by that user. Recently, a manifold learning technique to capture user preference over a *semantic manifold* from RF is proposed in [170].

Probabilistic Methods

Probabilistic models, while popular in early years of image retrieval for tackling the basic problem, have found increasing patronage for performing RF in the recent years. Probabilistic approaches have been taken in [50, 248, 267]. In [50], the PicHunter system is proposed, where uncertainty about the user’s goal is represented by a distribution over the potential goals, following which the Bayes’ rule helps select the target image. In [248], RF is incorporated using a Bayesian classifier based re-ranking of the images after each feedback step. The main assumption used here is that the features of the positive examples, which potentially reside in the same semantic class, are all generated by an underlying Gaussian density. The RF approach in [267] is based on the intuition that the system’s belief at a particular time about the user’s intent is a *prior*, while the following user feedback is *new* information. Together, they help compute the new belief about the intent, using the Bayes’ rule, which becomes the prior for the next feedback round.

Region-based Methods

With increased popularity of region-based image retrieval [31, 278, 145], attempts have been made to incorporate the *region* factor into RF. In [132], two different RF scenarios are considered, and retrieval is tailored to support each of them through query point modification and SVM-based classification respectively. In this feedback process, the region importance (RI) for each segmented region is learned, for successively better retrieval. This core idea, that of integrating region-based retrieval with relevance feedback, has been further detailed for the two RF scenarios in [133].

Other Advancements

Besides the set of methods grouped together, there have been a number of isolated advancements covering various aspects of RF. For example, methods for performing RF using visual as well as textual features (meta-data) in unified frameworks have been reported in [173, 319, 6, 134]. A tree-structured SOM has been used as an underlying technique for RF [148] in a CBIR system [149]. A well-known RF problem with query specification is that after each round of user interaction, the top query results need to be recomputed following some modification. A way to speed up this *nearest-neighbor* search is proposed in [294]. The use of RF for helping capture the relationship between low-level features and high-level semantics, a fundamental problem in image retrieval, has been attempted using logs of user feedbacks, in [108].

Discussion

Relevance feedback provides a compromise between a fully automated, unsupervised system and one based on the subjective user needs. While query refinement is an attractive proposition when it comes to a very diverse user base, there is also the question of how well the feedbacks can be utilized for refinement. Whereas a user would prefer shorter feedback sessions, there is an issue as to how much feedback is enough for the system to learn the user needs. One issue which has been largely ignored in past RF research is that the user's needs might evolve over the feedback steps, making the assumption of a fixed target weaker. New approaches such as [123, 79] have started incorporating this aspect of the user's mind in the RF process.

Relevance feedback was introduced into image retrieval at the fag end of the previous decade (Sec. 2.1.1). Today, it is a more mature field, spanning many different sub-topics and addressing a number of practical concerns keeping in mind the user in the loop. While this has happened, one issue is that we do not see many real-world implementations of the relevance feedback technology either in the image or in the text retrieval domain. This is potentially due to the feedback process that the users must go through, that tests the user's patience. New ideas such as memory retrieval, that actually provide the user with benefits in the feedback

process, may possibly be one answer to popularizing RF. The future of this field clearly lies in its practical applicability, focusing on how the user can be made to go through least effort to convey the desired semantics. The breaking points of the utility derived out of this process, at which the user runs out of patience and at which she is satisfied with the response, must be studied for better system design.

2.2.5 Multimodal Fusion and Retrieval

Media relevant to the broad area of multimedia retrieval and annotation includes, but is not limited to, images, text, free-text (unstructured, e.g., paragraphs), graphics, video, and any conceivable combination of them. Thus far, we have encountered a multitude of techniques for modeling and retrieval of images, and text associated with those images. While not covered here, the reader may be aware of equally broad spectrums of techniques for text, video, music, and speech retrieval. In many cases, these independent, media-specific methods do not suffice to satiate the needs of users who are seeking what they can best describe only by a combination of media. Therein lies the need for multimodal fusion as a technique for satisfying such user queries. We consider this as one of the ‘core’ techniques because in principal, it is distinct from any of the methods we have discussed so far. Even with very good retrieval algorithms available independently for two different media, effectively combining them for multimodal retrieval may be far from trivial. Research in fusion learning for multimodal queries therefore attempts to learn optimal combination strategies and models.

Fortunately (for researchers) or unfortunately (for users), precious little multimodal fusion has been attempted in the context of image retrieval and annotation. This opens avenues for exploring novel user interfaces, querying models, and result visualization techniques pertinent to image retrieval, in combination with other media. Having said that, we must point out that multimodal fusion has indeed been attempted in the more obvious problem settings within *video retrieval*. With this field as an example, we briefly expose readers to multimodal fusion, in the hope that it motivates image retrieval research that takes advantage of these techniques. We believe that the need for multimodal retrieval in relation to images will soon grow in stature.

When video data comes with closed-captions and/or associated audio track, these can prove to be useful meta-data for retrieval as well. One of the key problems faced in video retrieval research is therefore combination or fusion of responses from these multiple modalities. It has been observed and reported that multimodal fusion almost always enhances retrieval performance for video [111]. Usually, fusion involves learning some kind of combination rules across multiple decision streams (ranked lists or classifier response) using a certain amount of data with ground truth as validation set. This is also referred to as late fusion. Alternative approaches to fusion involve classifier re-training. In [296], multimodal fusion has been treated as a two-step problem. The first step involves finding statistically independent modalities, followed by super-kernel fusion to determine their optimal combination. Fusion approaches have been found to be beneficial for important video applications such as detection of documentary scene changes [271] and story segmentation [304]. Fusion learning has been found to outperform naive fusion approaches as well as the oracle (best performer) for TRECVID 2005 query retrieval task. [137].

Discussion

Fusion learning is an off-line process while fusion application at real-time is computationally inexpensive. Hence multimodal fusion is an excellent method to boost retrieval performance at real-time. However, special care needs to be taken to ensure that the fusion rules do not overfit the validation set used for learning them. Usually, data resampling techniques such as bagging are found to help avoid overfitting to some extent. Fusion techniques can also be used to leverage classifiers built for numerous concepts with possible semantic coherence, whether the underlying data is image or video.

Fusion for image retrieval is a fairly novel area, with very little achieved in the early ages. The ideas of fusion go hand in hand with practical, viable, system development, which is critical for the future of image retrieval research. We live in a truly multi-media world, and we as humans always take the benefit of each media for sensory interpretation (see, hear, smell, taste, touch). There is no reason why advantage of all available media (images, video, audio, text) should not be taken for building useful systems. The future lies in harnessing as many channels

of information as possible, and fusing them in smart, practical ways to solve real problems. Principled approaches to fusion, particularly probabilistic ones, can also help provide performance guarantees, which in turn convert to quality standards for public-domain systems.

2.3 Offshoots: Problems of the New Age

Smeulders et al. [242] surveyed CBIR at the end of what they referred to as early years. The field was presented as a natural successor to certain existing disciplines such as computer vision, information retrieval and machine learning. However, in the last few years, CBIR has evolved and emerged as a mature research effort in its own right. A significant section of the research community is now shifting attention to certain problems which are peripheral, yet of immense significance to image retrieval systems, directly or indirectly. Moreover, newly discovered problems are being solved with tools that were intended for image retrieval. In this section, we discuss some such directions. Note that much of these peripheral ideas are in their infancy, and have likelihood of breaking into adulthood if sufficiently nurtured by the relevant research communities. Owing to the exploratory nature of the current approaches to these problems, a discussion on where these sub-fields are heading and what opportunities lie ahead in the future for innovation is necessary.

2.3.1 Automatic Annotation

While at the problem of understanding picture content, it was soon learned that in principle, associating those pictures with textual descriptions was only one step ahead. This led to the formulation of a new but closely associated problem called *automatic image annotation*, often referred to as *auto-annotation* or *linguistic indexing*. The primary purpose of a practical content-based image retrieval system is to discover images pertaining to a given concept in the absence of reliable meta-data. All attempts at automated concept discovery, annotation, or linguistic indexing essentially adhere to that objective. Annotation can facilitate image search through the use of text. If the resultant automated mapping between images and words can be trusted, text-based image searching can be semantically more mean-

ingful than search in the absence of any text. Here we discuss two different schools of thought which have been used to address this problem.

2.3.1.1 Joint Word-Picture Modeling Approach

Many of the approaches to image annotation have been inspired by research in the text domain. Ideas from text modeling have been successfully imported to jointly model textual and visual data. In [74], the problem of annotation is treated as a *translation* from a set of image segments to a set of words, in a way analogous to linguistic translation. A multi-modal extension of a well known hierarchical text model is proposed. Each word, describing a picture, is believed to have been generated by a node in a hierarchical concept tree. This assumption is coherent with the hierarchical model for nouns and verbs adopted by Wordnet [190]. This *translation model* is extended [131] to eliminate uncorrelated words from among those generated, making use of the Wordnet ontology. In [15], Latent Dirichlet Allocation (LDA) is proposed for modeling associations between words and pictures.

In all such approaches, images are typically represented by properties of each of their segments or *blobs*. Once all the pictures have been segmented, quantization can be used to obtain a finite vocabulary of blobs. Thus pictures under such models are treated as bags of words and blobs, each of which are assumed to have been generated by *aspects*. Aspects are hidden variables which spawn a multivariate distribution over blobs and a multinomial distribution over words. Once the joint word-blob probabilities have been learned, the annotation problem for a given image is reduced to a likelihood problem relating blobs and words. The spatial relationships between blobs is not directly captured by the model. However, this is expected to be implicitly modeled in the generative distribution. Most of these techniques rely on precise segmentation, which is still challenging. Despite the limitations, such modeling approaches remain popular.

Cross-Media relevance models have been used for image annotation in [127, 151]. A closely related approach involves coherent language models, which exploits word-to-word correlations to strengthen annotation decisions [129]. All the annotation strategies discussed so far model visual and textual features separately prior to association. A departure from this trend is seen in [193], where probabilistic latent semantic analysis (PLSA) is used on uniform vectorized data

consisting of both visual features and textual annotations. This model is extended to a *nonlinear* latent semantic analysis for image annotation in [171].

2.3.1.2 Supervised Categorization Approach

An alternative approach is to treat image annotation as a supervised categorization problem. Concept detection through supervised classification, involving simple concepts such as city, landscape, and sunset is achieved with high accuracy in [264]. More recently, image annotation using a novel structure-composition model, and a WordNet-based word saliency measure has been proposed in [55]. One of the earliest attempts at image annotation can be found in [163]. The system, ALIP (Automatic Linguistic Indexing of Pictures) uses a 2-D multi-resolution hidden Markov models based approach to capture inter-scale and intra-scale spatial dependencies of image features of given semantic categories. Models for individual categories are learned independently and stored. The annotation step involves calculating likelihoods of the query image given each learned model/category, and choosing annotations with bias toward statistically salient words corresponding to the most likely categories. A real time image annotation system ALIPR (Automatic Linguistic Indexing of Pictures - Real Time) has been recently proposed in [166]. ALIPR inherits its high level learning architecture from ALIP. However, the modeling approach is simpler, hence leading to real-time computations of statistical likelihoods. Being the first real time image annotation engine, ALIPR has generated considerable interest for real-world applications [3].

Learning concepts from user's feedback in a dynamic image database using Gaussian mixture models is discussed in [70]. An approach to *soft* annotation, using Bayes Point machines, to give images a confidence level for each trained semantic label is explored in [34]. This vector of confidence labels can be exploited to rank images for keyword search. A confidence based ensemble of SVM classifiers is used for annotation in [158]. Multiple instance learning based approaches have been proposed for semantic categorization of images [42] and to learn the correspondence between image regions and keywords [299]. Concept learning based on a fusion of complementary classification techniques with limited training samples is proposed in [202]. Annotating images in dynamic settings (e.g., Flickr), where images and tags arrive asynchronously over time, has been explored in [57].

Discussion: Automated annotation is widely recognized as an extremely difficult question. We humans segment objects better than machines, having learned to associate over a long period of time, through multiple viewpoints, and literally through a “streaming video” at all times, which partly accounts for our natural segmentation capability. The association of words and *blobs* becomes truly meaningful only when blobs isolate objects well. Moreover, how exactly our brain does this association is unclear. While Biology tries to answer this fundamental question, researchers in information retrieval tend to take a pragmatic stand in that they aim to build systems of practical significance. Ultimately, the desire is to be able to use keyword queries for all images regardless of any manual annotations that they may have. To this end, a recent attempt at bridging the retrieval-annotation gap has been made [55].

2.3.2 Inference of Image Aesthetics

Thus far, the focus of CBIR has been on semantics. There have been numerous discussion on the semantic gap. Imagine a situation where this gap has been bridged. This would mean, for example, finding all ‘dog’ pictures in response to a ‘dog’ query. In text-based search engines, a query containing ‘dog’ will yield millions of Web pages. The smart search engine will then try to analyze the query to rank the best matches higher. The rationale for doing so is that of predicting what is most desirable based on the query. What, in CBIR, is analogous to such ranking, given that a large subset of the images are determined to be semantically relevant? This question has been recently addressed in [56].

We conjecture that one way to distinguish among images of similar semantics is by their *quality*. Quality can be perceived at two levels, one involving concrete image parameters like size, aspect ratio and color depth, and the other involving higher-level perception, which we denote as *aesthetics*. While it is trivial to rank images based on the former, the differences may not be significant enough to use as ranking criteria. On the other hand, aesthetics is the kind of emotions a picture arouses in people. Given this vague definition, and the subjectivity associated with emotion, it is open to dispute how to aesthetically distinguish pictures. As discussed below, current attempts to model aesthetics have had limited success,

and the limitation arises primarily from the inability to extract information related to perceived emotions from pixel information. In a sense, this is analogous to the concept of semantic gap [242] in the domain of aesthetics inference, and probably a wider one at this moment. To formalize this analogy, we propose to define what we call the *aesthetic gap*, as follows:

The aesthetic gap is the lack of coincidence between the information that one can extract from low-level visual data (i.e., pixels in digital images) and the interpretation of emotions that the visual data may arouse in a particular user in a given situation.

Despite the challenge in dealing with this gap, in our opinion, modeling aesthetics of images is an important open problem that will only get more prominent as time passes. Given a feasible model, a new dimension to image understanding will be added, benefiting CBIR and allied communities.

Discussion: The question remains how this problem can be approached. Given the high subjectivity of aesthetics, it may help to re-define the goal as a model that can characterize aesthetics *in general*. One way to model aesthetics in general is to study photo rating trends in public photo-sharing communities such as [213], an approach that has been followed in [56]. The site supports peer-rating of photographs based on aesthetics. This has generated a large database of ratings corresponding to the over one million photographs hosted. A discussion on the significance of these ratings, and aesthetic quality in general, can be found in [214]. Another attempt [140] at distinguishing high-quality images from low-quality ones has found similar levels of success with data obtained from yet another peer-rated photo contest oriented Website [71]. The idea of learning to assess visual aesthetics from such training data has been further pursued for the purpose of selecting high-quality pictures and eliminating low-quality ones from image collections, in [62]. *One caveat:* Uncontrolled publicly collected data are naturally inclined to noise. When drawing conclusions about the data, this assumption must be kept in mind. Alternatively, ways to get around the noisy portions must be devised.

2.3.3 Web Image Search

The Web connects systems to systems, systems to people, and people with other people. Hosting a system on the Web is significantly different from hosting it in a private network or a single machine. What makes things different is that we can no longer make assumptions about the users, their understanding of the system, their way of interacting, their contributions to the system, and their expectations from the system. Moreover, Web-based systems must support the masses only as long as they are useful to them. Without support, there is no meaning to such a system. This makes the creation of Web-based CBIR systems more challenging than the core questions of CBIR, aggravated further by the fact that multimedia searching is typically more complex than generic searching [126]. Thankfully, the problem has recently received a lot of attention from the community, enough to have a survey dedicated specifically to it [142].

While we cannot make assumptions about generic Web-based CBIR systems, those designed keeping in mind specific communities can be done with some assumptions. Web-based CBIR services for copyright protection, tourism, entertainment, crime prevention, research, and education are some domain-specific possibilities, as reported in [142]. One of the key tasks of Web image retrieval is crawling images. A smart Web-crawler that attempts to associate captions with images to extract useful meta-data in the crawling process is reported in [220].

There have been many algorithms proposed for image search based on surrounding text, including those implemented in Google and Yahoo! image search. Here we discuss work that exploits image content in part or full for retrieval. One of the earlier systems for Web-based CBIR, *iFind*, incorporating relevance feedback was proposed in [307]. More recently, *Cortina*, a combined content and meta-data based image search engine is made public [218]. Other approaches to Web-based image retrieval include mutual reinforcement [281], bootstrapping for annotation propagation [81], and nonparametric density estimation with application to an art image collection [246]. Image grouping methods such as unsupervised clustering are extremely critical for heterogeneous repositories such as the Web (as discussed in Sec. 2.2.3), and this is explored in [280, 91, 24, 135]. More recently, rank fusion for Web image retrieval from multiple online picture forums has been proposed [308]. Innovative interface designs for Web image search have been explored in [301, 169].

The SIMPLIcity system [278] has been incorporated into popular Websites such as Airliners.net [2], Global Memory Net [95], and Terragalleria [251].

Discussion: The impact of CBIR can be best experienced through a Web-based image search service that gains popularity to the proportion of its text-based counterparts. Unfortunately, at the time of writing this survey, this goal is elusive. Having said that, the significant progress in CBIR for the Web raises hopes for such systems in the coming years.

2.3.4 Image-based Security

The interactions between CBIR and information security had been non-existent, until new perspectives emerged to strengthen the ties. Two such perspectives are human interactive proofs (HIPs), and the enforcement of copyright protection.

While on one hand, we are constantly pushing the frontiers of science to design intelligent systems that can imitate human capabilities, we cannot deny the inherent security risks associated with extremely smart computer programs. One such risk is when Websites or public servers are attacked by malicious programs that request service on massive scale. Programs can be written to automatically consume large amount of Web resources or bias results in on-line voting. The HIPs, also known as CAPTCHAs, are a savior in these situations. These are interfaces designed to differentiate between humans and automated programs, based on the response to posed questions. The most common CAPTCHAs use distorted text, as seen in public Websites such as Yahoo!, MSN, and PayPal. Recently, a number of OCR-based techniques have been proposed to break text-based CAPTCHAs [195]. This has paved the way for natural image based CAPTCHAs, owing to the fact that CBIR is generally considered a much more difficult problem than OCR. The first formalization of image based CAPTCHAs is found in [45], where pictures chosen at random are displayed and questions asked, e.g., what does the picture contain, which picture is the odd one out conceptually, etc. A problem with this approach is the possibility that CBIR and concept learning techniques such as [9, 163] can be used to attack image based CAPTCHAs. This will eventually lead to the same problem faced by text-based CAPTCHAs. To alleviate this problem, a CBIR system is used as a validation technique in order to distort images

before being presented to users [60]. The distortions are chosen such that probabilistically, CBIR systems find it difficult to grasp the image concepts and hence are unable to simulate human response.

The second issue is image copy protection and forgery detection. Photographs taken by one person and posted online are often copied and passed on as someone else's artistry. Logos and trademarks of well-established organizations have often been duplicated by lesser-known firms, with or without minor modification, and with a clear intention to mislead patrons. While plagiarism of this nature is a world-wide phenomenon today, protection of the relevant copyrights is a very challenging task. The use of CBIR to help identify and possibly enforce these copyrights is a relatively new field of study. In the case of exact copies, detecting them is trivial: extraction and comparison of a simple file signature is sufficient. However, when changes to the pictures or logos are made, image similarity measures such as those employed in CBIR are necessary. The changes could be one or more of down-sampling, lowering of color-depth, warping, shearing, cropping, de-colorizing, palette shifting, changing contrast/brightness, image stamping, etc. The problem then becomes one of *near-duplicate detection*, in which case the similarity measures must be robust to these changes. Interest point detectors for generating localized image descriptors robust to such changes have been used for near-duplicate detection in [138]. A part-based image similarity measure that is derived from the stochastic matching of Attributed Relational Graphs is exploited for near-duplicate detection in [305].

Discussion: Much of security research is on anticipation of possible attack strategies. While image-based CAPTCHA systems anticipate the use of CBIR for attacks, near-duplicate detectors anticipate possible image distortion methods a copyright infringer may employ. Whether CBIR proves useful to security is yet to be seen, but dabbling with problems of this nature certainly helps CBIR grow as a field. For example, as noted in [305], near-duplicate detection also finds application in weaving news stories across diverse video sources for news summarization. The generation of new ideas as offshoots, or in the process of solving other problems is the very essence of this section.

2.3.5 Innovation in Machine Learning

While more often than not machine learning has been used to help solve the fundamental problem of image retrieval, there are instances where new and generic machine learning and data mining techniques have been developed in attempts to serve this purpose. The correspondence-LDA [15] model, proposed for joint word-image modeling, has since been applied to problems in bioinformatics [314]. Probabilistic graphical models such as 2-D multiresolution hidden Markov models [163] and cross-media relevance models [127], though primarily used for image annotation applications, are contributions to machine learning research. Similarly, multiple instance learning research has benefited by work on image categorization [42]. Active learning using SVMs were proposed for relevance feedback [259] and helped popularize active learning in other domains as well.

Automatic learning of a similarity metric or distance from ground-truth data has been explored for various task such as clustering and classification. One way to achieve this is to learn a generalized Mahalanobis distance metric, such as those general-purpose methods proposed in [298, 8]. On the other hand, kernel-based learning of image similarity, using context information, with applications to image clustering was explored in [292]. This could potentially be used for more generic cases of metric learning given side-information. In the use of a Mahalanobis metric for distance computation, an implicit assumption is that the underlying data distribution is Gaussian, which may not always be appropriate. An important work uses a principled approach to determine appropriate similarity metrics based on the nature of underlying distributions, which is determined using ground-truth data [234]. In a subsequent work, a boosting approach to learning a *boosted distance* measure that is analogous to the weighted Euclidean norm, has been applied to stereo matching and video motion tracking [302] and classification/recognition tasks on popular datasets [5].

Discussion: When it comes to recognizing pictures, even humans undergo a learning process. So it is not surprising to see the synergy between machine learning and image retrieval, when it comes to training computers to do the same. In fact, the challenges associated with learning from images have actually helped push the scientific frontier in machine learning research in its own right.

Table 2.3. A qualitative requirement analysis of various CBIR offshoots.

Applications & Offshoots	<i>Similarity measure</i>	<i>User feed-back</i>	<i>Machine learning</i>	<i>Visualization</i>	<i>Scalability</i>
Automatic annotation	optional	optional	essential	optional	optional
Image-based CAPTCHA	essential	essential	optional	essential	essential
Visual aesthetics	optional	desirable	essential	desirable	optional
Web image search	essential	optional	optional	essential	essential

2.3.6 Epilogue

While Sec. 2.2 discussed techniques and real-world aspects of CBIR, in this section, we have described applications that employ those techniques. In Table 2.3 we present a qualitative requirement analysis of the various applications, involving a mapping from the *aspects* (techniques and features) to these applications. The entries are intended to be interpreted in the following manner:

- Essential - Aspects that are *required* in all scenarios.
- Optional - Aspects that *may/may not* be critical depending on the specifics.
- Desirable - Aspects that are *likely to add value* to the application in all cases.

The distinction between classifying an aspect as ‘optional’ or ‘desirable’ can be understood by the following examples. Scalability for automatic annotation is termed ‘optional’ here because such an application can serve two purposes: (1) to be able to quickly tag a large number of pictures in a short time, and (2) to be able to produce accurate and consistent tags to pictures or to refine existing noisy tags, perhaps as an off-line process. Because of the compromise made in these two goals, their scalability requirement may be different. As a second example, consider that in art image analysis, having an expert user to be involved in every step of the analysis is highly ‘desirable’, unlike in large scale image annotation, where a user validation at each step may be infeasible.

2.4 Summary

In this chapter, we have presented a comprehensive survey, highlighting current progress, emerging directions, and the spawning of new fields relevant to the young and exciting field of image retrieval. We have contrasted early years of image retrieval with the progress in the current decade, and conjectured specific future directions alongside. We believe that the field will experience a paradigm shift in the foreseeable future, with the focus being more on application-oriented, domain-specific work, generating considerable impact in day-to-day life.

In Appendix A, as part of an effort to understand the field of image retrieval better, we compiled research trends in image retrieval using Google Scholar's search tool, which presents citation counts as well. Graphs for publication and citation counts have been generated for (1) sub-fields of image retrieval, and (2) venues/journals relevant to image retrieval research. Further analysis has been made on the impact that image retrieval has had in merging interests among different fields of study, such as multimedia (MM), machine learning (ML), information retrieval (IR), computer vision (CV), and human-computer interaction (HCI). Firstly, the trends indicate that the field is extremely diverse, and can only grow to be more so in the future. Second, we note that image retrieval has likely caused a number of otherwise-unrelated fields of research to be brought close together. Third, interesting facts have emerged, such as: Most of the CV and AI work related to image retrieval have been published in information retrieval related venues and received high citations. At the same time, AI related work published in CV venues have generated considerable impact. At a higher level, the trends indicate that while systems, feature extraction, and relevance feedback have received a lot of attention, application-oriented aspects such as interface, visualization, scalability, and evaluation have traditionally received lesser consideration. We believe that these aspects will enjoy growing importance over time, as the focus moves toward real-world implementation.

The quality (resolution and color depth), nature (dimensionality), and throughput (rate of generation) of images acquired have all been on an upward growth path in the recent times. With the advent of very large scale images (e.g., Google and Yahoo! aerial maps), biomedical and astronomical imagery which are typically of

high resolution/dimension and are often captured at high throughput, pose yet new challenges to image retrieval research. A long term goal of research should therefore also include the ability to make high-resolution, high-dimension, and high-throughput images searchable by content. The future of image retrieval depends a lot on the collective focus and overall progress in each aspect of image retrieval, and how much the average individual stands to benefit from it.

Bridging the Semantic Gap: Improving Image Search via Automatic Annotation

Quick ways to capture pictures, cheap devices to store them, and convenient mechanisms for sharing them are all part and parcel of our daily lives today. There is indeed a very large amount of pictures to deal with. Naturally, everyone will benefit if there exist smart programs to manage picture collections, tag them automatically, and make them searchable by keywords. To satisfy such needs, the multimedia, information retrieval, and computer vision communities have, time and again, attempted automated image annotation, as we have witnessed in the recent past [9, 34, 163, 193]. While many interesting ideas have emerged, we have not seen much attention paid to the direct use of automatic annotation for image search. Usually, it is assumed that good annotation implies quality image search. Moreover, most past approaches are too slow for the massive picture collections of today to be of practical use. Much remains to be achieved.

The problem would not be interesting if all pictures came with tags, which in turn were reliable. Unfortunately, for today's picture collections such as Yahoo! Flickr, this is seldom the case. These collections are characterized by their mammoth volumes, lack of reliable tags, and the diverse spectrum of topics they cover. In Web image search systems such as those of Yahoo! and Google, surrounding text form the basis of keyword search, which come with their own problems.

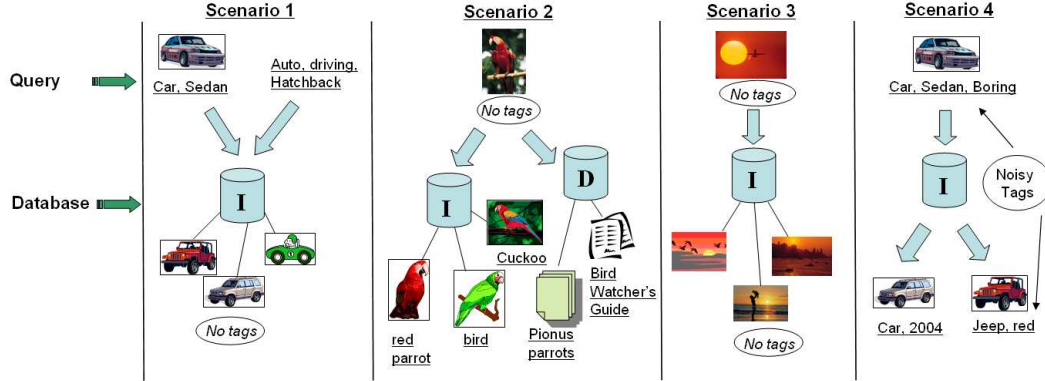


Figure 3.1. Four common scenarios for real-world image retrieval.

In this chapter, we discuss our attempt to build an image search system on the basis of automatic tagging. Our goal is to treat automatic annotation as a means of satisfactory image search. We look at realistic scenarios that arise in image search, and propose a framework that can handle them through a unified approach. To achieve this, we look at how pictures can be accurately and rapidly placed into a large number of categories, how the categorization can be used effectively for automatic annotation, and how these annotations can be harnessed for image search. For this, we use novel statistical models and the WordNet ontology [190], and use state-of-the-art content based image retrieval (CBIR) methods [59, 242, 278] for comparison.

In summary, our contributions are: **(1)** We propose a novel structure composition (S-C) model based on Beta distributions, aimed at capturing the spatial structure and composition of generic picture categories. Empirically, we find this model to be helpful in characterizing challenging picture categories. **(2)** We combine this S-C model with a color-texture Gaussian mixture model to generate rapid picture categorization involving upto 600 different categories, which also improves over best reported accuracy results on this problem. **(3)** We propose a novel tagging mechanism which considers (a) the evidence provided by the categorization results for potential tags, (b) the chance occurrence of these tags, and (c) the coherence each tag has with the remaining pool of candidate tags, as evidenced by WordNet. **(4)** To the best of our knowledge, this is the first formal work on using annotation directly for image search. Our method significantly outperforms competing strategies in all cases, producing some unexpected results as well.

3.0.1 Bridging the Gap

Our motivation to ‘bridge’ the annotation-retrieval gap is driven by a desire to effectively handle challenging cases of image search in a unified manner. These cases are schematically presented in Fig. 3.1, and elucidated below.

- **Scenario 1:** Either a tagged picture or a set of keywords is used as query. Problem arises when part or whole of the image database (e.g., Web images) is not tagged, making this portion inaccessible through text queries. We study how our annotation-driven image search approach performs in first annotating the untagged pictures, and then performing multiple keyword queries on the partially tagged picture collection.
- **Scenario 2:** An untagged image is used as query, with the desire to find semantically related pictures or documents from a tagged database or the Web. We look at how our approach performs in first tagging the query picture and then doing retrieval.
- **Scenario 3:** The query image as well as part/whole of the image database are untagged. This is the case that best motivates CBIR, since the only available information is visual content. We study the effectiveness of our approach in tagging the query image and the database, and subsequently performing retrieval.
- **Scenario 4:** A tagged query image is used to search on a tagged image database. The problem is that these tags may be noisy and unreliable, as is common in user-driven picture tagging portals. We study how our approach helps improve tagging by re-annotation, and subsequently performs retrieval.

In each case, we look at reasonable and practical alternative strategies for search, with the help of a state-of-the-art CBIR system. For scenario 4, we are also interested in analyzing the extent to which our approach helps improve annotation under varying levels of noise. Additional goals include the ability to generate precise annotations of pictures in near-realtime. While most previous annotation systems assess performance based on the quality of annotation alone, this is only a part of our goal. For us, the main challenge is to have the annotations help

generate meaningful retrieval. To this end, we develop our approach as follows. We first build a near-realtime categorization algorithm (~ 11 sec/image) capable of producing accurate results. We then proceed to generate annotation on the basis of categorization, ensuring high precision and recall. With this annotation system in place, we assess its performance as a means of image search under the preceding scenarios.

3.1 Model-based Categorization

We employ generative statistical models for accurate, near-realtime categorization of generic images. This implies training independent statistical models for each image category using a small set of training images. Assignment of category labels to new pictures can then be done by a smart utilization of the likelihoods over all models. In our system, we use two generative models (per image category) to provide ‘evidence’ for categorization from two different aspects of the images. We generate final categorization by combining these evidences.

Formally, let there be a feature extraction process or function \mathfrak{S} that takes in an image I and returns a collection of D feature vectors, each of dimension V , i.e., $\mathfrak{S}(I)$ has dimension $D \times V$, D varying with each image. Given C categories and N training images per category, each of C models $M_k, k = 1, \dots, C$ with parameters θ_k are built using training images $I_i^k, i = 1, \dots, N$, by some parameter estimation technique. Suppose the collection of feature vectors, when treated as random variables $\{X_1, \dots, X_D\}$, can be assumed conditionally independent given model parameters θ_k . For a test image I , given that $\mathfrak{S}(I) = \{x_1, \dots, x_D\}$ is extracted, the log-likelihood of I being generated by model M_k is

$$\ell_1(I|M_k) = \log p(x_1, \dots, x_D|\theta_k) = \sum_{d=1}^D \log p(x_d|\theta_k) . \quad (3.1)$$

Assuming equal category priors, a straightforward way to assign a category label y to I would be to have

$$y(I) = \arg \max_k \ell_1(I|M_k).$$

Now consider that we have another set of C generative models trained on a different set of image features and with a different underlying statistical distribution. Suppose the log-likelihoods generated by these models for the same image I are $\{\ell_2(I|M_1), \dots, \ell_2(I|M_C)\}$. Each category of generic images is typically described by multiple tags (e.g., tiger, forest, and animal for a tiger category). Given a large number of categories, many of them having semantic/visual overlaps (e.g., night and sky, or people and parade), the top ranked category alone from either model may not be accurate. One way to utilize both models in the categorization process is to treat them as two experts independently examining the images from two different perspectives, and reporting their findings. The findings are not limited to the two most likely categories for each model, but rather the entire set of likelihoods for each category, given the image. Hence, an appropriate model combination strategy $\rho(\cdot)$ may be used to predict the image categories in a more general manner:

$$y(I) = \rho\left(\ell_1(I|M_1), \dots, \ell_1(I|M_C), \ell_2(I|M_1), \dots, \ell_2(I|M_C)\right). \quad (3.2)$$

For a large number of generic image categories, building a robust classifier is an uphill task. Feature extraction is extremely critical here, since it must have the discriminative power to distinguish between a broad range of image categories, no matter what machine learning technique is used. We base our models on the following intuitions: (1) For certain categories such as sky, marketplace, ocean, forests, Hawaii, or those with dominant background colors such as paintings, color and texture features may be sufficient to characterize them. In fact, a structure or composition for these categories may be too hard to generalize. (2) On the other hand, categories such as fruits, waterfall, mountains, lions, and birds may not have dominating color or texture but often have an overall structure or composition which helps us identify them despite heavily varying color distributions. In [163], the authors use 2-D multi-resolution hidden Markov models (2-D MHMMs) to capture the inter-scale and intra-scale dependence of block-based color and texture based features, thus characterizing the composition/structure of image categories. Problems with this approach are that the dependence modeling is over relatively local image regions, the parameter estimation algorithm involves numerical approximation, and the overall categorization process is slow. While our

work is inspired by similar motivations, we aim at near-realtime and more accurate categorization. We thus build two models to capture different visual aspects, (1) a structure-composition model that uses Beta distributions to capture color interactions in a very flexible but principled manner, and (2) a Gaussian mixture model in the joint color-texture feature space. We now elaborate on each model.

3.1.1 Structure-Composition (S-C) Models

The idea of building such a feature arose from a desire to represent how the colors interact with each other in certain picture categories. The average beach picture could be described by a set of relationships between different colored regions, e.g., orange (sun) completely inside light-blue (sky), light-blue sharing a long border with dark-blue (ocean), dark-blue sharing a long border with brown (sand) etc. For tiger images, this description could be that of yellow and black regions sharing very similar borders with each other (stripes) and rest of the colors interacting without much pattern or motif. Very coarse texture patterns such as pictures of beads of different colors (not captured well by color distribution or localized texture features such as wavelets) could be described as any color (bead) surrounding any other color (bead), some color (background) completely containing most colors (beads), and so on. This idea led to a principled statistical formulation of rotational and scale invariant structure-composition (S-C) models.

Given the set of all training images across categories, we take every pixel from each image, converted to the perceptually uniform LUV color space. We thus have a very large population of LUV vectors in the \mathbb{R}^3 space representing the color distribution within the entire training set. The K -means geometric clustering with uniform initialization is performed on a manageable random sub-sample to obtain a set of S cluster centroids $\{T_1, \dots, T_S\}$, e.g., shades of red, yellow etc. We then perform a nearest-neighbor based segmentation on each training image I by assigning a cluster label to each pixel (x, y) as follows:

$$J(x, y) = \arg \min_i |I_{luv}(x, y) - T_i| . \quad (3.3)$$

In essence, we have quantized the color space for the entire set of training images to obtain a small set of representative colors. This helps to build a uniform model

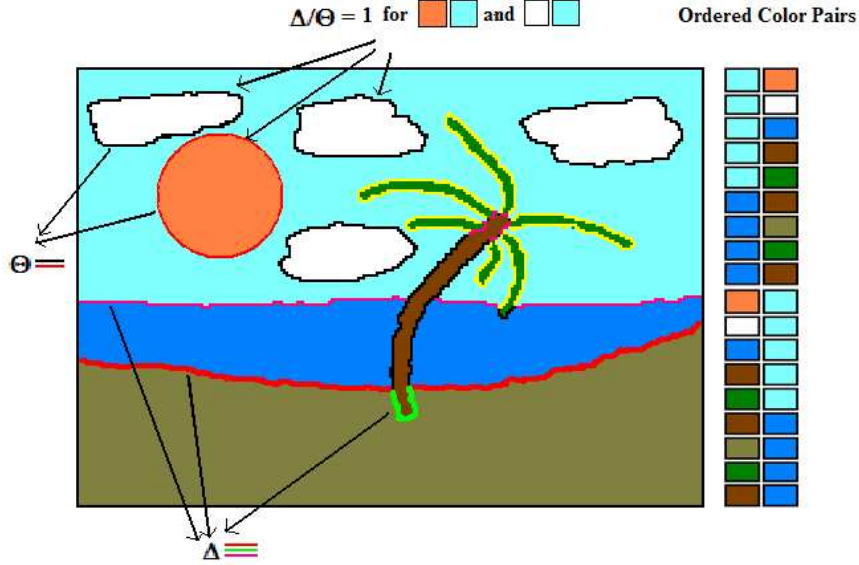


Figure 3.2. The idea behind the S-C model, shown here on a *toy* image. We denote the perimeters of each segment by Θ and the border lengths between pairs of segments by Δ . Intuitively, Δ/Θ ratios for the *orange*, *light-blue* (sun and sky) and *white*, *light-blue* (clouds and sky) pairs equals 1 since sun and cloud perimeters coincide with their borders shared with sky. In general, the ratio has low value when segments are barely touching, and near 1 when a segment is completely contained within another segment.

representation for all image categories. To uniquely identify each segment in the image, we perform a two-pass 8-connected component labeling on J . The image J now has P connected components or segments $\{s_1, \dots, s_P\}$. The many-to-one mapping from a segment s_i to a color T_j is stored and denoted by the function $G(s_i)$. Let χ_i be the set of neighboring segments to segment s_i . Neighborhood in this sense implies that for two segments s_i and s_j , there is at least one pixel in each of s_i and s_j that is 8-connected. We wish to characterize the interaction of colors by modeling how each color shares (if at all) boundaries with every other color. For example, a *red-orange* interaction (in the quantized color space) for a given image category will be modelled by how the boundaries are shared between every *red* segment with every other *orange* segment for each training image, and vice-versa (See Fig.3.2). More formally, let $(x, y) \oplus B$ indicate that pixel (x, y) in J is 8-connected to segment B , and let $N(x, y)$ denote the set of its 8 neighboring points (not segments). Now we define a function $\Delta(s_i, s_j)$ which denotes the length

of the shared border between a segment s_i and its neighboring segment s_j , and a function $\Theta(s_i)$ which defines the total length of the perimeter of segment s_i ,

$$\Delta(s_i, s_j) = \sum_{(x,y) \in s_i} In((x, y) \oplus s_j), s_j \in \chi_i, \text{ and} \quad (3.4)$$

$$\Theta(s_i) = \sum_{(x,y) \in s_i} In(\mathbb{N}(x, y) \not\subset s_i), \quad (3.5)$$

where $In(\cdot)$ is the indicator function. By this definition of \mathbb{N} , inner borders (e.g. holes in donut shapes) and image boundaries are considered part of segment perimeters. We want to model the Δ/Θ ratios for each color pair by some statistical distribution. For random variables bounded in the $[0, 1]$ range, the *Beta distribution* is a flexible continuous distribution defined in the same range, with *shape* parameters (α, β) . The Beta density function is defined as

$$f(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1}, \text{ given} \quad (3.6)$$

$$B(\alpha, \beta) = \int_0^1 v^{\alpha-1} (1-v)^{\beta-1} dv = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}, \quad (3.7)$$

where $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ is the well-known *Gamma function*. Our goal is to build models for each category such that they consist of a set of Beta distributions for every color pair. For each category, and for every color pair, we find each instance in the N training images in which segments of that color pair share a common border. Let the number of such instances be η . We then compute the corresponding set of Δ/Θ ratios and estimate a Beta distribution (i.e., parameters α and β) using these values for that color pair. The overall structure-composition model for a given category k thus has the following form:

k	1	2	...	S
1	n/a	α, β, η	...	α, β, η
2	α, β, η	n/a
...	α, β, η
S	α, β, η	...	α, β, η	n/a

Note that it is not possible to have segments with the same color as neighbors. Thus

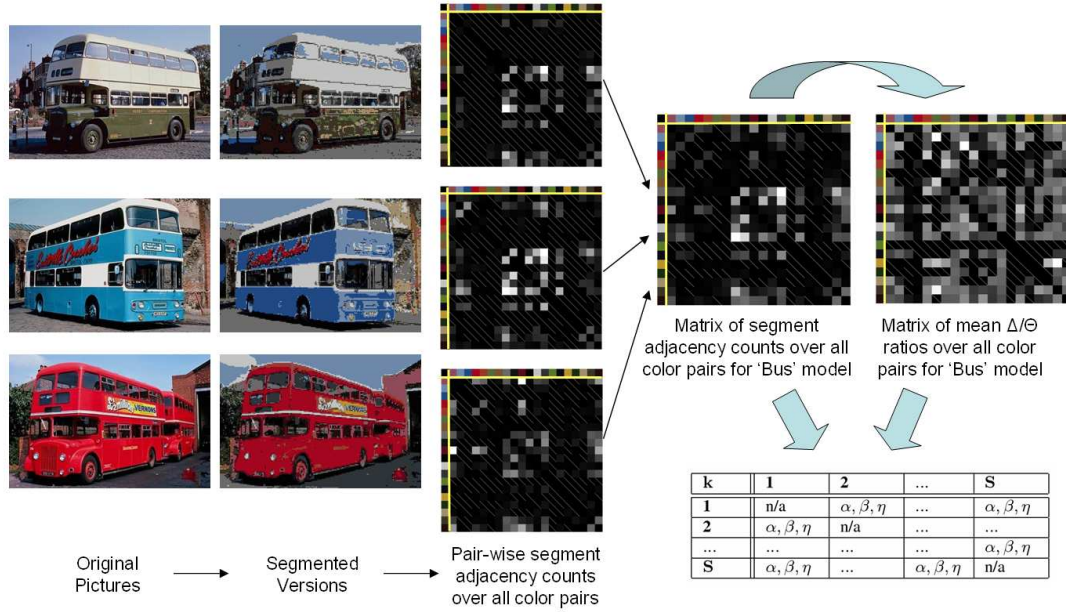


Figure 3.3. Steps toward generating the structure-composition model. On the left, we have three training pictures from the ‘bus’ category, their segmented forms, and a matrix representation of their segment adjacency counts. On the right, the corresponding matrix representations over all three training pictures are shown. Finally, these matrices are combine to produce the structure-composition model, shown here schematically as a matrix of Beta parameters and counts.

parameters of the form $\alpha(i, i), \beta(i, i)$ or $\eta(i, i)$ do not exist, i.e., same color pair entries in the model are ignored, denoted here by ‘n/a’. Note also that the matrix is *not* symmetric, which means the color pairs are ordered, i.e., we treat yellow-orange and orange-yellow color interactions differentially, for example. Further, the number of samples η used to estimate the α and β are also stored with the corresponding entries as part of the model. The reason for doing so will be evident shortly.

For the estimation of α and β , a moment matching method is employed for its computational efficiency. Given a set of $\eta(i, j)$ Δ/Θ samples for a given color pair (i, j) , having values $\{x_1, \dots, x_{\eta(i, j)}\}$, the parameters are estimated as follows:

$$\alpha(i, j) = \bar{x} \left(\left(\frac{\bar{x}(1-\bar{x})}{s^2} \right) - 1 \right)$$

$$\beta(i, j) = (1 - \bar{x}) \left(\left(\frac{\bar{x}(1-\bar{x})}{s^2} \right) - 1 \right)$$

Here $\bar{x} = \frac{1}{\eta(i, j)} \sum_{k=1}^{\eta(i, j)} x_k$, $s^2 = \frac{1}{\eta(i, j)} \sum_{k=1}^{\eta(i, j)} (x_k - \bar{x})^2$. There are two issues with

estimation in this manner, (1) the estimates are not defined for $\eta \leq 1$, and (2) for low values of η , estimation is poor. Yet, it is realistic for some categories to have few or no training samples for a given color pair, where estimation will be either poor or impossible respectively. But, low occurrence of neighboring segments of certain color pairs in the training set may or may not mean they will not occur in test images. To be safe, instead of penalizing the occurrence such color pairs in test images, we treat them as “unknown”. To achieve this, we estimate parameters α'_k and β'_k for the distribution of all Δ/Θ ratios across all color pairs within a given category k of training images, and store them in the models as prior distributions. The overall process of estimating S-C models, along with their representation, can be seen in Fig. 3.3.

During categorization, we segment a test image in exactly the same way we performed the training. With the segmented image, we obtain the set of color interactions characterized by Δ/Θ values for each segment boundary. For a given sample $x = \Delta/\Theta$ coming from color pair (i, j) in the test image, we compute its probability of belonging to a category k . Denoting the stored parameters for the color pair (i, j) for model k as α, β and η , we have

$$P_{sc}(x|k) = \begin{cases} f(x|\alpha'_k, \beta'_k), & \eta \leq 1 \\ \frac{\eta}{\eta+1}f(x|\alpha, \beta) + \frac{1}{\eta+1}f(x|\alpha'_k, \beta'_k), & \eta > 1 \end{cases}$$

where P_{sc} is the conditional p.d.f. for the S-C model. What we have here is typically done in statistics when the amount of confidence in some estimate is low. A weighted probability is computed instead of the original one, weights varying with the number of samples used for estimation. When η is large, $\eta/(\eta+1) \rightarrow 1$ and hence the distribution for that specific color pair exclusively determines the probability. When η is small, $1/(\eta+1) > 0$ in which case the probability from the prior distribution is given considerable importance. This somewhat solves both the problems of undefined and poor parameter estimates. It also justifies the need for storing the number of samples η as part of the models.

The S-C model is estimated for each training category $k \in \{1..C\}$. Each model consists of $3S(S-1)$ parameters $\{\alpha_k(i, j), \beta_k(i, j), \eta_k(i, j)\}, i \in \{1..S\}, j \in \{1..S\}, i \neq j$, and parameters for the prior distribution, α'_k and β'_k as explained. This set of parameters constitute θ_k , the parameter set for category k . We build and

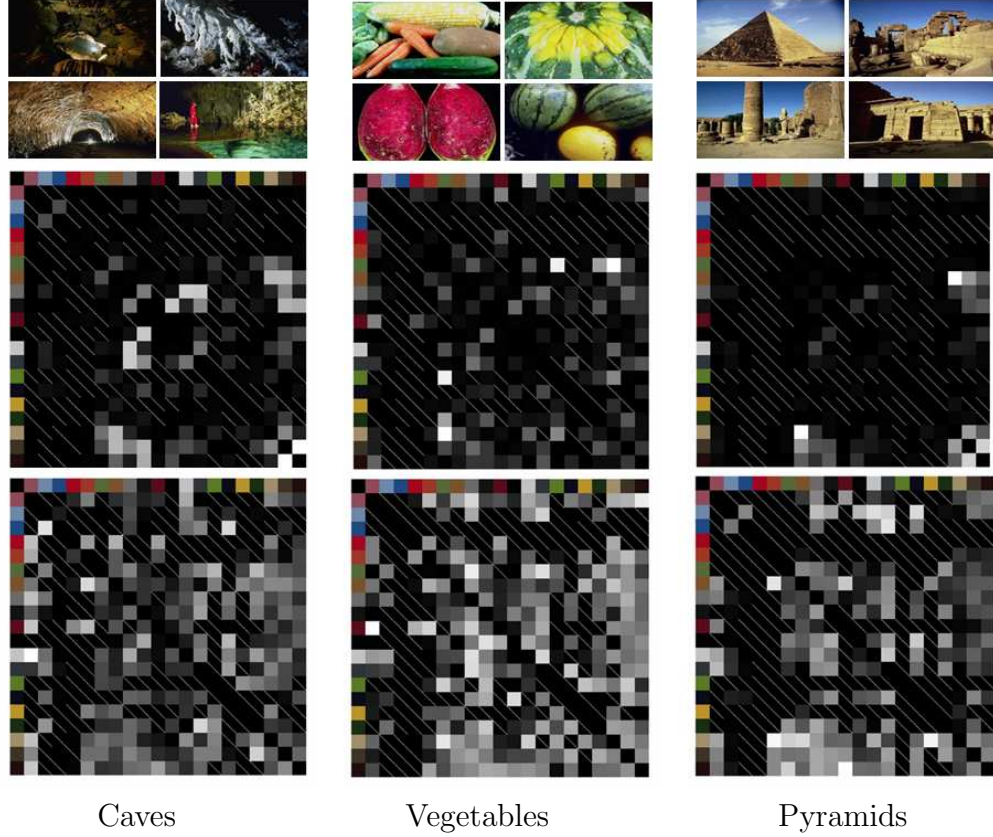


Figure 3.4. Sample categories and corresponding structure-composition model representations. *Top:* Sample training pictures. *Middle:* Matrices of segment adjacency counts. *Bottom:* Matrices of mean Δ/Θ ratios. Brightness levels represent relative magnitude of values.

store such models for every category. In Fig. 3.4, we show simple representations of the learned models for three such picture categories. The feature extraction process $\mathfrak{S}(I)$ generates the Δ/Θ ratios and the corresponding color-pairs for a given image I . We thus obtain a collection of D (varying with each image) feature vectors $\{x_1, \dots, x_D\}$, where each $x_d = \{\Delta_d/\Theta_d, i_d, j_d\}$. We assume conditional independence of each x_d . Hence, using equation (3.1), we have

$$\ell_{sc}(I|M_k) = \sum_{d=1}^D \log P_{sc}(\Delta_d/\Theta_d | \theta_k(i_d, j_d)) . \quad (3.8)$$

3.1.1.1 Fast Computation of S-C model Features

We wish to have a low complexity algorithm to compute the Δ/Θ ratios for a given image (training or testing). As discussed, these ratios can be computed in a naive manner as follows: **(1)** Segment the image by nearest neighbor assignments followed by connected component labeling. **(2)** For each segment, compute its perimeter (Θ), and length of border (Δ) shared with each neighboring segment. **(3)** Compute the Δ/Θ ratios and return them (along with the corresponding color pairs) for modeling or testing, whichever the case. This algorithm can be sped as follows. Denote the segment identity associated with each pixel (x, y) by $s(x, y)$. Each (x, y) is either (1) an interior pixel, not bordering any segment or the image boundary, (2) a pixel that is either bordering two or more segments, or is part of the image boundary, or (3) a pixel that has no neighboring segments but is part of the image boundary. Pixels in (1) do not contribute to the computation of Δ or Θ and hence can be ignored. Pixels in (2) are both part of the perimeter of segment $s(x, y)$ and the borders between $s(x, y)$ and each neighboring segment s_k (i.e., $(i, j) \oplus s_k$). Pixels in (3) are only part of the perimeter of $s(i, j)$. Based on this, a *single-pass algorithm* for computing the S-C feature vector $\{x_1, \dots, x_D\}$ of an image I is presented in Fig. 3.5.

The set of ordered triplets $[x_d, G(i), G(j)]$ can now be used to build Beta distributions with parameters $\alpha(G(i), G(j))$ and $\beta(G(i), G(j))$, provided no. of samples $\eta(G(i), G(j)) > 1$. Besides the two-pass connected component labeling, only a single scanning of the image is required to compute these features. It is not hard to see that this algorithm can be embedded into the two-pass connected component labeling algorithm to further improve speed. Note that though the asymptotic order of complexity remains the same, the improved computational efficiency becomes significant as the image database size increases.

3.1.2 Color-Texture (C-T) Models

Many image categories, especially those that do not contain specific objects, can be best described by their color and texture distributions. There may not even exist a well-defined structure per se, for high-level categories such as China and Europe, but the overall ambience formed the colors seen in these images often help identify

Single-pass Computation of S-C Model Features

```

Pair(1..P, 1..P)  $\leftarrow$  0      [P = No. of segments]
Perim(1..P)  $\leftarrow$  0
for each pixel (x, y) in I
    k  $\leftarrow$  0; Z  $\leftarrow$   $\phi$ 
    for each 8-neighbor (x', y')  $\in$  D(x, y)
        if (x', y') is inside image boundary
            if s(x', y')  $\neq$  s(x, y) and s(x', y') is unique
                Z  $\leftarrow$  Z  $\cup$  s(x', y')
                k  $\leftarrow$  1
            else
                k  $\leftarrow$  1
    for each s'  $\in$  Z
        Pair(s(x, y), s')  $\leftarrow$  Pair(s(x, y), s') + 1
    if k = 1
        Perim(s(x, y))  $\leftarrow$  Perim(s(x, y)) + 1
    [Now Generate  $\Delta/\Theta$  ratios :  $\mathfrak{S}(I) = \{x_1, \dots, x_D\}$ ]
    d  $\leftarrow$  0
    for i  $\leftarrow$  1 to P
        for j  $\leftarrow$  1 to P
            if Pair(i, j) > 0    [(i, j) segments shared border]
                d  $\leftarrow$  d + 1
                 $\Delta_d$   $\leftarrow$  Pair(i, j);  $\Theta_d$   $\leftarrow$  Perim(i)
                 $x_d$   $\leftarrow$   $\Delta_d/\Theta_d$ 
                return [ $x_d$ , G(i), G(j)]
    [G(.) - maps segment to color]

```

Figure 3.5. Algorithm for computing S-C features.

them. A mixture of multivariate Gaussians is used to model the joint color-texture feature space for a given category. The motivation is simple; in many cases, two or more representative regions in the color/texture feature space can represent the image category best. For example, beach pictures typically have one or more yellow areas (sand), a blue non-textured area (sky), and a blue textured region (sea). Gaussian mixture models (GMMs) are well-studied, with many tractable properties in statistics. Yet, these simple models have not been widely exploited in generic image categorization. Recently, GMMs have been used effectively for outdoor scene classification and annotation [168]. After model estimation, likelihood computation at testing is typically very fast.

Let a Gaussian mixture model have λ components, each of which is parameterized by $\theta_k = \{a_k, \mu_k, \Sigma_k\}$, $k = 1.. \lambda$, where a is the component prior, μ is the

component mean, and Σ is the component covariance matrix. Given a feature vector $x \in \mathbb{R}^m$, the joint probability density function of component k is defined as

$$f(x|\theta_k) = \frac{1}{\zeta} \exp \left(\frac{-(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)}{2} \right)$$

where $\zeta = \sqrt{(2\pi)^m \|\Sigma_k\|}$. Hence the mixture density is $f(x) = \sum_{k=1}^{\lambda} a_k f(x|\theta_k)$. The feature vectors in the C-T model are the same as those used in [163], where a detailed description can be found. Each training image is divided into 4×4 non-overlapping blocks, and a 6-dimensional feature vector x is extracted from each block. Three components are the mean LUV color values within the block, and the other three are moments of Daubechies-4 wavelet based texture coefficients. Our feature extraction process \mathfrak{S} for the color-texture model thus takes in an image I and computes $\mathfrak{S}(I) = \{x_1, \dots, x_D\}$, $x_i \in \mathbb{R}^6$, D depending on the image dimensions.

The parameters of GMMs are usually estimated iteratively using the Expectation Maximization (EM) algorithm, since there is no closed form solution to its maximum likelihood based estimate. Here, for each category c , the feature vectors $\mathfrak{S}(I_i^c)$ (or a subset) obtained from each training image $I_i^c, i = 1..N$ are used for building model M_c . We use Bouman's 'cluster' package [21] to do the modelling. This package allows λ to be specified, and then adaptively chooses the number of clusters less than or equal to λ , using Rissanen's minimum description length (MDL) criteria. Thus we use the feature set $\{\mathfrak{S}(I_1^c), \dots, \mathfrak{S}(I_N^c)\}$ and λ to generate C models $M_c, c = 1..C$. A test image I is thus represented by a collection of feature vectors $\mathfrak{S}(I) = \{x_1, \dots, x_D\}$, $x_d \in \mathbb{R}^6$. Here, our conditional independence assumption given model M_c is based on ignoring spatial dependence of the block features. However, spatial dependence is expected to be captured by the S-C model. Thus, based on Eq. 3.1, the log-likelihood of M_c generating I is

$$\ell_{ct}(I|M_c) = \sum_{d=1}^D \log \left(\sum_{k=1}^{\lambda} a_k^c f(x_d | \mu_k^c, \Sigma_k^c) \right). \quad (3.9)$$

For both models, the predicted categories for a given image I are obtained in rank order by sorting them by likelihood scores $\ell_{sc}(I|\cdot)$ and $\ell_{ct}(I|\cdot)$ respectively.

3.2 Annotation and Retrieval

The categorization results are utilized to perform image annotation. Tagging an image with any given word entails three considerations, namely (1) frequency of occurrence of the word among the evidence provided by categorization, (2) saliency of the given words, i.e., as is traditional in the text retrieval community, a frequently occurring word is more likely than a rare word to appear in the evidence by chance, and (3) the congruity (or fitness) of the word with respect to the entire set of words under consideration. Suppose we have a 600 category training image dataset (the setting for all our retrieval experiments), each category annotated by 3 to 5 tags, e.g., [sail, boat, ocean] and [sea, fish, ocean], with many tags shared among categories. Initially, all the tags from each category are pooled together. Tag saliency is measured in a way similar to computing inverse document frequency (IDF) in the document retrieval domain. The total number of categories in the database is C . We count the number of categories which contain each unique tag t , and denote it by $F(t)$. For a given test image I , the S-C models and the C-T models independently generate ranked lists of predicted categories. We choose the top 10 categories predicted by each model and pool them together for annotation. We denote the union of all unique words from both models by $U(I)$, which forms the set of *candidate tags*. Let the frequency of occurrence of each unique tag t among the top 10 model predictions be $f_{sc}(t|I)$ and $f_{ct}(t|I)$ respectively.

WordNet [190] is a semantic lexicon which groups English words into sets of synonyms and records the semantic relations among the synonym sets. Based on this ontology, a number of measures of semantic relatedness among words have been proposed. A measure that we empirically observe to produce reasonable relatedness scores among common nouns is the Leacock and Chowdrow (LCH) measure [153], which we use in our experiments. We convert the relatedness measure r_{LCH} from a $[0.365, 3.584]$ range to a distance measure d_{LCH} in the $[0, 24]$ range using the mapping $d_{LCH}(t_1, t_2) = \exp(-r_{LCH}(t_1, t_2) + 3.584) - 1$ for a pair of tags t_1 and t_2 . Inspired by the idea proposed in [131], we measure congruity for a candidate tag t by

$$G(t|I) = \frac{d_{tot}(I)}{d_{tot}(I) + |U(I)| \sum_{x \in U(I)} d_{LCH}(x, t)} \quad (3.10)$$

where $d_{tot}(I) = \sum_{x \in U(I)} \sum_{y \in U(I)} d_{LCH}(x, y)$ measures the all-pairwise semantic distance among candidate tags, generating scores in the $[0, 1]$ range. Essentially, a tag that is semantically distinct from the rest of the words predicted will have a low congruity score, while a closely related one will have a high score. The measure can potentially remove noisy and unrelated tags from consideration. Having computed the three measures, for each of which higher scores indicate greater support for inclusion, the overall score for a candidate tag is given by a linear combination as follows:

$$R(t|I) = a_1 f(t|I) + \frac{a_2}{\log C} \log \left(\frac{C}{1 + F(t)} \right) + a_3 G(t|I) \quad (3.11)$$

Here, $a_1 + a_2 + a_3 = 1$, and $f(t|I) = b f_{sc}(t|I) + (1 - b) f_{ct}(t|I)$ is the key model combination step for the annotation process, linearly combining the evidence generated by each model toward tag t . Experiments show that combination of the models helps in annotation significantly over either model. The value of b is a measure of relative confidence in the S-C model. A tag t is chosen for annotation only when its score is within the top ε percentile among the candidate tags, where ε intrinsically controls the number of annotations generated per image. Hence, in the annotation process, we are required to specify values of four parameters, namely $(a_1, a_2, b, \varepsilon)$. We perform annotation on a validation set of 1000 images and arrive at desirable values of precision/recall for $a_1 = 0.4$, $a_2 = 0.2$, $b = 0.3$, and $\varepsilon = 0.6$.

3.2.1 Performing Annotation-driven Search

We retrieve images using automatic annotation and the WordNet-based bag of words distances. Whenever tags are missing in either the query image or the database, automatic annotation is performed, and bag of words distance between query image tags and the database tags are computed. The images in the database are ranked by relevance based on this distance. We briefly describe the bag of words distance used in our experiments, inspired by the *average aggregated minimum* (AAM) distance proposed in [159]. The WordNet-based LCH distance $d_{LCH}(\cdot, \cdot)$ is again used to compute semantic distances between bags of words in a robust manner. Given two bags of words, $W_i = \{w_{i,1}, \dots, w_{i,m_i}\}$ and $W_j = \{w_{j,1}, \dots, w_{j,n_j}\}$,

we have the distance between them

$$\widehat{d}(W_i, W_j) = \frac{1}{2m_i} \sum_{k=1}^{m_i} \bar{d}(w_{i,k}, W_j) + \frac{1}{2m_j} \sum_{k=1}^{m_j} \bar{d}(w_{j,k}, W_i) \quad (3.12)$$

where $\bar{d}(w_{i,k}, W_j) = \min_{w_{j,l} \in W_j} d_{LCH}(w_{i,k}, w_{j,l})$. Naturally, $\widehat{d}(W_i, W_i)$ is equal to zero. In summary, the approach attempts to match each word in one bag to the closest word in the other bag and compute the average semantic distance over all such closest matches.

3.3 Experimental Validation

We investigate the performance of our system on four grounds, namely (1) how accurately it identifies picture categories, (2) how well it tags pictures, (3) how well it performs re-annotation of noisy tags, and (4) how much improvement it achieves in terms of image search, for the four scenarios described earlier. Note, however, that the improvement in image search quality is the main focus of this work. The datasets we look at consist of (a) 54,000 Corel Stock photos encompassing 600 picture categories, and (b) a 1000 picture collection from Yahoo! Flickr. Of the Corel collection, we use 24,000 to train the two statistical models, and use the rest for assessing performance.

3.3.1 Identifying Picture Categories

In order to fuse the two models for the purpose of categorization, we use a simple combination strategy [116] that results in impressive performance. Given a picture, we rank each category k based on likelihoods from both models, to get ranks $\pi_{sc}(k)$ and $\pi_{ct}(k)$. We then linearly combine these two ranks for each category, $\pi(k) = \sigma\pi_{sc}(k) + (1 - \sigma)\pi_{ct}(k)$, with $\sigma = 0.2$ working best in practise. We then assign that category, which yields the highest linearly combined score, to this picture.

We decide how well our system is doing in predicting categories by involving two picture datasets. The first one is a standard 10-class image dataset that have been commonly used for the same research question. Using 40 training pictures per category, we assess the categorization results on another 50 per category. We

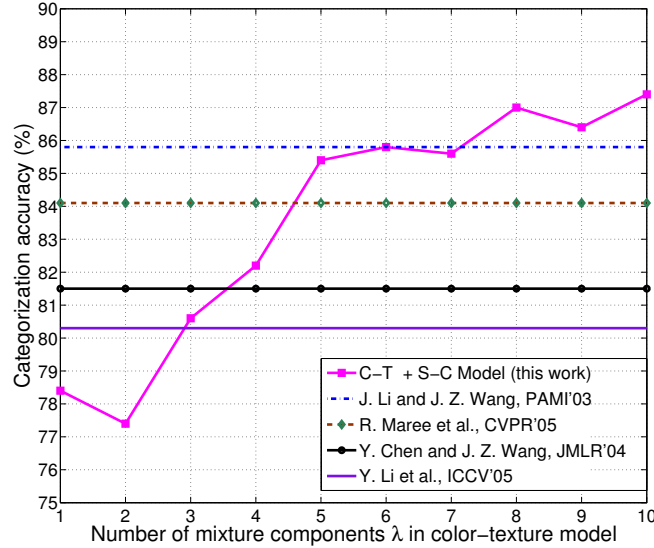


Figure 3.6. Categorization accuracies for the 10-class experiment are shown. Performance of our combined S-C+ C-T model is shown with varying number of mixture components in the C-T model. Previously reported best results shown for comparison.

compute accuracies while varying the number of mixture components in the C-T model. We present our results along with those that were previously reported on the same data, in Fig. 3.6. We see that our combined model does a better job at identifying categories than previous attempts. Not surprisingly, as we increase the number of mixture components, the C-T models become more refined. We thus continue to get improved categorization performance with greater components, although more components mean more computation as well. Our second dataset consists of the same 600 category Corel images that were used in the ALIP system [163]. With an identical training process for the two models (the number of mixture components is chosen as 10), we observe the categorization performance on a separate set of 27,000 pictures. What we find is that the actual picture categories coincide with our system’s top choice 14.4% of times, within top two choices 19.3% of the times, and within top three choices 22.7% of the times. The corresponding accuracy values for the ALIP system are 11.9%, 17.1%, and 20.8%.

Our system takes about 26 seconds to build a structure-composition category model, and about 106 seconds to build a color-texture model, both on a 40 picture training set. As with generative models, we can independently and parallelly build









				
Our Labels	sky, city, modern, building, Boston	door, pattern, Europe, historical building, city	train, car, people, life, city	man, office, indoor, fashion, people
Flickr Labels	Amsterdam, building, Mahler4, Zuidas	Tuschinski, Amsterdam	honeymoon, Amsterdam	hat, Chris, cards, funny
				
Our Labels	lake, Europe, landscape, boat, architecture	lion, animal, wild life, Africa, super-model	speed, race, people, Holland, motorcycle	dog, grass, animal, rural, plant
Flickr Labels	Amsterdam, canal, water	leopard, cat, snagged photo, animal	Preakness, horse, jockey, motion, unfound photo, animal	Nanaimo Torgersons, animal, Quinn, dog, cameraphone

Figure 3.7. Sample automatic tagging results on some Yahoo! Flickr pictures taken in Amsterdam, show along with the manual tags.

the models for each category and type. To predict the top five ranked categories for a given test picture, our system takes about 11 seconds. Naturally, we have a system that is orders of magnitude faster than the ALIP system, which takes about 30 minutes to build a model, and about 20 minutes to test on a picture, all else remaining the same. Most other automatic tagging systems in the literature do not explicitly report speed. However, a number of them depend on sophisticated image segmentation algorithms, which can well become the performance bottleneck. The improved performance in model building means that (1) even larger number of models can be built, e.g., one model per unique tag, and (2) the modeling process can be made dynamic (re-training at intervals) to accommodate changing picture collections, e.g., Web sites that allow users to upload pictures.

3.3.2 Tagging the Pictures

We now look at how our system performs when it comes to automatic picture tagging. Tagging is fast, since it depends primarily on the speed of categorization. Over a random test set of 10,000 Corel pictures, our system generates about seven tags per picture, on an average. We use standard metrics for evaluating annotation performance. These are *precision*, the fraction of tags predicted that are actually correct, and *recall*, the fraction of actual tags for the picture that are correctly guessed. We find that average precision over this test set is 22.4%, while average recall is 40.7%. Thus, on an average, roughly one in four of our system’s predicted tags are correct, while two in five correct tags are guessed by our system. In general, results of this nature can be used for filtering and classification purposes. A potential domain of thousands of tags can be reduced to a handful, making human tagging much easier, as used in the ALIPR system (<http://www.alipr.com>). Increased homogeneity and reduced ambiguity in the tagging process are additional benefits.

We make a more qualitative assessment of tagging performance on the 1,000 Flickr pictures. We point out that the training models are still those built with Corel pictures, but because they represent the spectrum of photographic images well, they serve as fair ‘knowledge bases’. We find that in this case, most automatically generated tags are meaningful, and generally very encouraging. In Fig. 3.7, we present a sampling of these results. Getting quantitative performance is harder here because Flickr tags are often proper nouns (e.g., names of buildings, people) that are not contained in our training base.

3.3.3 Re-annotating Noisy Tags

We assess the performance of our annotation system in improving tagging at high noise levels. The scenario of noisy tags, at a level denoted by e , is simulated in the following manner. For each of 10,000 test pictures with the original (reliable) tags, a new set of tags are generated by replacing a tag by a random tag e fraction of the times, at random, and are unchanged at other times. The resultant noisy tags for these test images, when assessed for performance, give precision and recall values that are directly correlated with e . In the absence of learned models, this

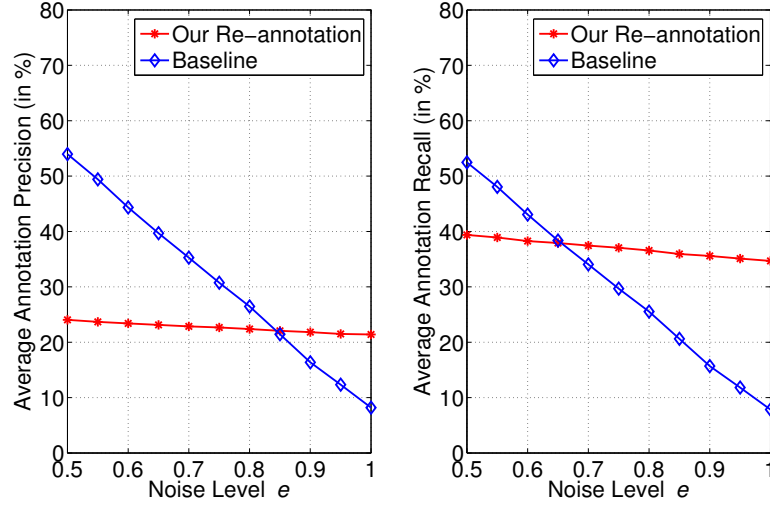


Figure 3.8. Precision (left) and recall (right) achieved by re-annotation, with varying noise levels in original tags. Note that the linear correlation of the baseline case to e is intrinsic to the noise simulation.

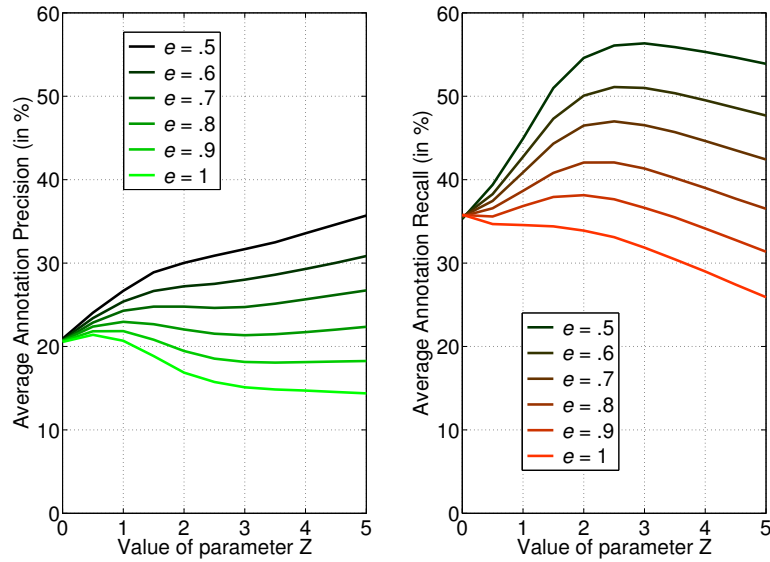


Figure 3.9. Precision (left) and recall (right) achieved by re-annotation, varying parameter Z , shown for 5 noise levels.

is our *baseline* case. When such models are available, we can use the noisy tags and the categorization models to re-annotate the pictures, because the noisy tags still contain exploitable information. We perform re-annotation by simply treating

each noisy tag t of a picture I as an additional instance of the word in the pool of *candidate tags*. In effect, we increment the values of $f_{sc}(t|I)$ and $f_{ct}(t|I)$ by a constant Z , thus increasing the chance of t to appear as an actual tag. The value of Z controls how much we wish to ‘promote’ these tags, and is naturally related to the noise in the tags. The annotation precision and recall achieved by this approach, with e varying from 0.5 (moderately noisy) to 1 (completely noisy, no useful information) for the case of $Z = 0.5$ is shown Fig. 3.8. We notice that at high levels of noise, our re-annotation produces better performance than the baseline. A more general analysis of the trends, with larger values of Z (i.e., greater confidence in the noisy tags), is summarized in Fig. 3.9. The graph shows precision and recall for our re-annotation, varying Z for each of 5 error levels. We observe that for the same value of Z , less noisy tags lead to better re-annotation. Moreover, after reaching a peak near $Z = 2.5$, the recall starts to drop, while precision continues to improve. This graph can be useful in selecting parameters for a desired precision/recall level after re-annotation, given an estimated level of noise in the tags.

3.3.4 Searching for Pictures

We examine how the actual image search performance improves with our approach, compared to traditional ways. We assume that either the database is partially tagged, or the search is performed on a picture collection visually coherent with some standard ‘knowledge base’. In all our cases, the statistical models are learned from the Corel dataset. For scenario 4, we assume that everything is tagged, but some tags are incorrect/inconsistent. Once again, we train a knowledge base of 600 picture categories, and then use it to do categorization and automatic tagging on the test set. This set consists of 10,000 randomly sampled pictures from among the remaining Corel pictures (those not used for training).

We now consider the four image search scenarios discussed in Sec. 3.0.1. For each scenario, we compare results of our annotation-driven image search strategy with alternative strategies. For those alternative strategies involving CBIR, we use the IRM distance used in the SIMPLicity system [278] to get around the missing tag problem in the databases and queries. We chose the alternative strategies and

their parameters by considering a wide range of possible methods. We perform assessment of the methods based on the standard information retrieval concepts of precision (percentage of retrieved pictures that are relevant) and recall (percentage of relevant pictures that are retrieved). We consider two pictures/queries to be relevant whenever there is overlap between their actual set of tags.

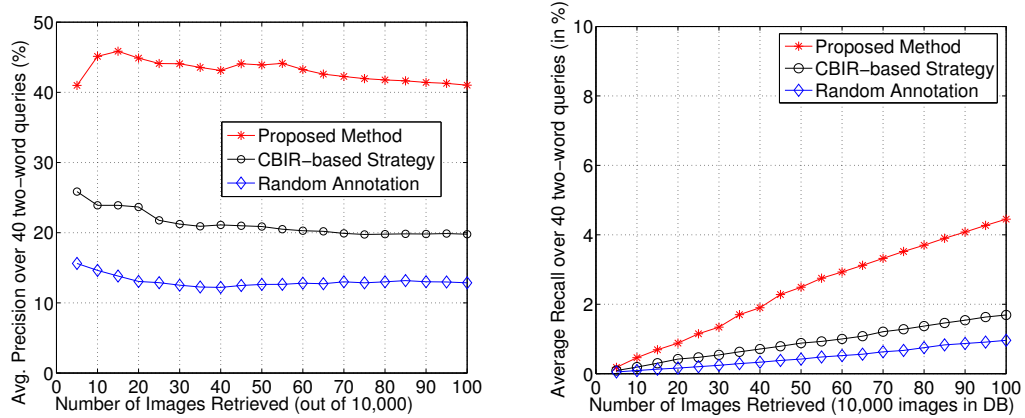


Figure 3.10. Precision (left) and recall (right) under scenario 1, compared to baseline.

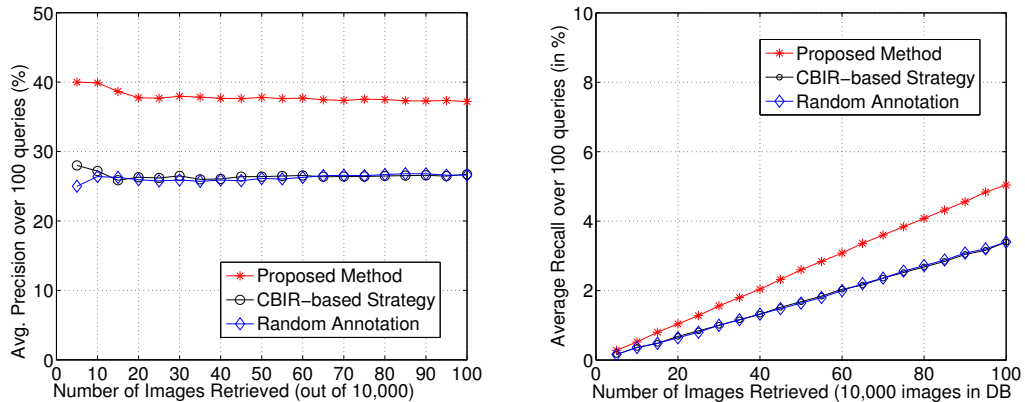


Figure 3.11. Precision (left) and recall (right) under scenario 2, compared to baseline.

Scenario 1: Here, the database does not have any tags. Queries may either be in the form of one or more keywords, or tagged pictures. Keyword queries on an untagged picture database is a key problem in real-world image search. We look at 40 randomly chosen pairs of query words (each word is chosen from the 417 unique words in our training set). In our strategy, we perform search by first automatically tagging the database, and then retrieving images based on bag of the

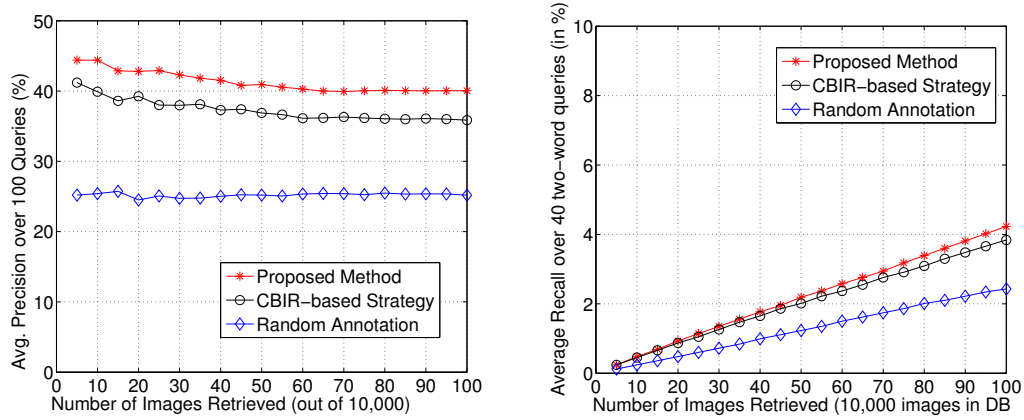


Figure 3.12. Precision (left) and recall (right) under scenario 3, compared to baseline.

words distances between query tags and our predicted tags. The alternative CBIR-based strategy used for comparison is as follows: without any image as query, CBIR cannot be performed directly on query keywords. Instead, suppose the system is provided access to a knowledge base of tagged Corel pictures. A random set of three pictures for each query word is chosen from the knowledge base, and the IRM distances between these images and the database are computed. We then use the average IRM distance over the six pictures for ranking the database pictures. We report these two results, along with the random results, in Fig. 3.10. Clearly, our method performs significantly better than the alternative approach.

Scenario 2: Here, the query is an untagged picture, and the database is tagged. What we do here is first tag the query picture automatically, and then rank the database pictures using bag-of-words distance. We randomly choose 100 query pictures from Corel and test it out on the database of 10,000 pictures. The alternative CBIR-based strategy we use is as follows: the IRM distance is used to retrieve five (empirically observed to be the best count) pictures most visually similar to the query, and the union of all their tags is filtered using the expression for $R(t|I)$ to get automatic tags for the query (the same way as our annotation is filtered, as described in Sec. 3.2). Now, search proceeds in a manner identical to ours. We present these results, along with the random scheme, in Fig. 3.11. As we see, our strategy has a significant performance advantage over the alternate strategy. The CBIR-based strategy performs almost as poorly as the random scheme, which is probably due to the direct use of CBIR for tagging.

Scenario 3: In this case, neither the query picture nor the database is tagged. We test 100 random picture queries are tested on the 10,000 image database. Our strategy is simply to tag both the query picture as well as the database automatically, and then perform bag-of-words based retrieval. Without any tags present, the alternative CBIR-based strategy used here is essentially a standard use of the IRM distance to rank pictures based on visual similarity to the query. We present these results, along with the random case, in Fig. 3.12. Once again, we see the advantage of our common image search framework over straightforward visual similarity based retrieval. What we witness here is how, in an indirect way, the learned knowledge base helps to improve search performance, over a strategy that does not involve statistical learning.

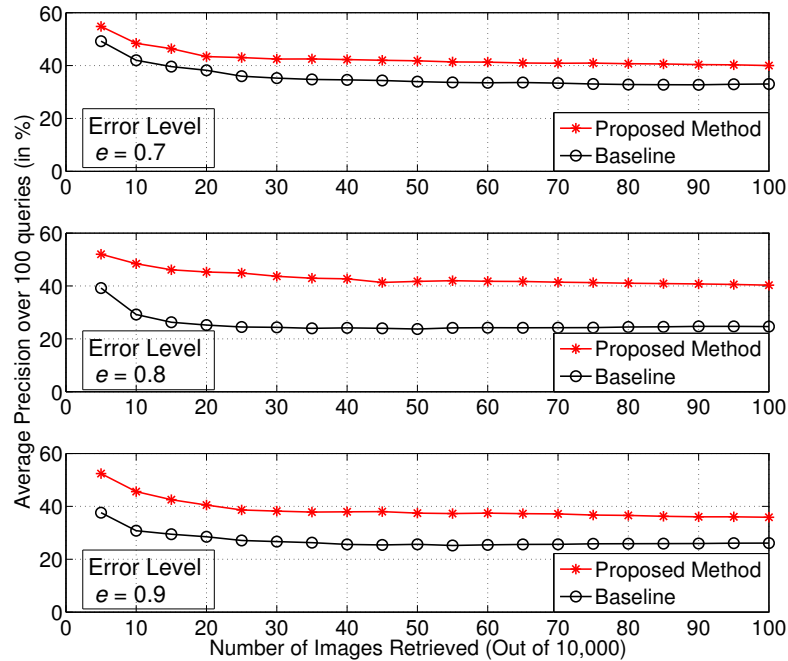


Figure 3.13. Retrieval precision for scenario 4 at three noise levels.

Scenario 4: Here, the query picture and the database are both fully tagged, but many tags are incorrect, a situation that often arises under user-driven tagging, for reasons such as subjectivity. Here, our re-annotation approach (refer to Sec. 3.3.3) is used to refine these noisy tags prior to performing retrieval. Introducing noise levels of $e = 0.7, 0.8$, and 0.9 , and using parameter $Z = 1.5$, we test 100

random picture queries on the 10,000 images. For this, queries and the database are first re-annotated. Alternate strategy here includes the baseline case, as described in Sec. 3.3.3. The precision results over the top 100 retrieved pictures, for the three noise levels are shown in Fig. 3.13. Interestingly, even at $e = 0.7$, where our re-annotation approach does not surpass the baseline in annotation precision, it does so in retrieval precision, making it a useful approach at this noise level. Moreover, the difference with the baseline is maximum at noise level 0.8. Note that for $e \leq 0.5$, our approach did not yield better performance than the baseline, since the tags were sufficiently ‘clean’. These results suggest that at relatively high noise levels, our re-annotation approach can lead to significantly improved image retrieval performance, compared to the baseline.

3.4 Summary

We have proposed a novel annotation-driven image search approach. By experimenting with standard picture sets as well as publicly contributed collections, we have shown its potential in various aspects. The framework is standard for different scenarios and different types of queries, which should make implementation fairly straightforward. We see that in each such scenario, our approach turns out to give more promising results than traditional methods. The categorization performance in itself is an improvement upon previous attempts. Moreover, we are able to categorize and tag the pictures in very short time. All of these factors make our approach attractive for real-world implementation.

Bridging Dynamic Semantic Gap: Adapting Automatic Image Tagging via Meta-learning

Automatic image annotation is the task of producing tags for images based on their visual content. In the context of machine learning, automatic annotation falls into the class of learning tasks that involve making multiple binary decisions on each data point. If we could generate comprehensive, accurate, semantically meaningful tags for images, it would bring image organization up to roughly the level of text documents. Of late, many image annotation ideas have been proposed [9, 27, 34, 82, 92, 129, 166, 167, 171, 193, 275, 282, 291, 295]. Virtually all propositions are based on supervised learning, and take roughly the following form, (a) use a limited set of manually tagged images to train generative or discriminative models for associating visual features to tags, and (b) given a new image, use the model inferences on its visual content to assign a variable-size set of tags from its limited vocabulary. Performance is typically reported based on a *static* training/testing split of one or more manually tagged image datasets. However, this may not accurately represent real-world settings for automatic tagging.

We argue for a fundamentally different notion of automatic image annotation in real-world settings involving users, whereby the goal is mainly to mimic the users in the tagging process as closely as possible. To elaborate, let us take the case of Yahoo! [88], a dynamic online photo-sharing Website where collaborative

image tagging, also referred to as *folksonomic tagging*, plays a key role in making the image collections meaningful, organizable by semantics and searchable by text [273]. It is also an apt service platform for automatic image tagging. If the goal of the tagging system is to reduce human effort, or to assist humans tag photos more accurately and/or with less effort (e.g., by automatically suggesting tags to choose from), then a natural performance metric will be how closely the automatically generated tags mimic user-generated ones for the same images. If the goal is indeed to maximize performance defined in this manner, then it is easy to see how the standard procedure of splitting a dataset into training and test, and computing performance on the test set, can be misleading. What is problematic is the assumption that training and test cases are sampled from the same underlying distribution in the joint image-tag space. There are at least three factors that can affect the test time image-tag distribution and hence the generalization of model training to, say, a Flickr tagging scenario:

- **Context:** The Flickr users may tag the same photos differently than the users who tagged the training dataset. Some visual elements may be more commonly tagged in a different language, a local dialect, or slang, rather than their correct English language descriptors. The types of images uploaded by the users may also be different.
- **Time evolution:** The kinds of photos uploaded and the nature of user tags may evolve with time for various reasons, including news, current affairs, and changing political situations. Trends may spread through the network of users, and over time, get more pervasive. This trend of evolution over time has been observed in photo-sharing systems and reported in [57, 73].
- **User/community preferences:** The kinds of photos uploaded, the tags given to them, or the frequency distribution over tags, may vary to a great extent across individuals and small communities. This is particularly evident in Flickr, where the user has the option of choosing from her own set of previously used tags [93], thereby promoting locality in the tag space.

In the remainder of this chapter, we will refer to all three of these commonly as a changed *image tagging environment* (ITE). In this sense, an annotation system

which has the ability to adapt to changes in ITE can potentially be effective in all three scenarios.

4.0.1 A Challenging Setting for Learning

Given that generic images vary widely in their composition, learning a mapping function from the image space to the tag space is extremely challenging, as has been found in previous studies [58, 242]. Most proposed image annotation approaches therefore resort to complex statistical learning models in efforts to learn semantics from visual content, which involve computationally expensive training. We are unlikely to be able to reduce the complexity of these models significantly. On the other hand, we require efficient annotation adaptability to various ITE changes. If the ITE change involves expanding the vocabulary to include concepts that were trained under different tag names, we also wish to take advantage of the previously learned knowledge.

One obvious way to adapt an automatic annotation system to a changed ITE would be to re-train it using data sampled from that ITE. Unfortunately, with most systems this can take hours or even days [9, 163], which means that for dynamically changing ITE, a significant amount of latency will be introduced, and training resources will stay blocked on a continual basis. When changes are frequent or the problem is scaled up, it will be impossible to keep up with the changes this way. Furthermore, most annotation systems are not well suited for incremental learning. We thus require a learning system with a unique need; given a core learning algorithm that is expensive to train, we require an augmentation which can quickly adapt to various kinds of ITE changes in a scalable manner, while taking advantage of previously learned knowledge. Additionally, ITE changes over short periods are typically localized, which presents an opportunity to train incrementally rather than having to re-train over entire datasets at each change point.

4.0.2 Overview of Our Approach

To meet the aforementioned needs, we propose a meta-learning layer above the core learning system which can adapt to evolving ITE incrementally, in a light-weight

manner, without requiring a re-training of the core system. Suppose there is a core annotation *black-box* that, by some means (e.g., a one-time learning process), can analyze the visual content of images and generates sets of tags. Expectedly, these tags come from a fixed size vocabulary, and the association of visual features with these tags is intrinsic to the particular set of training images used. Our goal is to expand the capabilities of the black-box to allow adaptability to various kinds of ITE changes while maximizing the use of previously learned knowledge.

In order to make a case for meta-learning, let us draw analogy with a robot learning to solve math problems. Suppose robot X has initially learned to solve a fundamentally important pool of math problems. In this process, its fundamentals have been cleared and it has developed critical but limited skills. This X is the black-box system, and initial problem pool is an ITE. Now, X is required to solve harder, more diverse problems. Here, the problems are sampled from a larger source, and the same tactics may not work. We thus have a changed ITE, and X needs to get adapted to the new conditions and challenges. It may be too expensive to re-train the robot on an all-inclusive set of problems all over again, and even that training may not be sufficient to solve a continuously evolving problem pool. Instead, if we have another robot Y which is aware of the early-stage training of X , observes its response to new problems, analyzes its mistakes and is able to rectify it eventually by taking the output of X and produces a new one, the combined $X+Y$ system can likely be more powerful. The assumption here is that the output of X is still fundamentally in the right direction, and only needs refinement and perhaps a better representation of the solution. The knowledge acquired by X can be used to solve problems in a new setting, similar in principle to *inductive transfer* or transfer learning. Thus, Y here is equivalent to the meta-learning framework, in the sense that it sits and observes X , learns from mistakes, and eventually improves upon the output of X .

In terms of image tagging, the scenario is as follows. We have a fundamentally grounded black-box (which we assume to be any annotation system proposed and found to perform moderately), a meta-learning framework which we call PLMFIT, which, for some fraction of the images in a changed ITE, can observe the tags generated by the black-box, as well as the ground-truth tags in those cases, and quickly learn to adapt to it. The change in ITE can be any of (a) change in context,

(b) evolution over time, or (b) change in the people who the images belong to or are tagged by. Because the observations are received differently for the three cases, the PLMFIT training process varies as well. For a change in context, we can train the PLMFIT one-time using a batch of image samples collected from the new ITE. For changes over time, PLMFIT can learn incrementally as new images get uploaded, assessed by the black-box, and tagged by users. For a particular person’s uploaded images, PLMFIT can be trained on her previously tagged pool of images, if such a pool exists. We therefore first need a formulation for the meta-learning component PLMFIT, and then the algorithms for training it to achieve adaptability under varied settings.

4.0.3 Related Work

There is a wealth of machine learning literature on meta-learning, incremental learning, and inductive transfer, concepts that our work is directly related to, although we are unaware of the use of these techniques for image tagging or related problem areas. Here, we briefly discuss literature most pertinent to this work. The term meta-learning has historically been used to describe the learning of meta-knowledge about learned knowledge. Research in meta-learning covers a wide spectrum of approaches and applications, as presented in a review [272]. One of the most popular meta-learning approaches, *boosting* is widely used in supervised classification [90]. Boosting involves iteratively adjusting weights assigned to data points during training, to adaptively reduce misclassification. In *stacked generalization*, weighted combinations of responses from multiple learners are taken to improve overall performance [290]. The goal here is to learn optimal weights using validation data, in the hope of generalization to unseen data. Another research area under the meta-learning umbrella that bears relevance to our work is *inductive transfer*. Research in inductive transfer is grounded on the belief that knowledge assimilated about certain tasks can potentially facilitate the learning of certain other tasks [32]. A recent workshop [239] at the NIPS conference was devoted to discussing advances and applications of inductive transfer. Incremental learning deals with adapting predictions to contextual changes as new data enters the system. Adapting to radical contextual changes via incremental learning was

proposed by [287]. Incrementally learning support vectors as and when training data is encountered has been explored as a scalable supervised learning procedure by [33]. The idea of *partial instance memory*, whereby only a relevant subset of the incoming stream of training samples are maintained (thereby saving memory) and used for incremental learning, was proposed and shown to be empirically effective [182]. Authors Kolter and Maloof proposed a weighted ensemble of incremental learners for *concept drift* [230], a formalization of the idea that concepts to learn change over time [146]. In the case of image tagging, it is the mapping between tags and visual semantics that drifts over time.

Research in automatic image annotation can be roughly categorized into two different ‘schools of thought’: (1) Words and visual features are jointly modeled to yield compound predictors describing an image or its constituent regions. The words and image representations used could be disparate [82, 129] or single vectored representations of text and visual features [193, 171]. (2) Automatic annotation is treated as a two-step process consisting of supervised image categorization, followed by word selection based on the categorization results [34, 166, 27]. While the former approaches can potentially label individual image regions, ideal region annotation would require precise image segmentation, an open problem by itself in computer vision. While the latter techniques cannot label regions, they are typically more scalable to large image collections. Though less relevant to our work, approaches such as [282] employ yet another philosophy for image tagging, i.e., to avoid learning and hence stay ‘model-free’. Specifically within the machine learning community, significant recent work in the domain of image categorization and annotation include [9], who proposed the use of latent dirichlet allocation [16] for the purpose of associating images and tags, [42], who proposed a multiple-instance learning approach to image categorization, and [86], who explore the use of stationary visual features for the detection of cats in gray-scale images.

Evolution of image tags over time, or their variation across people in online communities, have only recently begun to get research focus. Researchers at Yahoo! have studied the problem of visualizing the evolution of salient tags popular among Flickr users [73]. Assuming such an evolution of tags over time in general, [57] proposed a meta-learning approach to handling the evolution as part of an automatic image tagging system, and found the approach to be effective on

traces obtained from the Alipr system [3, 166]. Image tag recommendation strategies for Flickr users was proposed by [238] and found to be effective. On similar lines, personalized image tag recommendation was briefly explored by [93] using tag co-occurrences and tag history, but neither approach involved the exploitation of visual content of images for tagging purposes.

4.0.4 Contributions

Broadly speaking, the main contributions of this work are learning approaches and algorithms that greatly improve automatic image tagging in real-world settings. We take a complex learning task, that of finding a mapping function from low-level image features to semantically meaningful tag sets, and create an appropriate meta-learner around it which is well-suited to incremental training, and designed to meet the needs of real-world usage. To the best of our knowledge, this is the first attempt at meta-learning and incremental learning for image tagging. Moreover, because image tagging is not simply formulated as a classification problem, existing meta-learning and incremental learning methods cannot be applied directly. We thus design a new statistical modeling approach, which also aims to achieve efficiency in real-world deployment. Specific contributions are summarized below.

- We propose a principled, lightweight, meta-learning framework for image tagging (PLMFIT), based on few simplifying assumptions, inspired by inductive transfer, and backed by experiments, which can augment any existing annotation system. This is the basic component that allows adaptation to ITE changes. Experimentally, we find that PLMFIT can adapt to new contexts¹ very effectively, showing dramatic performance improvement over the core systems.
- We propose algorithms to use PLMFIT to adapt to concept drift [230], or changes in ITE over time. Specifically, we propose fast algorithms for incremental/decremental learning over time that take advantage of the simplicity of the PLMFIT formulation. Two different memory models for incremental learning, *persistent* and *transient*, are explored. Experimentally, we find

¹One can think of a contextual change as a *global change* of ITE, e.g., when a model is trained using labeled Corel images, but applied to Flickr tagging.

the algorithms to be highly effective in adapting over time on a real-world dataset.

- We propose algorithms for personalized tagging, adapting to ITE changes across people, assuming that some of their photos are already tagged. As part of this, we propose an approach to allow expansion of the tag vocabulary beyond the initial set, and incrementally updating model parameters for new users. Experiments on actual user data show that personalization greatly boosts tagging performance.
- Throughout, we use real-world data to justify each aspect of our approach, as well as to show that assumptions of changing ITE over time and across people hold true.

In all these cases, we assume the existence of a black-box annotation system that exhibits better-than-random performance. This is the *only* assumption we make about the black-box system, so most previously reported annotation algorithms qualify. Experiments are conducted using two different annotation systems. The datasets used for the experiments include the popular Corel image, two real-world image traces and user-feedback obtained from the Alipr system [3], and a large set of collaboratively tagged images obtained from Yahoo! [88], spanning hundreds of users.

The rest of this chapter is arranged as follows. The technical details of PLMFIT are presented in Sec. 4.1, including model estimation and smoothing steps. In Sec. 4.2 we present the algorithms for using PLMFIT for adaptation over context, time, and people. Experimental results are presented in Sec. 4.3. We discuss results, limitations, and implications in Sec. 4.4. We conclude in Sec. 4.5.

4.1 PLMFIT: Principled Meta-learning Framework for Image Tagging

In this section, we describe PLMFIT, our meta-learning framework that forms the backbone of this work. Consider any black-box image annotation system, such as [9, 27, 34, 82, 92, 129, 166, 167, 171, 193, 275, 295, 291], that takes an image

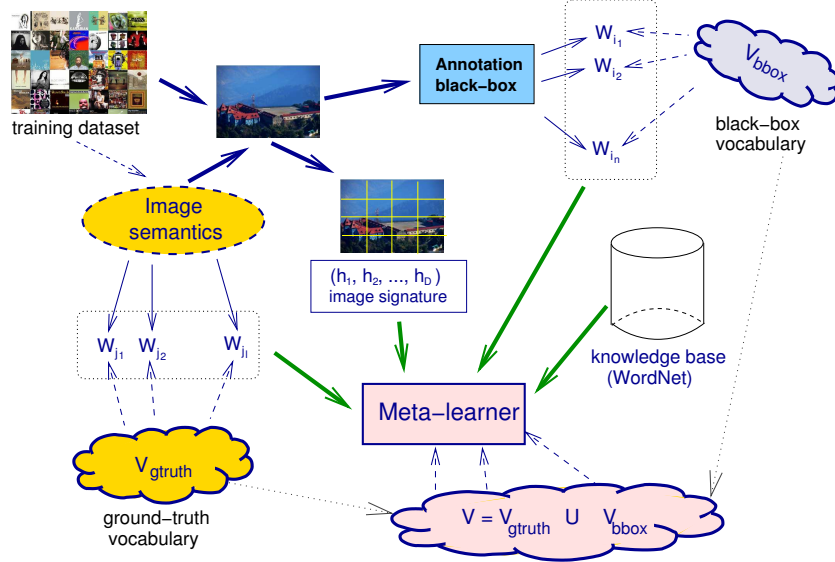


Figure 4.1. High-level overview of our PLMFIT meta-learning framework.

as input and guesses one or more tags as its annotation. This work does not deal with the algorithm behind the annotation black-box, but simply assumes that it captures, from visual analysis, the semantics of the images in the form of tags, to a better-than-random degree of reliability. Now let us assume that for a certain set of images not used in training the black-box, ground-truth tags are available. Clearly, for each such image, two sets of tags are available, (1) the ground-truth tags, and (2) the tags predicted by the black-box. Let us also assume that we have at our disposal a knowledge base such as WordNet [190], and the original images from which we can independently extract visual features. A high-level overview of a meta-learning framework that incorporates these components is shown in Fig. 4.1.

As part of motivating the model, in what follows, we will graphically represent empirical evidence to support its various components. For this purpose, we have used 10,000 images randomly chosen from the Alipr dataset which is described in detail in Sec. 4.3. Each image is accompanied by human-provided tags as well as the corresponding machine predictions [166, 3]. In the remainder of this section, we begin with notation and an overview of the basic components of PLMFIT, then describe its formulation, and finally present estimation steps.

4.1.1 Notation and Basics

Let the black-box annotation system be known to have a tag vocabulary denoted by V_{bbox} . For a given ITE, let us denote the ground-truth vocabulary by V_{gtruth} . Let the full vocabulary of interest, V , be their union, i.e., $V = (V_{bbox} \cup V_{gtruth}) = \{w_1, \dots, w_K\}$, where size $K = |V|$. Given an image I , the black-box predicts a set of tags to be its correct annotation. We introduce indicator variables $G_w \in \{0, 1\}$, $w \in V$, to denote if a *guess* w is predicted or not. Similarly, for ground-truth, let $A_w \in \{0, 1\}$ denote the whether w is a correct tag or not. The black-box can be denoted by a function f_{bbox} mapping an image I to a set of indicator variables, i.e., $f_{bbox}(I) = \{G_{w_1}, \dots, G_{w_K}\}$, and ground-truth can be denoted analogously by function $f_{gtruth}(I) = \{A_{w_1}, \dots, A_{w_K}\}$.

Regardless of the abstraction of visual content that the black-box uses for annotation, the pixel-level image representation is still available to the meta-learner. If some visual features, which can be cheaply extracted and hence are suitable for highly efficient incremental modeling, represent a different abstraction than what the black-box uses, they can help form a different ‘viewpoint’ and thus can potentially complement semantics recognition. Suppose we have a D -dimensional vector representation for such visual features extracted from an image, denoted by $f_{vis}(I) = (h_1, \dots, h_D)$. Note that though non-vector visual representations (e.g., variable sized sets of features) can be more powerful representations, we use this form here for computational advantages.

Furthermore, the English language semantic lexicon WordNet, which has been previously found useful for image one that has been found useful for automatic tagging [131, 55], is also available at our disposal. In particular, WordNet-based semantic relatedness measures [153] have benefited annotation tasks. While this is a potentially useful external knowledge base, it is rendered useless for non-English words, proper nouns, contemporary slang, and incorrect usages that are commonly found among user tags. However, in cases for which WordNet based relatedness measures can be computed, we show how the transfer of learned knowledge can be better directed.

4.1.2 Model Formulation

We first describe the PLMFIT formulation, and then present the estimation steps. For each image I , a decision is taken on each word independently, based on all available information. To do so, we compute the following *odds* in favor of each word $w_j \in V$ to be a ground-truth tag, conditioned on pertinent information:

$$\ell_{w_j}(I) = \frac{Pr(A_{w_j} = 1 \mid f_{bbox}(I), f_{vis}(I))}{Pr(A_{w_j} = 0 \mid f_{bbox}(I), f_{vis}(I))} \quad (4.1)$$

In what follows, we will make simplifying assumptions to make this formulation tractable. Note that here $f_{bbox}(I)$ (and similarly, the other terms) denotes a *joint* realization of the corresponding random variables given the image I . Using Bayes' Rule, we can re-write:

$$\ell_{w_j}(I) = \frac{Pr(A_{w_j} = 1, f_{bbox}(I), f_{vis}(I))}{Pr(A_{w_j} = 0, f_{bbox}(I), f_{vis}(I))} \quad (4.2)$$

If realization of variable A_{w_j} is denoted by $a_j \in \{0, 1\}$ and that of variables G_{w_i} for each word w_i are denoted by $g_i \in \{0, 1\}$, then using the chain rule of probability, and without loss of generality, we can re-write the following:

$$\begin{aligned} & Pr(A_{w_j}=a_j, f_{bbox}(I), f_{vis}(I)) \\ = & Pr(G_{w_j}=g_j, A_{w_j}=a_j, \bigcap_{i \neq j} (G_{w_i}=g_i), f_{vis}(I)) \\ = & Pr(G_{w_j}=g_j) \times Pr(A_{w_j}=a_j \mid G_{w_j}=g_j) \\ \times & Pr\left(\bigcap_{i \neq j} (G_{w_i}=g_i) \mid A_{w_j}=a_j, G_{w_j}=g_j\right) \\ \times & Pr\left(f_{vis}(I) \mid \bigcap_{i \neq j} (G_{w_i}=g_i), A_{w_j}=a_j, G_{w_j}=g_j\right) \end{aligned} \quad (4.3)$$

The odds in Eq. 4.1 can now be factored using Eq. 4.2 and 4.3:

$$\ell_{w_j}(I) = \frac{Pr(A_{w_j} = 1 \mid G_{w_j}=g_j)}{Pr(A_{w_j} = 0 \mid G_{w_j}=g_j)} \quad (4.4)$$

$$\begin{aligned} & \times \frac{\Pr(\bigcap_{i \neq j} (G_{w_i} = g_i) \mid A_{w_j} = 1, G_{w_j} = g_j)}{\Pr(\bigcap_{i \neq j} (G_{w_i} = g_i) \mid A_{w_j} = 0, G_{w_j} = g_j)} \\ & \times \frac{\Pr(f_{vis}(I) \mid A_{w_j} = 1, \bigcap_{i \neq j} (G_{w_i} = g_i), G_{w_j} = g_j)}{\Pr(f_{vis}(I) \mid A_{w_j} = 0, \bigcap_{i \neq j} (G_{w_i} = g_i), G_{w_j} = g_j)} \end{aligned}$$

Note that the priors ratio $\frac{\Pr(G_{w_j} = g_j)}{\Pr(G_{w_j} = g_j)}$ is 1, and hence is eliminated. The ratio $\frac{\Pr(A_{w_j} = 1 \mid G_{w_j} = g_j)}{\Pr(A_{w_j} = 0 \mid G_{w_j} = g_j)}$ is a *sanity check* on the black-box for each word. For $g_j = 1$, it can be paraphrased as “Given that word w_j is guessed by the black-box for I , what are the odds of it being correct?”. Naturally, a higher odds indicates that the black-box has greater *precision* in guesses (i.e., when w_j is guessed, it is usually correct). A similar paraphrasing can be done for $g_j = 0$, where higher odds implies lower word-specific *recall* in the black-box guesses. A useful annotation system should be able to achieve independently (word-specific) and collectively (overall) good precision and recall. These probability ratios therefore give the meta-learner indications about the black-box model’s strengths and weaknesses over its entire vocabulary. In Fig. 4.2, we have plotted empirical estimates of these probability terms for frequently occurring tags.

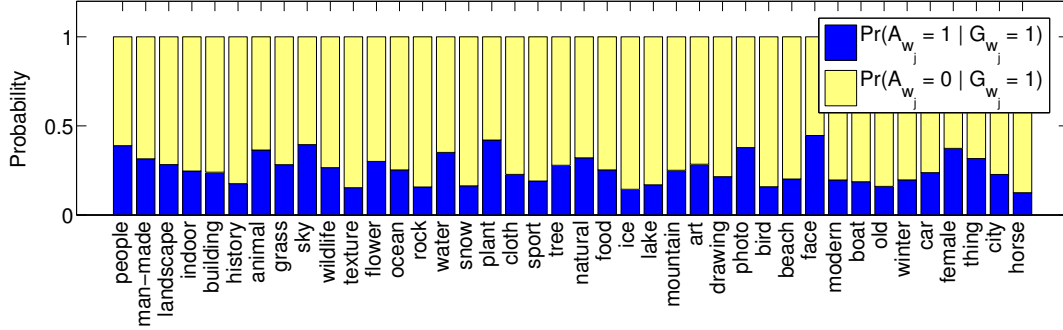


Figure 4.2. Estimates of $\Pr(A_{w_j} \mid G_{w_j} = 1)$ as obtained empirically with images from the Alipr dataset (see Sec. 4.3) for 40 most frequently occurring tags (decreasing frequency from left to right). As can be seen, the black-box is much more precise in predicting tags such as ‘face’ or ‘plant’, compared to ‘texture’ or ‘ice’.

When $g_j = 1$, the ratio $\frac{\Pr(\bigcap_{i \neq j} (G_{w_i} = g_i) \mid A_{w_j} = 1, G_{w_j} = g_j)}{\Pr(\bigcap_{i \neq j} (G_{w_i} = g_i) \mid A_{w_j} = 0, G_{w_j} = g_j)}$ in Eq. 4.4 relates each correctly or wrongly guessed word w_j to how every other word $w_i, i \neq j$ is guessed by the black-box. This component has strong ties with the concept of co-occurrence

popular in the language modeling community, the difference being that here it models the word co-occurrence of the black-box’s outputs with respect to ground-truth. Similarly, for $g_j = 0$, it models how certain words do not co-occur in the black-box’s guesses, given the ground-truth. Since the meta-learner makes decisions about each word independently, it is intuitive to separate them out in this ratio as well. That is, the question of whether word w_i is guessed or not, given that another word w_j is correctly/wrongly guessed, is treated independently. Furthermore, efficiency and robustness become major issues in modeling joint probability over a large number of random variables, given limited data. Considering these factors, we assume the guessing of each word w_i conditionally independent of each other, given a correctly/wrongly guessed word w_j , leading to the following approximation:

$$Pr\left(\bigcap_{i \neq j} (G_{w_i}=g_i) \mid A_{w_j} = a_j, G_{w_j}=g_j\right) \approx \prod_{i \neq j} Pr(G_{w_i}=g_i \mid A_{w_j} = a_j, G_{w_j} = g_j)$$

The corresponding ratio term can then be written as

$$\frac{Pr(\bigcap_{i \neq j} (G_{w_i}=g_i) \mid A_{w_j} = 1, G_{w_j}=g_j)}{Pr(\bigcap_{i \neq j} (G_{w_i}=g_i) \mid A_{w_j} = 0, G_{w_j}=g_j)} = \prod_{i \neq j} \frac{Pr(G_{w_i}=g_i \mid A_{w_j} = 1, G_{w_j} = g_j)}{Pr(G_{w_i}=g_i \mid A_{w_j} = 0, G_{w_j} = g_j)}$$

A conditional multi-word co-occurrence model has been effectively transformed into that of pairwise co-occurrences, which is attractive in terms of modeling, estimation, and efficiency. While co-occurrence really happens when $g_i = g_j = 1$, the other combinations of values can also be useful, e.g., how the frequency of certain word pairs not being both guessed differs according to the correctness of these guesses. The contributions of the component ratio terms, namely $\frac{Pr(G_{w_i}=g_i \mid A_{w_j}=1, G_{w_j}=g_j)}{Pr(G_{w_i}=g_i \mid A_{w_j}=0, G_{w_j}=g_j)}$, can be understood by intuition. For the purpose of illustration, a visual plot of the ratio $\frac{Pr(G_{w_i}=1 \mid A_{w_j}=1, G_{w_j}=1)}{Pr(G_{w_i}=1 \mid A_{w_j}=0, G_{w_j}=1)}$ for a set of 30 most frequently occurring tags in the Alipr dataset is presented in Fig. 4.3. The following examples provide further insights:

- Some scene elements that are visually similar can often be mistakenly confused among themselves by the black-box. For example, if ‘sky’ is guessed for an image, and it is a correct guess more often when ‘water’ is also guessed

than when it is not, then the ratio will have a high value for $w_i = \text{'water'}$, $g_i = 1$, $w_j = \text{'sky'}$, $g_j = 1$. Here, the black-box prediction of one scene element (water) *reinforces belief* in the existence of another (sky) in the scene (See Fig. 4.3, location A).

- For some word w_j , the black-box may not have learned anything due to lack of good training images, inability to capture apt visual properties, or simply its absence in V_{bbox} . For example, consider images where ground-truth is $w_j = \text{'feline'}$ but black-box regularly guesses $w_i = \text{'cat'}$, only the latter being in its vocabulary. Here, $g_j = 0$ always, and the above ratio is high for $g_i = 1$. Here, the training on one tag *induces guesses* at another tag in the vocabulary (see also Fig. 4.3, location B).

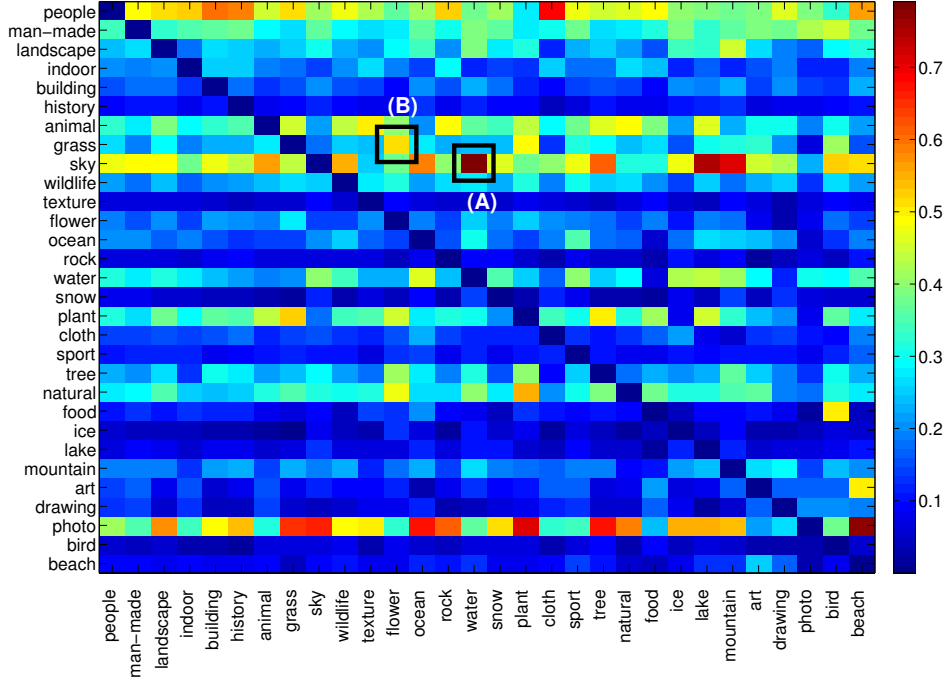


Figure 4.3. Visualization of ratio $\frac{Pr(G_{w_i}=1|A_{w_j}=1,G_{w_j}=1)}{Pr(G_{w_i}=1|A_{w_j}=0,G_{w_j}=1)}$ as obtained empirically with images from the Alipr dataset (see Sec. 4.3) for 30 most frequently occurring tags. Two interesting cases, that highlight the importance of these terms to meta-learning effectiveness, are marked. For example, the value at location (A) can be read as the ratio of probabilities of ‘water’ being guessed for an image by the black-box given that ‘sky’ is also guessed, correctly versus incorrectly.

Finally, $\frac{Pr(f_{vis}(I)|A_{w_j}=1, \bigcap_{i \neq j}(G_{w_i}=g_i), G_{w_j}=g_j)}{Pr(f_{vis}(I)|A_{w_j}=0, \bigcap_{i \neq j}(G_{w_i}=g_i), G_{w_j}=g_j)}$ in Eq. 4.4 can be simplified, since $f_{vis}(I)$, being the meta-learner’s own visual representation, is independent of the black-box’s visual abstraction. Therefore, we can re-write

$$\frac{Pr(f_{vis}(I)|A_{w_j}=1, -)}{Pr(f_{vis}(I)|A_{w_j}=0, -)} \approx \frac{Pr(h_1, \dots, h_D|A_{w_j}=1)}{Pr(h_1, \dots, h_D|A_{w_j}=0)} \quad (4.5)$$

which is the ratio of conditional probabilities of the meta-learner’s visual features extracted. We can think of this as a second, highly simplified image recognition black-box, that can be computed efficiently, and as described in Sec. 4.2.2, is suitable for incremental learning. In our experiments we have used LUV color space based histograms as features, described in Sec. 4.1.3. The main idea is that for some classes of images (see Fig. 4.8), even such simple features can add to performance without adding to complexity.

Putting the pieces together, and taking logarithm to get around issues of machine precision, we can re-write Eq. 4.4 as a *logit*:

$$\begin{aligned} \log \ell_{w_j}(I) = & \log \frac{Pr(A_{w_j}=1 | G_{w_j}=g_j)}{1 - Pr(A_{w_j}=1 | G_{w_j}=g_j)} + \sum_{i \neq j} \log \frac{Pr(G_{w_i}=g_i | A_{w_j}=1, G_{w_j}=g_j)}{Pr(G_{w_i}=g_i | A_{w_j}=0, G_{w_j}=g_j)} \\ & + \log \frac{Pr(h_1, \dots, h_D | A_{w_j}=1)}{Pr(h_1, \dots, h_D | A_{w_j}=0)} \end{aligned} \quad (4.6)$$

This logit is essentially the backbone of PLMFIT inference, where a higher value for a tag indicates greater support for its selection, for the image I . The final tags for image I can then be based on any of the following selection methods:

- **Top r :** After ordering all words $w_j \in V$ in the increasing magnitude of $\log \ell_{w_j}(I)$ to obtain a rank ordering, we annotate I using the top r ranked words.
- **Threshold $r\%$:** We can annotate I by thresholding at the top r percentile of the range of $\log \ell_{w_i}(I)$ values for the given image over all the words.
- **User Average:** For the personalization case, since there may be trends in the number of tags a particular user’s images have, we can compute the average number r of tags in a given user’s training samples, and predict top r tags for that user’s test cases.

In our experiments reported in Sec. 4.3, we have used all three methods and shed light on their performance differences, wherever applicable.

4.1.3 Model Estimation

For a given image I , predictions are first made by the black-box annotation system, which are then used for evaluating Eq. 4.6, to finally propose a set of tags. To facilitate evaluation of the equation, the probability terms need to be estimated using training data that mimics an ITE in question. These terms are estimated and indexed on, e.g., variables A_{w_j} and G_{w_j} , for efficient evaluation. Let us consider the estimation of each term separately, given a training set of size L for a particular ITE, consisting of images $\{I^{(1)}, \dots, I^{(L)}\}$, the corresponding tags guessed by the black-box, $\{f_{bbox}(I^{(1)}), \dots, f_{bbox}(I^{(L)})\}$, and the actual ground-truth tags, $\{f_{gtruth}(I^{(1)}), \dots, f_{gtruth}(I^{(L)})\}$. To make the system lightweight, all estimation is based on empirical frequencies, making computation times deterministic.

The term $Pr(A_{w_j} = 1 \mid G_{w_j} = g_j)$ in Eq. 4.6 can be estimated from the size L training data as follows:

$$\widehat{Pr}(A_{w_j}=1 \mid G_{w_j}=g_j) = \frac{\sum_{n=1}^L \mathcal{I}\{G_{w_j}^{(n)}=g_j \ \& \ A_{w_j}^{(n)}=1\}}{\sum_{n=1}^L \mathcal{I}\{G_{w_j}^{(n)}=g_j\}} \quad (4.7)$$

Here, $\mathcal{I}(\cdot)$ is the indicator function. A natural issue of robustness arises when the training set contains few or no samples for $G_{w_j}^{(n)}=1$. Therefore, we perform interpolation-based smoothing using

$$\widetilde{Pr}(A_{w_j}=1 \mid G_{w_j}=g_j) = \begin{cases} \widehat{Pr}_{prior}(g_j) & m \leq 1 \\ \frac{1}{m} \widehat{Pr}_{prior}(g_j) + \frac{m}{m+1} \widehat{Pr}(A_{w_j}=1 \mid G_{w_j}=g_j) & m > 1 \end{cases}$$

where $m = \sum_{n=1}^L \mathcal{I}\{G_{w_j}^{(n)}=g_j\}$, the number of instances out of L where w_j is guessed (or not guessed, depending upon g_j), and the prior $\widehat{Pr}_{prior}(g_j)$ is estimated using

$$\widehat{Pr}_{prior}(g_j) = \frac{\sum_{i=1}^K \sum_{n=1}^L \mathcal{I}\{G_{w_i}^{(n)}=g_j \ \& \ A_{w_i}^{(n)}=1\}}{\sum_{i=1}^K \sum_{n=1}^L \mathcal{I}\{G_{w_i}^{(n)}=g_j\}} \quad (4.8)$$

where $g_j \in \{0, 1\}$. The prior term needs some explanation. For $g_j = 1$, $\widehat{Pr}_{prior}(g_j) = \frac{\text{\#correct tag predictions overall}}{\text{\#tag predictions overall}}$. Thus, this prior stands for the probability that whenever an arbitrary tag is predicted, what the chances are of it being correct. Similarly, for $g_j = 0$, the prior stands for the probability that an arbitrary tag, when not guessed, is actually correct. Given the total lack of training samples for a tag, these priors are optimal estimates.

The probability term $Pr(G_{w_i}=g_i|A_{w_j}=1, G_{w_j}=g_j)$ in Eq. 4.6 can be estimated using the empirical frequency ratio

$$\widehat{Pr}(G_{w_i}=g_i|A_{w_j}=1, G_{w_j}=g_j) = \frac{\sum_{n=1}^L \mathcal{I}\{G_{w_i}^{(n)}=g_i \ \& \ G_{w_j}^{(n)}=g_j \ \& \ A_{w_j}^{(n)}=1\}}{\sum_{n=1}^L \mathcal{I}\{G_{w_j}^{(n)}=g_j \ \& \ A_{w_j}^{(n)}=1\}} \quad (4.9)$$

In this case, robustness is more critical, because many word pairs may appear together very infrequently among the black-box's guesses. Here, we describe and justify using the knowledge base WordNet [190] for robust smoothing of these probability estimates. If the vocabulary consists only of semantically meaningful tags such that WordNet-based word relatedness metrics are defined for all word pairs within it, we can take advantage of it to perform a different form of inductive transfer. Similarity based smoothing [53], a method commonly used in word pair co-occurrence modeling, is appropriate here. Given a WordNet-based similarity measure $W(w_i, w_j)$ between w_i and w_j , we smooth the frequency based estimates as follows.

$$\begin{aligned} \widetilde{Pr}(G_{w_i}=g_i|A_{w_j}=1, G_{w_j}=g_j) &= \theta \cdot \widehat{Pr}(G_{w_i}=g_i|A_{w_j}=1, G_{w_j}=g_j) \\ &+ (1 - \theta) \cdot \sum_{k \neq j} \frac{W(w_j, w_k)}{Z} \widehat{Pr}(G_{w_i}=g_i|A_{w_k}=1, G_{w_k}=g_k) \end{aligned} \quad (4.10)$$

where Z is the normalization factor. A choice of $\theta = 0.5$ ensures that a particular smoothed estimate is 50% dependent on the original estimate, and is found to work well in our experiments. The remainder of the contribution comes from other tags weighted by semantic relatedness. In effect, for poorly estimated terms, probability estimates over semantically related terms ‘substitute’ for each other.

Let us elaborate further on the use of WordNet for smoothing, and its role in inductive transfer. The function $W(\cdot, \cdot)$ stands for the Leacock and Chodorow (LCH)

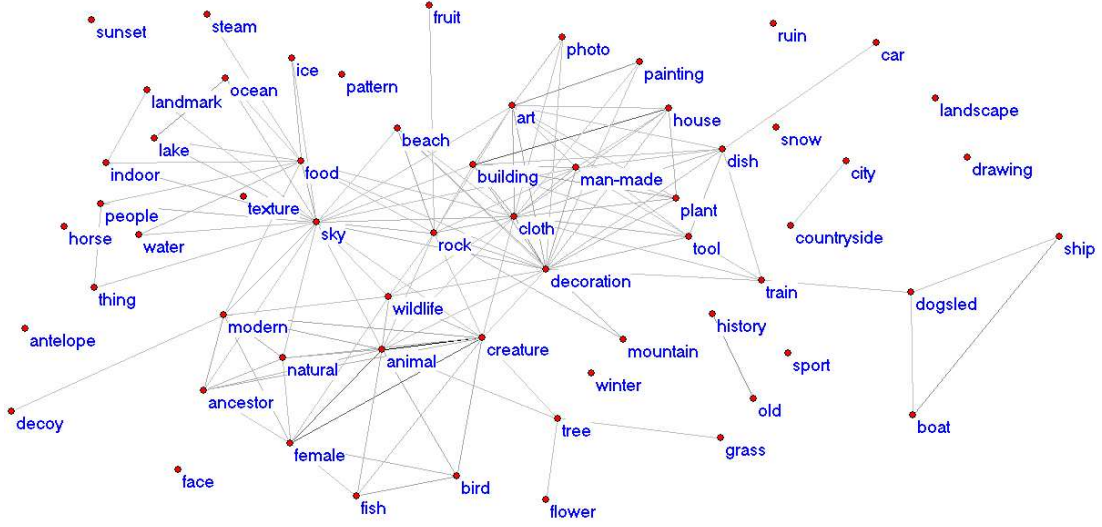


Figure 4.4. Network depicting WordNet-based relations among the 60 most frequently occurring tags in the Alipr dataset (Sec. 4.3). An edge between a pair of words indicates that the relatedness measure LCH [153] exceeds 1.7, roughly the mid-point of its $[0.37, 3.58]$ range of values.

word relatedness measure [153], which takes values between 0.37 and 3.58, higher value meaning more semantically related. If we defined a pair of words w_i and w_j to be semantically related if $W(w_i, w_j) \geq 1.7$, then we would get a relatedness network among the most frequently occurring tags in the Alipr dataset as shown in Fig. 4.4. We observe that while some relationships make little sense, much of the network is meaningful, and hence the LCH measure can be generally trusted. The role played by this measure, based on Eq. 4.10, is to effectively transfer learned knowledge (estimates) among semantically related tags. To further validate that such knowledge transfer is practical, we reverse-engineered the problem by computing the averaged out absolute differences between the pre-smoothed empirical probability estimates among tags, and using a threshold to connect tag pairs with low average differences. The resultant network is shown in Fig. 4.5. Though not depicting the same underlying aspects, comparison with the network in Fig. 4.4 reveals interesting overlaps. In Fig. 4.6, we present a more direct attempt at assessing whether the use of semantic relatedness leads to better ‘substitution’ of

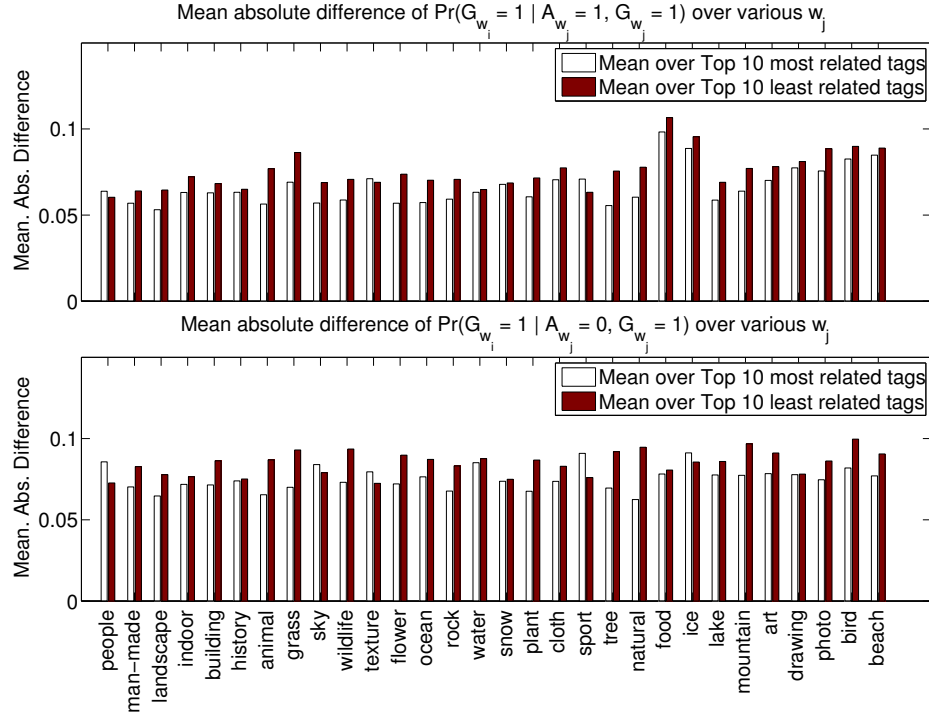


Figure 4.6. Empirical differences between estimated $\Pr(G_{w_i}=1|A_{w_j}=a, G_{w_j}=1)$ when using semantically related words as against unrelated ones, are shown. For each of the 30 most frequent tags w_i in Alipr dataset, we compute $y(w_i)$ (defined in Eq. 4.11 in the text). What we see is that out of 30 tags, 25 and 27 tags for $a = 0$ and $a = 1$ respectively have better estimates of the probability terms when substituted with semantically related terms, as against unrelated ones. This indicates that smoothing with relatedness weights is an attractive strategy.

Finally, the parameters of $\Pr(h_1, \dots, h_D \mid A_{w_j}=a)$, $a \in \{0, 1\}$, which models how simple visual features of image I (external to the black-box) differ when w_j is correct or incorrect, are estimated. The goal is to allow PLMFIT the opportunity to be directly influenced by visual features which can be efficiently and incrementally computed. Note that PLMFIT is already indirectly affected by visual aspects of I via the black-box. A formulation that we successfully incorporated into PLMFIT is now described. Each image is divided into 16 equally spaced, non-overlapping, orthogonal tiles. For each tile, the RGB color values are transformed into the LUV space, and the triplet of average L , U , and V values represent that block. Thus, each image is represented by a 48-dimensional vector (h_1, \dots, h_{48}) of global visual features (see Fig. 4.7). For estimation, each of the 48 components

are fitted with univariate Gaussians, which involves calculating the sample mean $\hat{\mu}_{j,d,a}$ and std. dev. $\hat{\sigma}_{j,d,a}$ over training data with $a = \{0, 1\}$ on tag w_j . As before, smoothing is performed by interpolation with priors $\tilde{\mu}_{d,a}$ and $\tilde{\sigma}_{d,a}$ estimated over all tags. The joint probability is computed by treating each component conditionally independent given w_j :

$$\widetilde{Pr}(h_1, \dots, h_{48} \mid A_{w_j}=a) = \prod_{d=1}^{48} \mathcal{N}(h_d \mid \hat{\mu}_{j,d,a}, \hat{\sigma}_{j,d,a}) \quad (4.12)$$

Here, $\mathcal{N}(\cdot)$ denotes the Gaussian density function. Again, to provide intuition behind the role of these ratios in PLMFIT, we present $\hat{\mu}_{j,d,a}$ values for two exemplary cases, estimated with real-world data, in Fig. 4.8. With this, we have covered the estimation of each term in the PLMFIT model. We continue discussion on the application of this static meta-learning model to dynamic scenarios.

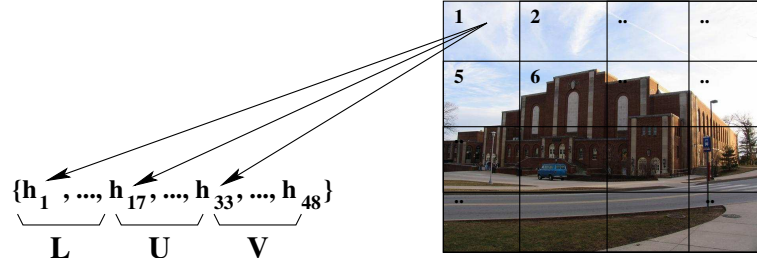


Figure 4.7. The 48-dimensional *LUV* features extracted in PLMFIT.

4.2 Tagging Improvements with PLMFIT

With the PLMFIT fundamentals and estimation steps described, we proceed to discuss algorithms and setups for adaptation. As discussed before, three settings where PLMFIT can be applied for adaptation purposes are (1) Context, (2) Time evolution, and (3) Personalization. Contextual adaptation, which is achieved essentially by batch-learning the PLMFIT using a sample set, is discussed first. The latter two entail new algorithms that employ PLMFIT, and therefore these are discussed in more details in following subsections.

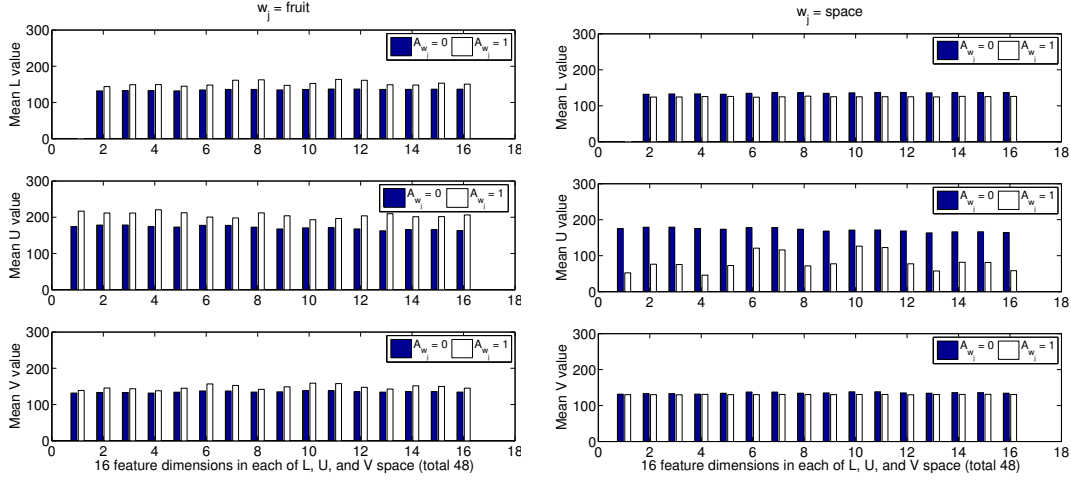


Figure 4.8. Estimated values of $\hat{\mu}_{j,d,a}$, model parameters for $Pr(h_1, \dots, h_{48} \mid A_{w_j}=a)$, for the 48-dimensional global image features for two tags with observed differences, are shown. As in the case of ‘space’ and ‘fruit’, if differences are significant for $A_{w_j} = 0$ and 1, then this ratio contributes to the inference. An intuition behind the difference in case of ‘fruit’, for example, is that close-up shots of fruits tend to be brighter and more colorful than is typical.

4.2.1 Contextual Adaptation of Tagging

Contextual adaptation, with reference to automatic image tagging, is the application of image annotation models, trained with data generated in one ITE (image tagging environment), to a different ITE. This includes cases where (a) the test conditions vary only in sample selection (e.g., trained using one set of Corel images, and tested on a different set of Corel images), or (b) the test data is from an entirely different source as compared to the training data (e.g, training using Corel, testing on Flickr images). In this sense, the context is not changing on a continual basis, but is rather a one-time shift.

The underlying black-box model is trained using images from say ITE_1 . The test scenario involves samples from ITE_2 which could represent either of the above mentioned cases. We assume the availability of N tagged training samples from ITE_2 , which we use to estimate the PLMFIT meta-learner described in Sec. 4.1.3. For example, if the underlying black-box is Alipr [166] trained using Corel images (ITE_1), and photo-sharing site Flickr (ITE_2) is our target application, we can use some user-tagged Flickr images to train PLMFIT, and then used the

Alipr/PLMFIT combination to tag future images on Flickr. If the combination performed better than Alipr alone, it would be convincing that the meta-learning layer indeed adds to tagging performance. Empirical assessment of contextual adaptation strength of PLMFIT is presented in Sec. 4.3.1.

4.2.2 Adapting Tagging over Time

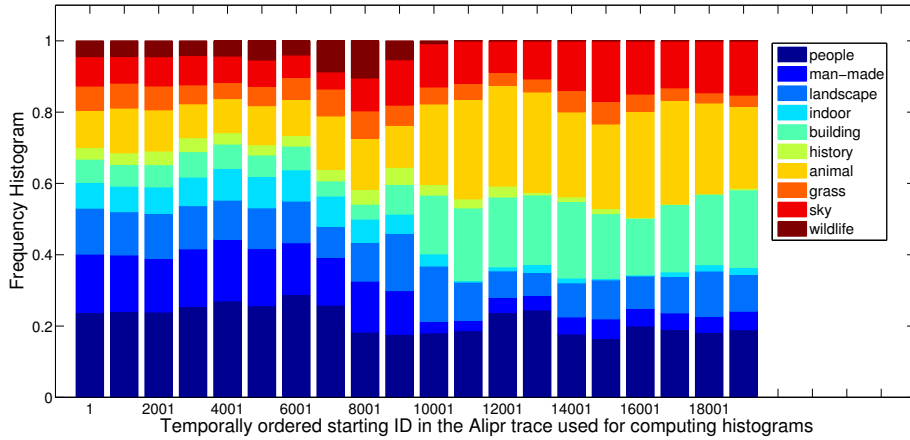


Figure 4.9. Time-ordered histograms of occurrence of the top 10 most frequent tags in the Alipr dataset (consisting of 20,000 images), computed over 2,000 image overlapping windows (except last one) with window starting points at 1,000 image intervals. Notice how tag popularity fluctuate over time, e.g., after a point, ‘wildlife’ diminishes in frequency while ‘animal’ gains prominence.

When an annotation black-box is deployed in an online environment such as Alipr or Flickr, where there is continuous image uploading and tagging over time, the user base, the kinds of photos uploaded by them, and the types of tags given to them may evolve with time (see evidence of this in Fig. 4.9). External impetus such as news and current affairs may further protract this [57]. To deal with this issue, we can employ PLMFIT to meta-learn and adapt itself as things change. However, continuously re-training the full model is computationally intensive. In this section we present algorithms that help adapt over time efficiently. A schematic view of this scenario and our approach to handling it is shown in Fig. 4.10.

In Flickr, images are publicly uploaded, and independently or collaboratively tagged, not necessarily at the time of uploading. In Alipr, feedback is solicited

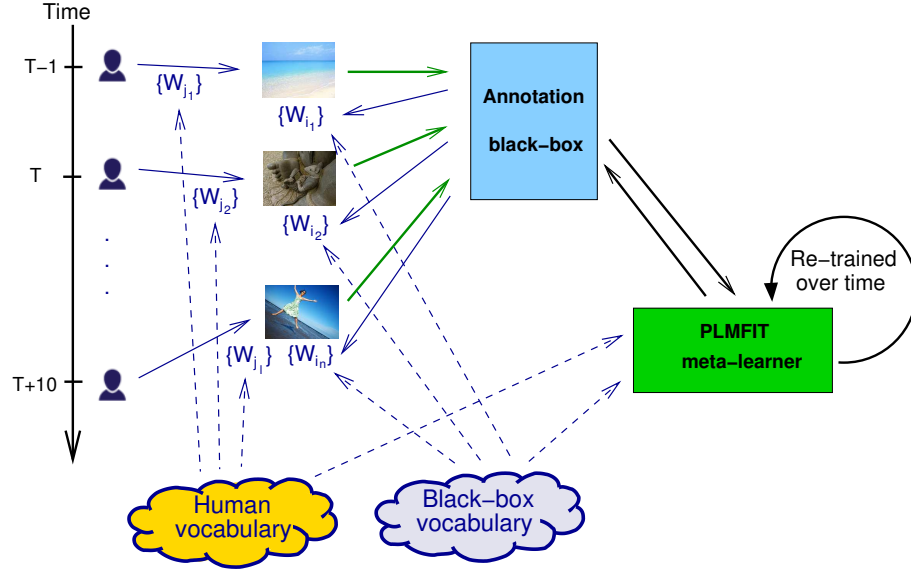


Figure 4.10. Tagging adaptation over time using a black-box augmented with PLMFIT.

immediately upon uploading. In both these cases, ground-truth arrives into the system sequentially, giving an opportunity to learn from it to tag future uploads better. As described in Sec. 4.1.3, PLMFIT estimation is purposely designed to involve summation of instances only, followed by $O(1)$ parameter computation. Inference steps are also lightweight in nature. We can take advantage of this to perform incremental/decremental learning, thereby eliminating the need for full-fledged re-estimation over time.

To start with, the PLMFIT needs to be estimated with seed images taken from the application ITE. Hence, over a certain initial period, the meta-learner stays inactive, collecting an L_{seed} number of user-tagged images. At this point, the meta-learner is trained, and starts tagging incoming images. After an L_{inter} number of new images has been received, the meta-learner is re-trained (see Fig. 4.11). The primary challenge here is to make use of the models already learned, so as not to redundantly train on the same data. Re-training can be of two types depending on the underlying ‘memory model’:

- **Persistent Memory:** Here, PLMFIT accumulates new data into the current model, so that at steps of L_{inter} , it learns from all data since the very beginning, inclusive of the seed data. Technically, this only involves incremental learning.

- **Transient Memory:** Here, while the model learns from new data, it also ‘forgets’ an equivalent amount of the earliest memory it has. Technically, this involves incremental and decremental learning, whereby at every L_{inter} jump, PLMFIT is updated by (a) assimilating new data, and (b) ‘forgetting’ old data.

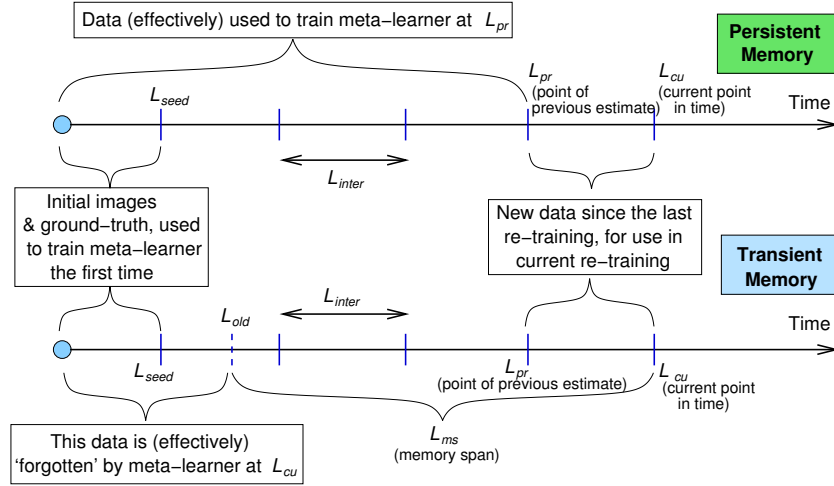


Figure 4.11. Overview of persistent/transient memory models for tagging adaption over time.

4.2.2.1 Incremental/Decremental Meta-Learning

The PLMFIT formulation makes incremental and decremental learning efficient. Let us denote ranges of image sequence indices, ordered by time, using the superscript $[start : end]$, and let the index of the current image be L_{cu} . We first discuss incremental learning, required for the case of *persistent memory*. Here, probabilities are re-estimated over all available data upto the current time, i.e., over $[1 : L_{cu}]$. This is done by maintaining *summation* terms, denoted $\mathcal{S}(\cdot)$, computed in the most recent re-training at L_{pr} (say), over a range $[1 : L_{pr}]$, where $L_{pr} < L_{cu}$. For the first term in Eq. 4.6, suppressing the irrelevant variables,

$$\begin{aligned}
 \widehat{Pr}(A_{w_j} | G_{w_j})^{[1:L_{cu}]} &= \frac{\sum_{n=1}^{L_{cu}} \mathcal{I}\{G_{w_j}^{(n)} \& A_{w_j}^{(n)}\}}{\sum_{n=1}^{L_{cu}} \mathcal{I}\{G_{w_j}^{(n)}\}} \\
 &= \frac{\mathcal{S}(G_{w_j} \& A_{w_j})^{[1:L_{pr}]} + \sum_{n=L_{pr}+1}^{L_{cu}} \mathcal{I}\{G_{w_j}^{(n)} \& A_{w_j}^{(n)}\}}{\mathcal{S}(G_{w_j})^{[1:L_{pr}]} + \sum_{n=L_{pr}+1}^{L_{cu}} \mathcal{I}\{G_{w_j}^{(n)}\}} \quad (4.13)
 \end{aligned}$$

Therefore, updating and maintaining summation values $\mathcal{S}(G_{w_j})$ and $\mathcal{S}(G_{w_j} \& A_{w_j})$ suffices to re-train PLMFIT without using time/space on past data. The *priors* are also computed using these summation values in a similar manner, for smoothing. Since PLMFIT is re-trained at fixed intervals of L_{inter} , i.e., $L_{inter} = L_{cu} - L_{pr}$, only a fixed amount of time/space is required every time for getting the probability estimates, regardless of the value of L_{cu} . The second term in Eq. 4.6 can also be estimated in a similar manner, by maintaining the summations, taking their quotient, and smoothing with re-estimated priors. For the last term related to visual features, the estimated mean $\hat{\mu}_{j,d,a}$ and std.dev. $\hat{\sigma}_{j,d,a}$ can also be updated with values of (h_1, \dots, h_{48}) for the new images by only storing summation values. Since $\sigma^2(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2$,

$$\hat{\mu}_{j,d,a}^{[1:L_{cu}]} = \frac{1}{L_{cu}} \left(\mathcal{S}(h_d)^{[1:L_{pr}]} + \sum_{n=L_{pr}+1}^{L_{cu}} h_d^{(n)} \right)$$

$$\hat{\sigma}_{j,d,a}^{[1:L_{cu}]} = \sqrt{\frac{1}{L_{cu}} \left(\mathcal{S}(h_d^2)^{[1:L_{pr}]} + \sum_{n=L_{pr}+1}^{L_{cu}} (h_d^{(n)})^2 \right) - \left(\hat{\mu}_{j,d,a}^{[1:L_{cu}]} \right)^2}$$

Here, $\mathcal{S}(h_d^2)^{[1:L_{pr}]}$ is the sum-of-squares of the past values of feature h_d , to be maintained, and $\mathbf{E}(\cdot)$ denotes expectation. This justifies the simple visual representation we have, since it becomes convenient for incremental learning. Overall, this process continues to re-train PLMFIT, using the past summation values, and updating them at the end, as depicted in Fig. 4.11.

In the *transient memory* model, estimates need to be made over a fixed number of the most recent data instances. This can also be performed efficiently by combining incremental with decremental learning. We can again maintain summation values, but here we need to *subtract* the portion that is to be removed from consideration. Suppose the *memory span* is decided to be L_{ms} , meaning that at the current time L_{cu} , the model estimate must only be based on data over the range $[L_{cu} - L_{ms} : L_{cu}]$. Let $L_{old} = L_{cu} - L_{ms}$. Here, we show the re-estimation of $\hat{\mu}_{j,d,a}$.

Along with summation $\mathcal{S}(h_d)^{[1:L_{pr}]}$, we also need $\mathcal{S}(h_d)^{[1:L_{old}-1]}$. Therefore,

$$\hat{\mu}_{j,d,a}^{[L_{old}:L_{cu}]} = \frac{1}{L_{ms}+1} \sum_{n=L_{old}}^{L_{cu}} h_d^{(n)} = \frac{1}{L_{ms}+1} \left(\mathcal{S}(h_d)^{[1:L_{pr}]} + \sum_{n=L_{pr}+1}^{L_{cu}} h_d^{(n)} - \mathcal{S}(h_d)^{[1:L_{old}-1]} \right)$$

Since L_{ms} and L_{inter} are decided *a priori*, we can pre-compute the values of L_{old} for which $\mathcal{S}(h_d)^{[1:L_{old}-1]}$ will be required, and store them along the way. Other terms in Eq. 4.6 can be estimated similarly.

Algorithm 1 Adapting Tagging over Time with PLMFIT

Require: Image stream, Black-box, tagged image pool (seed)
Ensure: Annotation guesses for each new image
1: /* Learn an initial seed model using available tagged data */
2: Train PLMFIT using seed data.
3: **repeat** { $I \leftarrow$ incoming image}
4: Annotate I using PLMFIT
5: **if** User tags image I **then**
6: $L_{cu} \leftarrow L_{cu} + 1$, $I_{L_{cu}} \leftarrow I$
7: $\text{Dat}(L_{cu}) \leftarrow$ Black-box guesses, user tags, etc.
8: **end if**
9: **if** $((L_{cu} - L_{seed}) \text{ modulo } L_{inter}) = 0$ **then**
10: **if** Strategy = ‘Persistent Memory’ **then**
11: Re-train PLMFIT on $\text{Dat}(1 : L_{cu})$
12: /* Use incremental learning for efficiency */
13: **else**
14: Re-train PLMFIT on $\text{Dat}(L_{cu} - L_{ms} : L_{cu})$
15: /* Use incremental/decremental learning for efficiency */
16: **end if**
17: **end if**
18: **until** End of time

In summary, a high-level version of tagging adaptation over time is presented in Algo. 1 starts with an initial training of PLMFIT using seed data of size L_{seed} . This could be accumulated online using the annotation system itself, or from an external source of images with ground-truth (e.g., Corel images). The process then takes one image at a time, annotates it, and when ground-truth is made available, it is stored for future meta-learning. After gaps of l_{inter} , the model is re-trained based on one of the two chosen strategies.

4.2.3 Personalized Tagging across People

In environments such as Flickr, where image tagging is typically performed by the owner and her connected relations, there is an opportunity for automatic annotation systems to personalize the tagging process in order to improve performance.

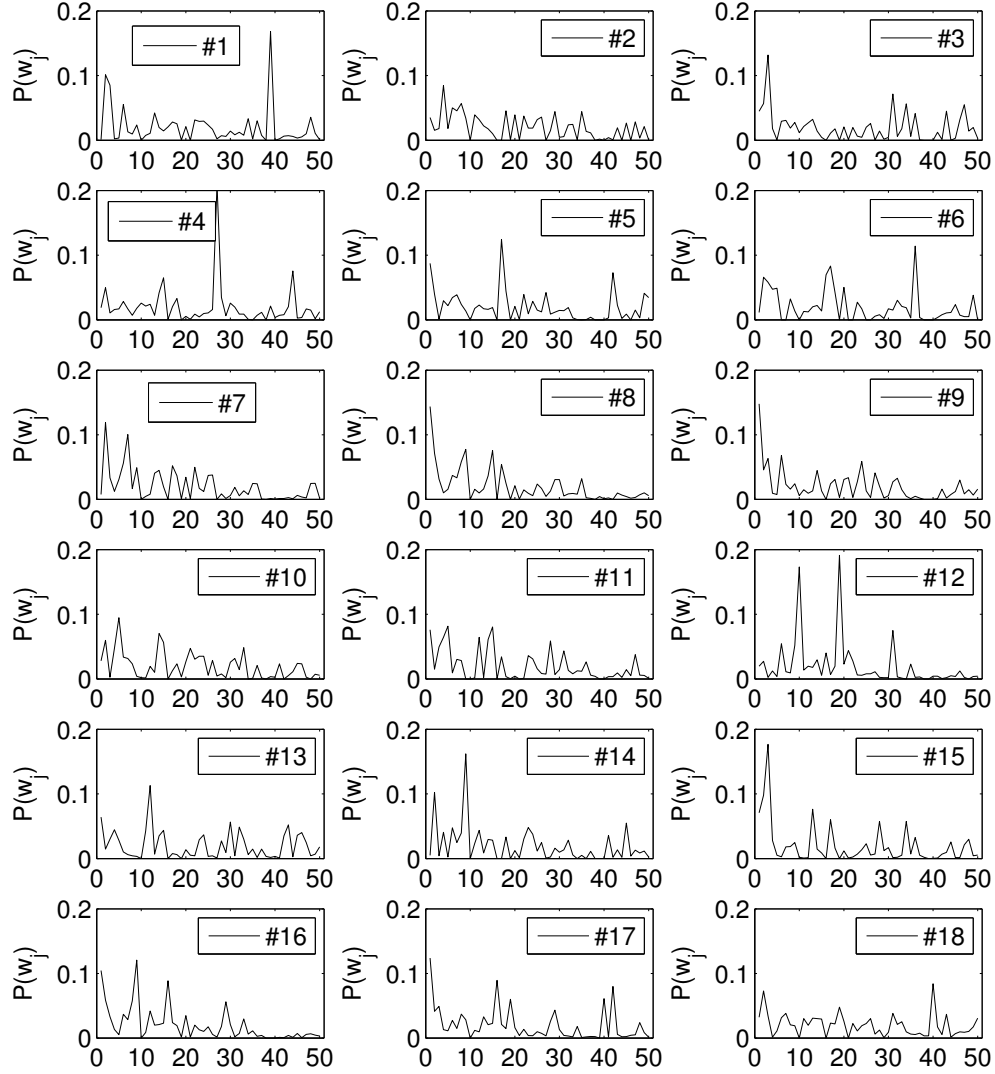


Figure 4.12. Distribution over the 50 most frequent tags in the Flickr dataset (see Sec. 4.3) for 18 randomly sampled users. Note how the distribution greatly varies, which reinforces our belief that personalization can give automatic image tagging a significant performance boost.

Different users upload different kinds of images and may also follow different tagging patterns. In fact, from Fig. 4.12, we can see that the distribution over the set of 50 most frequently occurring tags in the Flickr dataset varies greatly across

users. To build personalization models, the following must be assumed; (a) there are tagged images available for the user in question, and (b) there is locality in the tag space of images belonging to the user, i.e., the tag distribution of an user's images does not represent a microcosm of the entire collection of images, but rather something characteristic of that particular user.

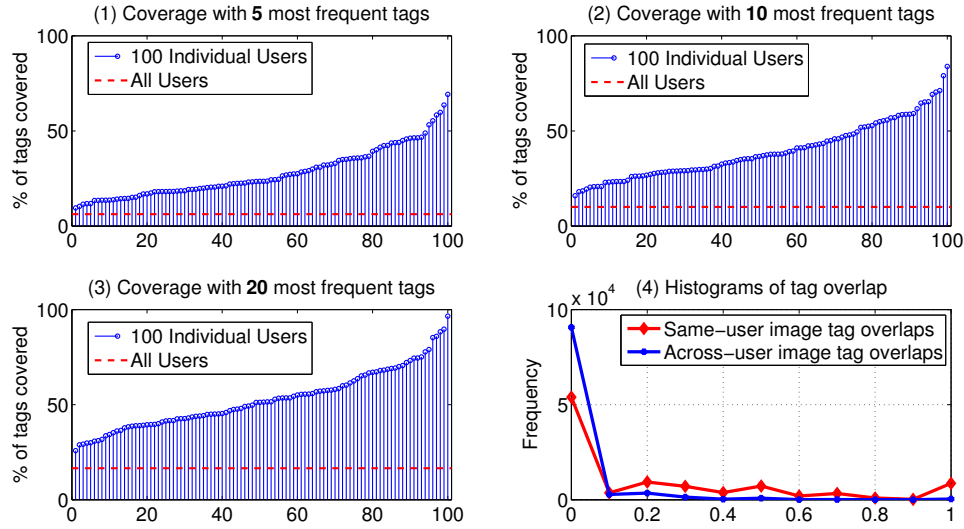


Figure 4.13. Motivating the case for personalization with Flickr data: Graph (1)-(3) depict the fraction of all tags covered by the most frequent 5, 10, and 20 tags respectively for each of 100 randomly chosen users (sorted by % covered). The dashed line shows the same for all users pooled together, providing evidence that the tags space is localized for most users. Graph (4) shows the distribution of overlap between tags of 100,000 image pairs each sampled randomly from (a) within same users, and (b) across users, normalized by the minimum number tags for each pair. While more than 90% of the across-user cases have no tag overlaps, almost 50% of the same-user pairs have some overlap, while 8%+ have maximally identical tags.

In Fig. 4.13, data obtained from Flickr shows strong evidence of the validity of these assumptions, focusing on (a) tag space locality, and (b) within-user tag similarity compared to across-user tag dissimilarity. One personalization approach could be that a separate black-box model is trained for every single individual. Given millions of users, who at a given time have a varying number of tagged images associated with them, this approach runs the risk of (a) being prohibitively expensive, (b) requiring re-training as more images are tagged, and (c) lacking sufficient data points in many cases. Instead, we can employ PLMFIT to personalize

Algorithm 2 Personalized Tagging across People with PLMFIT

Require: Black-box, Tagged image pool from population (seed)
Require: Previously tagged set of images for user U , population vocabulary V^{pop}
Ensure: Personalized tagging model $PLMFIT^U$ for U , with vocabulary V^U

- 1: /* Learn a seed model over the entire population */
- 2: Train $PLMFIT_{pop}$ using seed data (once for all users)
- 3: $PLMFIT^U \leftarrow PLMFIT_{pop}$, $V^U \leftarrow V^{pop}$
- 4: $\mathcal{Q}^U \leftarrow$ Tagged image pool of user U
- 5: $W \leftarrow \{\text{user tags for } \mathcal{Q}^U\} \cup \{PLMFIT^U \text{ tag predictions for } \mathcal{Q}^U\}$
- 6: **repeat** $\{w_k \in W, \text{ taken in arbitrary order}\}$
- 7: **if** $(w_k \in V^U)$ **then**
- 8: /* Tag is in the vocabulary, so update model */
- 9: Perform **incremental update** of $PLMFIT^U$ using relevant subset of \mathcal{Q}^U
- 10: **else**
- 11: /* Tag is not in current vocabulary, so add to model */
- 12: $V^U \leftarrow V^U \cup \{w\}$
- 13: Perform **vocabulary expansion** of $PLMFIT^U$ using relevant subset of \mathcal{Q}^U
- 14: **end if**
- 15: **until** all tags in W are covered

the tags in a lightweight manner, effectively using a prior model over all individuals, and incrementally incorporating new data points per individual by incurring low overhead. Algo. 2 presents a sketch of the personalization algorithm.

4.2.3.1 Incremental Update

In Algo. 2, when a tag w_k associated with user pool \mathcal{Q}^U is already part of the vocabulary V^{pop} , the personalization step is to update the parameters in model $PLMFIT^U$ that are associated with w_k . This can be done incrementally in a fashion very similar to Eq. 4.13 as described in Sec. 4.2.2. As before, summation values need to be maintained. The only difference is that instead of pooling over time, we pool tagged images specifically for user U . Suppose $\mathcal{Q}^U = \{I_{U_1}, \dots, I_{U_n}\}$, then $Pr(A_{w_k} | G_{w_k})^U$, the first term of $PLMFIT^U$ is estimated as

$$\widehat{Pr}(A_{w_k} | G_{w_k})^U = \frac{\mathcal{S}(G_{w_k} \& A_{w_k})^{pop} + \sum_{q=1}^n \mathcal{I}\{G_{w_k}^{(q)} \& A_{w_k}^{(q)}\}}{\mathcal{S}(G_{w_k})^{pop} + \sum_{q=1}^n \mathcal{I}\{G_{w_k}^{(q)}\}} \quad (4.14)$$

where superscripts pop and U denote population and user U specific terms respectively, and $\mathcal{S}(\cdot)$ denotes summation terms, as before. The other terms in Eq. 4.6 can also be updated in a similar manner, as mentioned in Sec. 4.2.2. Intuitively, a larger size of the pool \mathcal{Q}^U should have greater influence on $PLMFIT$, and hence the personalization should be more effective for users with more tagged images available.

4.2.3.2 Vocabulary Expansion

When a tag w_k associated with \mathcal{Q}^U is not part of V^{pop} , it needs to be added to vocabulary V^U for user U , and corresponding probability terms need to be estimated from scratch. The first term $Pr(A_{w_k} | G_{w_k})^U$ of Eq. 4.6 is essentially $Pr(A_{w_k})^U$, the prior on w_k , since $G_{w_k} = 0$, and the last term $Pr(h_1, \dots, h_{48} | A_{w_j})^U$ can be estimated using Eq. 4.12, since the settings are identical. Estimation of probabilities $Pr(G_{w_i} = g_i | A_{w_j} = a_j, G_{w_j} = g_j)^U$, which make up the second term, needs more attention. We need to generate new estimates for terms of the form $Pr(G_{w_k} = g_k | A_{w_j} = a_j, G_{w_j} = g_j)^U$ and $Pr(G_{w_i} = g_i | A_{w_k} = a_k, G_{w_k} = g_k)^U$, for all $w_i, w_j \in V^{pop}$. For a new tag w_k , $G_{w_k} = 0$ in all cases, so we can re-write the corresponding ratio terms as follows:

$$\begin{aligned} \frac{Pr(G_{w_k}=g_k | A_{w_j}=1, G_{w_j}=g_j)}{Pr(G_{w_k}=g_k | A_{w_j}=0, G_{w_j}=g_j)} &= 1, \text{ and} \\ \frac{Pr(G_{w_i}=g_i | A_{w_k}=1, G_{w_k}=g_k)}{Pr(G_{w_i}=g_i | A_{w_k}=0, G_{w_k}=g_k)} &= \frac{Pr(G_{w_i}=g_i | A_{w_k}=1)}{Pr(G_{w_i}=g_i | A_{w_k}=0)} \end{aligned} \quad (4.15)$$

In plain words, this means that (a) new tags play no role in the prediction of the tags in the black-box vocabulary, and (b) since the new tags are never guessed by the black box, their prediction is based entirely upon guesses on the black-box vocabulary. This also shows that in Algo. 2, the order in which the new words are added to the vocabulary does not matter. To summarize, for a new word w_k , the logit in Eq. 4.6 can be simplified as follows:

$$\begin{aligned} \log \ell_{w_k}(I) &= \log \frac{Pr(A_{w_k}=1)}{1 - Pr(A_{w_k}=1)} + \sum_{i \neq k} \log \left(\frac{Pr(G_{w_i}=g_i | A_{w_k}=1)}{Pr(G_{w_i}=g_i | A_{w_k}=0)} \right) \\ &+ \log \left(\frac{Pr(h_1, \dots, h_D | A_{w_k}=1)}{Pr(h_1, \dots, h_D | A_{w_k}=0)} \right) \end{aligned} \quad (4.16)$$

Vocabulary expansion in this manner is general enough that it can apply to tagging adaptation over time without any personalization. A combination of efficient adaptation over time and user-specific personalization is a natural extension, but we have not experimented with such combinations in this work.

4.3 Experimental Results

We perform image tagging experiments to validate the effectiveness of PLMFIT in (1) contextual adaptation, (2) adaptation over time, and (3) personalization. Standard datasets as well as real-world data are used. First, we use the Corel Stock photos [163] to compare our meta-learning approach with the state-of-the-art. This collection of images is tagged with a 417 word vocabulary. Second, we obtain two real-world, temporally ordered traces from the Alipr Website [3], each 10,000 in length, taken over *different periods of time* in the year 2006. Each trace consists of publicly uploaded images, the automatic annotations provided by Alipr, and the tags provided by users for them. The Alipr system provides the user with 15 guessed tags, and the user can opt to select the correct guesses and/or add new ones. The vocabulary for this dataset consists of 329 unique tags. Third, using the Flickr API [88], we obtain upto 1000 public images belonging to each of 300 random users, totaling to 162,650 real-word images. After pruning tags that appeared less than 10 times in the entire dataset, we were left with a 2971 word vocabulary, with a mean of $7 (\pm 5)$ user tags per image, which we treat as ground-truth. As expected, we observed a great number of user-specific tags (e.g., names of people) in the dataset, although many of the less significant ones were eliminated in the aforementioned pruning process.

Two different *black-box* annotation systems, which use different algorithms for image tagging, are used in our experiments. A good meta-learner should fare well for different underlying black-box systems, which is what we set out to explore here. The first is Alipr [166], which is a real-time annotation system, and the second is a recently proposed approach [55] which was shown to outperform earlier algorithms. Both models generate tag guesses given an image, ordered by decreasing likelihoods. Annotation performance is gauged using three standard measures, namely *precision*, *recall* and *F-score* that have been used in the past. Specifically, for *each* image, $precision = \frac{\#(\text{tags guessed correctly})}{\#(\text{tags guessed})}$, $recall = \frac{\#(\text{tags guessed correctly})}{\#(\text{correct tags})}$, and $F\text{-score} = \frac{2 \times Precision \times Recall}{Precision + Recall}$ (harmonic mean of precision and recall). Results reported in each case are averages over all images tested with.

The ‘lightweight’ nature of PLMFIT is validated by the fact that the (re-)training of each visual category in [166] and [55] are reported as 109 and 106

Table 4.1. Contextual adaptation performance on 10,000 Corel images

Approach	Precision	% Change	Recall	% Change	F- score
Baseline [55]	1 in 4	-	2 in 5	-	31.3
PLMFIT (Top r)	1 in 3	+28%	3 in 4	+83%	45.2
PLMFIT (Threshold)	2 in 5	+59%	3 in 5	+50%	48.6

seconds respectively. Therefore, at best, re-training will take these times when the models are trained fully in parallel. In contrast, our meta-learner re-trains on 10,000 images in ~ 6.5 sec. on a single machine having equivalent configuration. Furthermore, the additional computation time due to the meta-learner during annotation is negligible.

4.3.1 Contextual Adaptation of Tagging

Our first set of experiment tests the notion that PLMFIT can adapt to contextual change, i.e., it can take a black-box annotation system trained on one type of sample and can improve its performance on a different dataset. In the work by [55], 24,000 Corel images, drawn from 600 image categories were used for training, and a separate 10,000 test images were used to assess performance. We use this system as black-box by obtaining the word guesses made by it, along with the corresponding ground-truth, for each image. Our meta-learner PLMFIT uses an additional $L_{seed} = 2,000$ images (randomly chosen, non-overlapping) from the Corel dataset as the *seed* data. Therefore, effectively, (black-box + PLMFIT) uses 26,000 instead of 24,000 images for training. We present results on this case in Table 4.1. The PLMFIT performance is shown for both **Top r** ($r = 5$) and **Threshold $r\%$** ($r=60$), as described in Sec. 4.1.2. The baseline results are those reported in [55]. Note the significant jump in performance with our meta-learner in both cases. Strictly speaking, this is not a contextual change, since the black-box was trained on Corel images and the testing was also done using Corel, although they were non-overlapping sets. This makes the results more surprising, in that simply adding PLMFIT to the mix makes a significant difference in performance.

Next, we perform an experiment that truly tests the contextual adaptation power of PLMFIT. Real-world images obtained from Alipr do not share the typical

Table 4.2. Contextual adaptation performance on 16,000 Alipr images

Approach	Precision	% Change	Recall	% Change	F- score
Baseline [166]	17 in 100	-	2 in 5	-	24.2
PLMFIT (Top r)	22 in 100	+28%	1 in 2	+18%	30.3
PLMFIT (Threshold)	1 in 3	+95%	3 in 5	+42%	48.6

characteristics of Corel images in terms of types of images uploaded and tags given to them. Therefore there is considerable challenge in using a Corel-trained black-box model to tag them, as will also become evident from the baseline performance presented here. We use both Alipr traces consisting of 10,000 images each. It turns out that given the Alipr Website, most people provided feedback by selection, and a much smaller fraction typed in new tags. As a result, the recall is by default very high for the black-box system, but it also yields poor precision. For each Alipr trace, our meta-learner is trained on $L_{seed} = 2,000$ *seed* images, and tested on the remaining 8,000 images. In Table 4.2, averaged-out results using PLMFIT for both **Top r** ($r = 5$) and **Threshold $r\%$** ($r=75$), as described in Sec. 4.1.2, are presented alongside the baseline [166] performance on the same data (top 5 guesses). Again we observe significant performance improvements over the baseline in both cases. As is intuitive, a lower percentile cut-off for threshold, or a higher number r of top words both lead to higher recall, at the cost of lower precision. Therefore, either number can be adjusted according to the specific needs of the application. Given these results, we can conclude that PLMFIT layer provides a statistically significant boost to image tagging performance of a black-box system under contextual changes.

4.3.2 Adapting Tagging over Time

We now test whether the PLMFIT layer is effective at adapting to changes over time or not. Because the Alipr data was generated by a real-world process with real users, it makes an apt dataset for this test. Again, the black-box here is the Alipr system, which provides guessed tags, and the Website users provide ground-truth tags. First, we experiment with the two data traces separately. For each trace, a seed data consisting of the first $L_{seed} = 1,000$ images (in temporal order) is used to

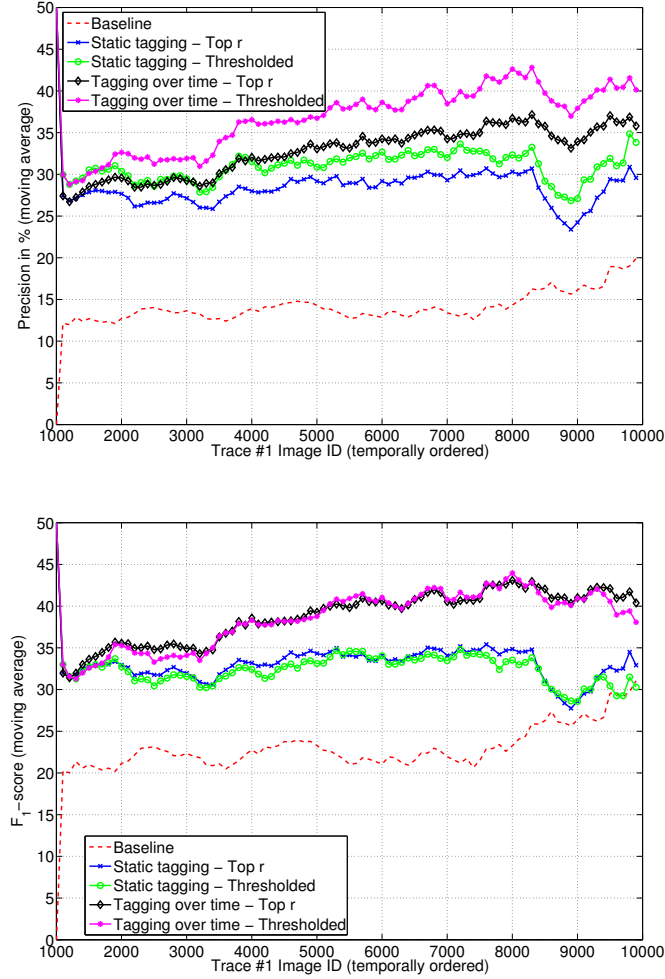


Figure 4.14. Performance (precision and F-score) of adaptation over time for Alipr trace #1.

initially train PLMFIT. Re-training is performed in intervals of $L_{inter} = 200$. We test on the remaining 9,000 images of the trace for (a) *static tagging* - PLMFIT is not further re-trained after seed training, and (b) *tagging over time* - PLMFIT is re-trained over time, using (a) **Top r** ($r = 5$), and (b) **Threshold r%** ($r=75$) in each case. For these experiments, the *persistent memory* model is used. Comparison is made using *precision* and *F-score*, with the baseline performance being that of Alipr, the black-box. These results are shown in Figs. 4.14 and 4.15. The scores shown are moving averages over 500 images (or less, in the case of the initial 500 images). We observe that seed training considerably boosts performance over the

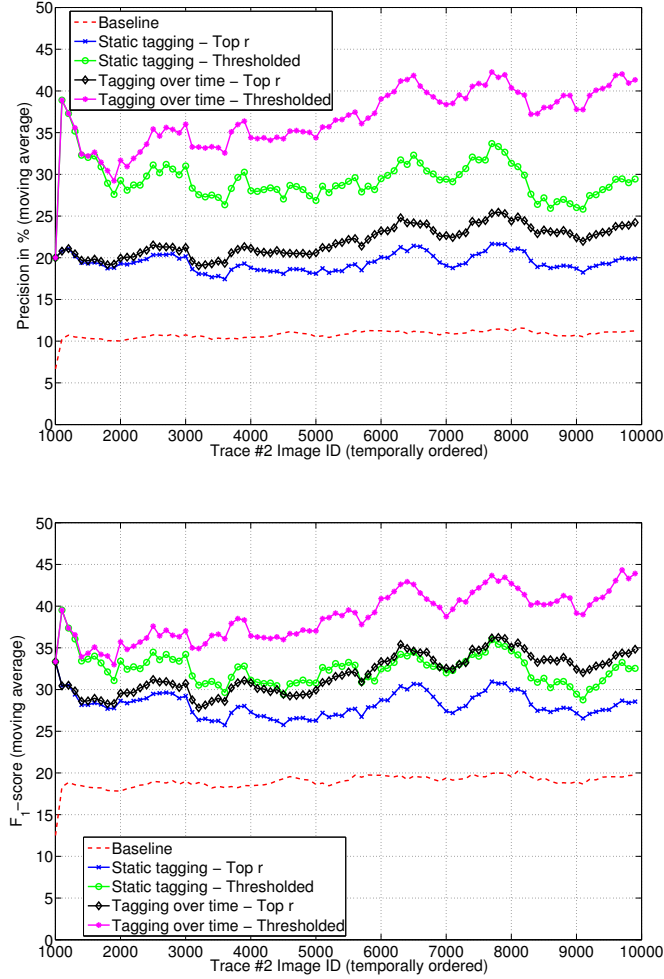


Figure 4.15. Performance (precision and F-score) of adaptation over time for Alipr trace #2.

baseline, and this performance keeps getting better over time.

Next, we explore how the *persistent* and *transient* memory models fare against each other. The main motivation for transient learning is to ‘forget’ earlier training data that may have become irrelevant, due to concept drift or otherwise. Because we observed such a shift between Alipr traces #1 and #2 (being taken over distinct time-periods), we merged them together to obtain a single 20,000 image trace to emulate a scenario of shifting trend in image tagging. Performing a seed learning over images 4,001 to 5,000 (part of trace #1), we test on the trace from 5,001 to 15,000. The results obtained using the two memory models, along with the static

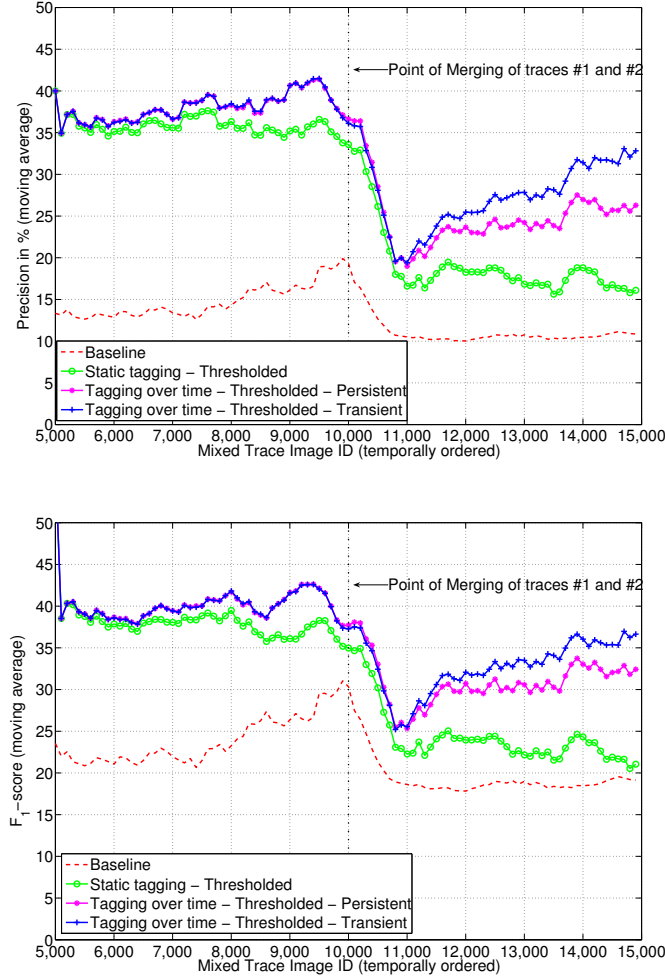


Figure 4.16. Comparison of precision and F-score for the two memory models of incremental learning, *persistent* and *transient*.

and baseline cases, are presented in Fig. 4.16. Observe the performance dynamics around the 10,000 mark where the two traces are merged. While the persistent and transient models follow each other closely till around this mark, the latter performs better after it (by upto 10%, in precision), verifying our hypothesis that under significant changes over time, ‘forgetting’ helps PLMFIT get adapted better.

A strategic question to ask, on implementation, is ‘How often should we re-train PLMFIT, and at what cost?’. To analyze this, we experimented with the 10,000 images in Alipr trace #1, varying the interval L_{inter} between re-training while keeping everything else identical, and measuring the F-score. In each case,

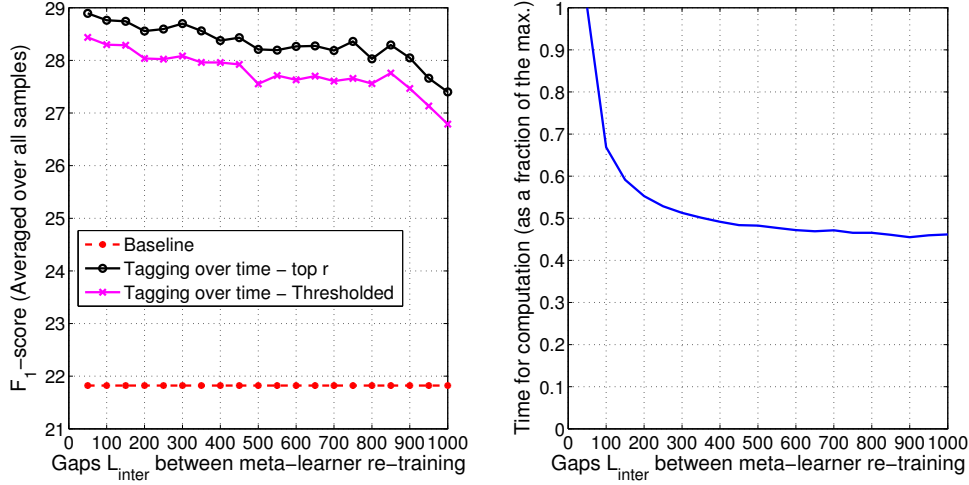


Figure 4.17. Comparing F-score and computation time with varying L_{inter} .

the computation durations are noted and normalized by the maximum time incurred, i.e., at $L_{inter} = 100$. These results are presented in Fig. 4.17. Note that with increasing gaps in re-training, F-score decreases to a certain extent, while computation time hits a lower bound, which is the amount needed exclusively for tagging. There is a clear trade-off between computational overhead and the F-score achieved. A graph of this nature can therefore help decide on this trade-off for a given application.

Finally, in Fig. 4.18, we show a sampling of images from a large number of cases (which we found via eye-balling) where annotation performance improves meaningfully with PLMFIT re-training over time. Specifically, at time 0 we show the top 5 tags given to the image by Alipr. This is followed by PLMFIT’s guesses after training with 1000 and 3000 temporally ordered images. Clearly, more correct tags are pushed up by the meta-learning process, which improves with more re-training data.

4.3.3 Personalized Tagging across People

Our final set of experiments are aimed at measuring the effectiveness of PLMFIT meta-learning for the purpose of user-wise personalization. Our experiments here are all based on 162,650 public Flickr images belonging to 300 real users. The black-box annotation system (baseline) used here is the Alipr algorithm [166]. Of



Figure 4.18. Sample annotation results found to improve over time with PLMFIT adaptation.

its vocabulary of 329, we find that 108 do not appear even once in the Flickr dataset. Aside from performance improvement with personalization, some other key aspects of interest were (a) *vocabulary expansion*, described in Sec. 4.2.3, for expanding the tag vocabulary of the black-box, (b) the effect of the *user average* method of tag selection, on personalization, and (c) the variation of performance with the amount of per-user data samples used.

Table 4.3. Personalization performance on 162,650 Flickr images (300 users)

Approach	Precision	% Change	Recall	% Change	F-score
Baseline [166]	1 in 60	-	1 in 25	-	2.3
PLMFIT (seed only)	1 in 14	+359%	1 in 10	+149%	8.5
PLMFIT (personalized)	1 in 7	+778%	1 in 8	+203%	13.04
PLMFIT (personalized+VE)	2 in 9	+1270%	2 in 5	+881%	27.98
VE = <i>vocabulary expansion</i>					

First, we computed the overall performance of different PLMFIT settings as compared to the baseline. A set of 5000 images, sampled from the full set of users, is set aside, and used as seed. For the case of PLMFIT, the top 10 tags are used for annotation prediction. For the purpose of personalization, the images from each user are divided into 75% training (images previously uploaded by the user)

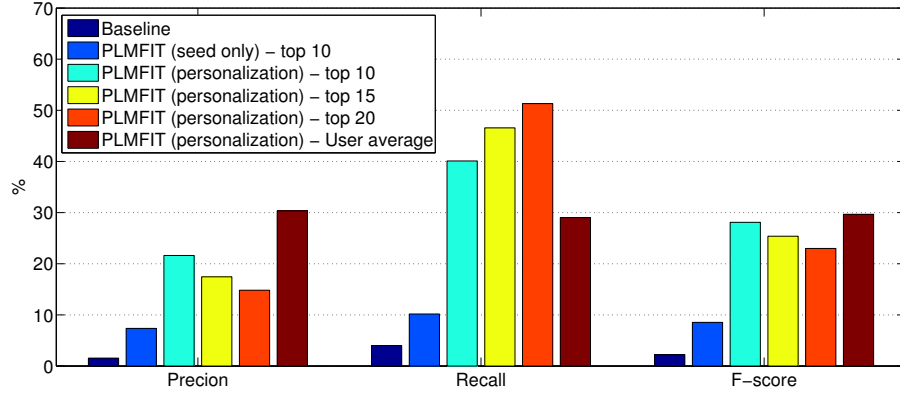


Figure 4.19. Graph showing precision, recall, and F-score for a number of settings, including Alipr’s tagging of the Flickr images (baseline), PLMFIT adaptation without personalization (seed), and personalization with different numbers of tags predicted. The case of ‘User Average’ uses the average number r of tags for a given user’s training data and predicts top r tags for the test cases.

and 25% (new images uploaded) testing. The results are presented in Table 4.3. We see that Alipr performance is significantly improved in all cases of PLMFIT use. For 287 out of the 300 users (i.e., 95.7%), PLMFIT with personalization leads to higher precision as well as recall than the baseline. When a random seed is used alone, performance improvement is relatively lower than with personalization, which can be partly explained by the graphs in Fig. 4.13. This justifies the need for personalization as against simply using one PLMFIT model for every user, i.e., contextual adaptation 4.2.1 helps but can be further improved with personalization. Of course, personalization in this manner is not possible for new users entering the system.

Second, we explore how the final tag selection strategy affects personalization performance. The setting remains the same as the previous case, except as specified. In Fig. 4.19, metrics are shown for the baseline, for PLMFIT with top 10, 15, or 20 tags being selected, and when the number of tags that PLMFIT predicts is based on the average for the given user. As expected, we see a clear trend whereby more tags predicted lead to higher recall at the cost of lower precision. More interestingly, the strategy of picking the average number of tags in the user training data for future prediction seems to be the winning strategy in terms of F-score. While this can be thought of as an additional personalization step, it was found to

be much less effective for users with high variance in tag counts. A more effective strategy can possibly be built around this fact.

Finally, we set out to test the effect of different mixes of seed data and user-specific data for personalization. Settings remain same as before, except that the PLMFIT model is trained differently. In these experiments, we only consider users which have over 800 images, so that upto 800 of them can be used for training. In our dataset, there were 83 such users, with a collective pool of 78,928 images. We mix randomly drawn seed images with these user-specific samples, to always come up with 800 training images. We train PLMFIT with this set, and compute performance metrics on the remainder of the images for that user, not used in training. Results, averaged over all users, are plotted in Fig. 4.20. At $x = 0$ in this graph, no user-specific data is used, and hence performance improvement is not dramatic. As the user-specific sample share gets larger, performance keeps improving, eventually flattening out at about the 0.6 mark. This can be justified by the fact that the non-Alipr tags tend to be less generic and more localized, thus more user data means improved estimation on these tags. We also observe that adding more user data seems to improve performance more significantly with vocabulary expansion than without it.

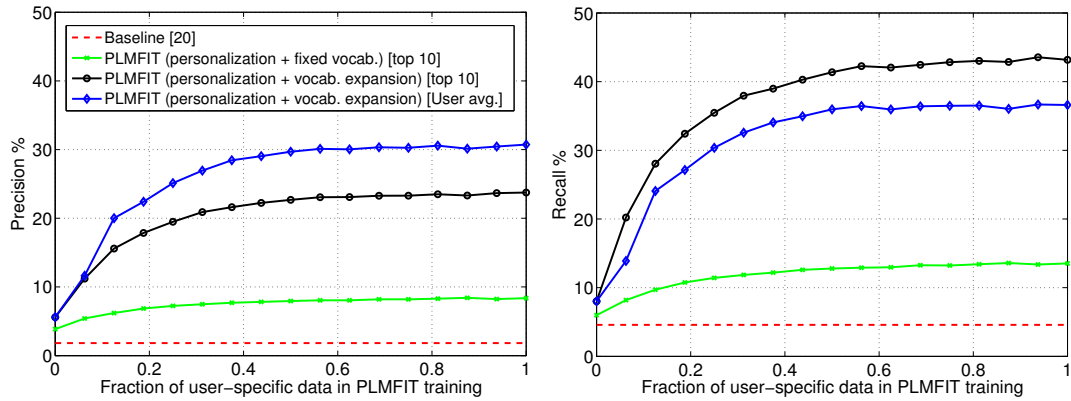


Figure 4.20. Graph showing variation of precision and recall with a varying proportion of seed data to user-specific samples, used for training the PLMFIT for personalization.

4.4 Issues and Limitations

The need to bring automatic image tagging to real-world applicability inspired us to employ a meta-learning approach. The ability to adapt to various kinds of scenario changes in a scalable manner led us to incorporate incremental learning into the approach. The results with our approach were found to be strongly positive, with large jumps in tagging performance, without any significant addition to the image analysis sophistication. That the results would be positive is very encouraging but not surprising. At the inception of work on this problem, we held the belief that even without extracting more profound semantics from visual features, there is a certain unknown amount of performance gain to be got for ‘free’. During analysis of the datasets, the rates of changes of tagging trends over time and across people that we observed exceeded our expectations, and further strengthened this belief. The more surprising result was how much exactly this amount turned out to be. A small amount of additional computation leads to large performance gain.

While we are much excited about the proposed approach, here we discuss some of its issues and limitations, and make some general comments:

- The PLMFIT approach is based on the assumption that the underlying black-box annotation system does, to some extent, learn semantics from visual features. A black-box system that maps image features to tags randomly is unlikely to benefit from meta-learning. The effectiveness of inductive transfer also depends on the nature of the black-box system.
- Unlike many other machine-learning problems, automatic image tagging approaches have historically been reported to perform moderately at best, and the *semantic gap* [242] has been most often cited as the main challenge in this learning task. PLMFIT, being dependent on an underlying annotation system, is thus bounded by the same. Despite the significant jumps in performance with our approach, the absolute values of the performance metrics leave much to be desired. Regardless, in this problem domain, a moderate precision and recall can still prove to be very useful in real-world applications.
- The choice of black-box annotation systems for testing PLMFIT was based on availability. We wish to experiment with other systems as well.

- The experimental results are based on real-world samples, but those samples do not represent every possible dynamism. We are unable to conclude on the adaptability of our approach under, for example, extreme changes.
- The ‘new user’ problem of personalization exists here as well. We use a generic ‘prior’ model for new users, estimated from all users, but performance gain is not as significant as when some of the user’s data is used in training. There is a possibility of exploiting the local neighborhood around a new user to obtain a more specific prior model, but we have not experimented with this idea.
- While adaptation over time and across people have been treated as separate scenarios, it is practical to also analyze adaptation performance under simultaneous change of time and users. We have skipped this combined analysis in order to keep the focus on the main contributions.
- In our Flickr dataset, tags exhibit a heavy-tailed distribution. Prior to pruning the tags by frequency, over 73% of the tags appeared only once. Models cannot be trained with them, which means that the heavy-tail imposes an upper-bound on recall.
- If the tag vocabulary consists of mostly proper nouns and non-English terms, as in the case of our Flickr dataset, an annotation system may well be predicting semantically correct tags, but there is no easy way to assess performance. Such a system will be useful for semantic image organization, but not in mimicking user tagging. The role of WordNet is also greatly diminished in such cases.
- While the proposed models and algorithms are designed specifically for the image tagging problem, they can generally apply to any learning task that involves making multiple binary decisions on each data point.

Despite these issues and limitations, the results presented in this chapter should encourage a greater use of meta-learning and a greater focus on the real-world applicability of automatic image tagging.

4.5 Summary

In this chapter, we have proposed the use of meta-learning and incremental learning to make automatic image tagging applicable in the real-world. We have proposed a principled, lightweight, meta-learning framework for image tagging (PLMFIT), inspired by inductive transfer, which can augment existing annotation systems, to allow adaptation to changes of any nature, such as contextual changes. We have proposed an algorithm to make use of PLMFIT to improve and adapt tagging over the time domain, and showed it to be effective. Finally, an algorithm that uses PLMFIT for personalized tagging has been proposed. The personalized version of PLMFIT is found to produce significantly better results than the generic version, showing improved tagging precision as well as recall for over 95% of the users in the Flickr dataset. In achieving this goal, methods to allow expansion of the system's tag vocabulary beyond that of the initial version, has been presented. In all cases, efficiency is achieved via incremental learning, and the methods have been validated using large real-world datasets. In general, we can conclude that the meta-learning approach to image tagging appears has many attractive properties.

Beyond Semantics: Basics, Inference, and Applications of Aesthetics

The image processing and analysis community has, for long, attempted to quantify and rectify image quality at a low-level, given the original image [77] or without it [236]. At a higher level, the perception often affects our emotion and mood, but there has been little headway made in automatic inferencing of the quality in images that affect mood or emotion. What makes the latter problem hard is that low-level image properties are insufficient to characterize high-level perception of aesthetics. Furthermore, there is a lack of precise definitions, assessment metrics, and test data for this problem, despite being desirable for many applications, e.g., image search, photography, story illustration, and photo enhancement.

In this chapter, we attempt to clear the cloud on the problem of natural image aesthetics inference from visual content, by defining problems of interest, target audiences and how they affect the problem at hand, assessment metrics, and introduce real-world datasets for testing. Insights are drawn from the handful of previous attempts [56, 61, 141, 258] at solving related problems. While facial attractiveness has been a theme for many popular Websites [206], and has led to work on automatic facial aesthetics inference [75] that make use of symmetry and proportion, here we concern ourselves with generic images.



Figure 5.1. Three aesthetics inferencing problems of significance.

5.1 Questions of Interest

Being in its nascent stage, research on algorithmic aesthetics inference needs concretely defined tasks to solve, to start with. Aesthetics of natural images are, simply put, the emotions they arouse in people, which makes it relatively ill-defined. Contentious issues are ‘emotion’ and ‘people’. Emotions are subjective across individuals, and they are of varied types (pleasing, boring, irritating, etc.). We leave aside subjectivity for now and consider aesthetic attributes to be a *consensus* measure over the entire population, such that they are meaningful to the average individual. Three data-driven aesthetics inference questions (Fig. 5.1) are discussed below.

5.1.1 Aesthetics Score Prediction

When a photograph is rated by a set of n people on a 1 to D scale on the basis of its aesthetics, the average score can be thought of as an estimator for its *intrinsic* aesthetic quality. More specifically, we assume that an image I has associated with it a true aesthetics measure $q(I)$, which is the asymptotic average if the entire population rated it. The average over the size n sample of ratings, given by $\hat{q}(I) = \frac{1}{n} \sum_{i=1}^n r_i(I)$ is an estimator for the population parameter $q(I)$, where $r_i(I)$ is the i^{th} rating given to image I . Intuitively, a larger n gives a better estimate.

A formulation for aesthetics score prediction is therefore to infer the value of $\hat{q}(I)$ by analyzing the content of image I , which is a direct emulation of humans in the photo rating process. This lends itself naturally to a regression setting, whereby some abstractions of visual features act as predictor variables and the estimator for $\hat{q}(I)$ is the dependent variable. An attempt at regression based score prediction has been reported in [56], showing very limited success.

Assessment: One method for assessing the quality of scoring prediction is to compute the rate or distribution of error [56].

5.1.2 Aesthetics Class Prediction

It has been observed both in [56] and [141] that score prediction is a very challenging problem, mainly due to noise in user ratings. Given the limited size rating samples, their averaged estimates have high variance, e.g., 5 and 5.5 on a 1–7 scale could easily have been interchanged if a different set of users rated them, but there is no way to infer this from content alone, which leads to large prediction errors. To make the problem more solvable, the regression problem is changed to one of classification, by thresholding the average scores to create *high* vs. *low* quality image classes [56], or *professional* vs. *snapshot* image classes [141]. Suppose threshold values are *HIGH* and *LOW* respectively, then $class(I)$ is 1 if $\hat{q}(I) \geq HIGH$ and 0 if $\hat{q}(I) \leq LOW$. When the *band gap* $\delta = HIGH - LOW$ increases, the two classes are more easily separable, a hypothesis that has been tested and found to hold, in [56]. An easier problem but of practical significance is that of selecting a few representative high quality or highly aesthetic photographs from a large collection. In this case, it is important to ensure that most of the selected images are of high quality even though many of those not selected may be of high quality as well. An attempt at this problem [61] has proven to be more successful than the general HIGH/LOW classification problem described previously.

Assessment: The HIGH/LOW classification problem solutions can be evaluated by standard accuracy measures [56, 141]. On the other hand, the selection of high-quality photos need only maximize the *precision* in high quality within the top few photos, with *recall* being less critical.

5.1.3 Emotion Prediction

If we group emotions that natural images arouse into categories such as ‘pleasing’, ‘boring’, and ‘irritating’, then emotion prediction can be conceived as a multi-class categorization problem. These categories are fuzzily defined and judgments are highly subjective. Consider K such emotion categories, and people select one or more of these categories for each image. If an image I gets votes in the proportion $\Pi_1(I), \dots, \Pi_K(I)$, then two possible questions arise, none of which have been attempted in the past.

Most Dominant Emotion: We wish to predict, for an image I , the most voted emotion category $k(I)$, i.e., $k(I) = \arg \max_i \Pi_i(I)$. The problem is only meaningful when there is clear dominance of $k(I)$ over others, thus only these samples must be used for learning.

Emotion Distribution: Here, we wish to predict the distribution of votes (or an approximation) that an image receives from users, i.e., $\Pi_1(I), \dots, \Pi_K(I)$, which is well-suited when images are fuzzily associated with multiple emotions.

Assessment: The ‘most dominant emotion’ problem is assessed like any standard multi-class classification problem. For ‘emotion distribution’, assessment requires a measure of similarity between discrete distributions, for which Kullback-Leibler (KL) divergence is a possible choice.

5.1.4 Context

In practice, any solution to the above problems can be tested either by user-generated feedback in online photo-sharing communities [213, 71, 3, 206], or by controlled user studies. Given this data-dependence, none of the models proposed will be fundamental or absolute in what they learn about aesthetics, but will be tempered to the given data acquisition setup, which we call the *context*. For example, what is considered ‘interesting’ (Flickr) may not be treated as being ‘aesthetically pleasing’ (Photo.net) by the population, and vice-versa. Therefore, we implicitly refer to it as aesthetics inference *under a given context* \mathcal{X} . Examples of key contextual aspects of test data are (a) the exact question posed to the users about the images, e.g., ‘aesthetics’ [213], ‘overall quality’ [71], ‘like it’ [3], (b) the type of people who visit and vote on the images, e.g., general enthusiasts [71,

213], photographers [213], and (c) The type of images rated, e.g., travel [251], topical [71]. Until fundamentals of aesthetics judgment are uncovered, contextual information is critical. The long-term goal is to have solutions that apply to as general a context as possible.

5.1.5 Personalization

While consensus measures and averaged-out ratings provide a generic learning setting, *personalized* models are of high relevance here due to the significant amount of subjectivity. In line with recommender systems, personalized models of aesthetics can potentially be learned, given sufficient feedback from a single user. In the absence of sufficient feedback from individuals, one solution is to consider *cliques* (groups or clusters of people with shared taste) instead of individuals, and make personalized inferences with respect to an user’s parent clique, thus providing more data to learn. The cliques should ideally be determined automatically, may be overlapping, and an individual may belong to multiple cliques. There has been no reported attempt at personalized aesthetics.

5.2 Technical Solution Approaches

Analogous to the concept of *semantic gap* that implies the technical limitations of image recognition, we can define the technical challenge in automatic inference of aesthetics in terms of the *aesthetic gap*, as follows:

The aesthetic gap is the lack of coincidence between the information that one can extract from low-level visual data (i.e., pixels in digital images) and the interpretation of emotions that the visual data may arouse in a particular user in a given situation.

Past attempts [61, 141, 258] at aesthetics and quality inference have followed a logical series of steps, as discussed below.

5.2.1 Feature Shortlisting

Possibly the most challenging part of the problem is conceiving meaningful visual properties that may have correlation with human ratings, and devising ways to convert them into numerical features. While feature shortlisting is largely ad-hoc in [258], the photography literature provides much of the intuitions for [56, 141]. The hypothesis there is that photographers follow principles (rule of thirds, complementary colors, etc.) that lead to aesthetically pleasing shots. The features proposed previously are limited, so there is scope for more comprehensive shortlisting.

5.2.2 Feature Selection

Once a feature set is decided, the hypothesis needs to be tested so as to eliminate those that in reality show no correlation with human ratings, given the data. For feature selection, [258] employs *boosting*, while [56] uses *forward selection*. There is further scope for effective exploitation of correlation across features in aesthetics modeling.

5.2.3 Statistical Learning and Inferencing

A suitable learning method, that makes use of the selected features to model aesthetics, is essential. Previous attempts have employed decision trees [56], Bayesian classifiers [61, 141, 258], SVMs [56, 258], boosting [258], and regression [56, 61], for answering one or more of the questions in Sec. 5.1. In general, we need some form of regression for score prediction (Sec. 5.1.1), a two-class classifier for class prediction (Sec. 5.1.2), and a multi-class discriminative or generative classifier for emotion prediction (Sec. 5.1.3). Because past efforts have yielded only limited success, a deeper exploration is needed to figure out if feature extraction alone is the performance bottleneck, or whether better learning method can also improve performance.

Table 5.1. Datasets available for emotion/aesthetics learning.

Source	Feedback Type	Average Scores	Score Distribution	Individual Scores
Photo.net	1-7 (aesthetics)	Yes	Yes	Yes (partial)
DPChallenge	1-10 (quality)	Yes	Yes	No
Terragalleria	1-10 (liking)	Yes	Yes	No
Alipr.com	Emotion (8 types)	n/a	n/a	n/a

5.3 Public Datasets for Empirical Modeling

Due to lack of theoretical grounding and controlled experimental data, there is heavy dependence on publicly available data for understanding, development, and validation for this problem, which include Web-based sources [3, 213, 251, 71] that solicit user feedback on image quality and aesthetics. A summary of some sources and the characteristics of available data is presented in Table 5.1. We collected large samples from each data source, drawing at random, to create real-world datasets (to be available at <http://riemann.ist.psu.edu/>) that can be used to compare competing algorithms. A description and preliminary analysis follows.

Photo.net: This Website [213] provides a platform for photography enthusiasts to share and get their shots peer-rated on a 1 – 7 scale on their aesthetic quality. We collected a set of 14,839 images, each rated by at least one user. The mean number of ratings per image is 12, with a std. dev. of 13. A smaller dataset from this source has been used before [56, 61].

DPChallenge: This Website [71] allows users to participate in theme-based photography contests, and peer-rating on overall quality, on a 1-10 scale, determines winners. We collected 16,509 images, each rated by at least one user. The mean number of ratings per image is 205, with a std. dev. of 53. A smaller dataset from this source has been before [141].

Terragalleria: This Website [251] showcases travel photography of Quang-Tuan Luong, and is one of the best sources of US national park photography. Thus, all photographs are taken by one person (unlike before), but multiple users rate them on overall quality, on a 1-10 scale. The mean number of ratings per image is 22, with a std. dev. of 23. We obtained 14,449 images from here, each rated by at least one user.

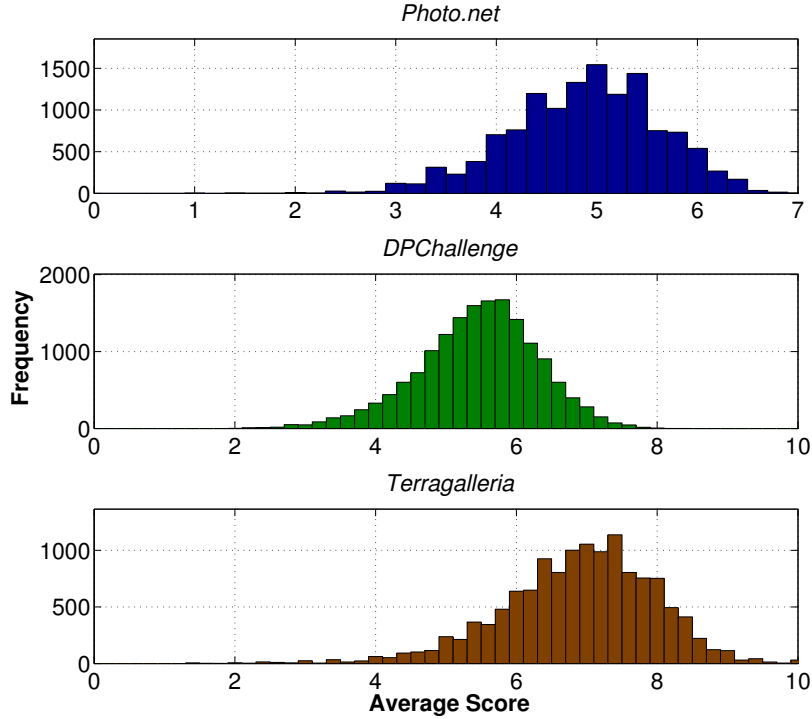


Figure 5.2. Distributions of the average photo ratings received.

Alipr: This Website [3], primarily meant for image search and tagging, also allows users to rate photographs on the basis of 10 different emotions (See Fig.5.6). We collected 13,010 emotion-tagged images (with repetitions).

5.3.1 Analysis

For the benefit of experimental design and dataset selection, we report on an analysis of each dataset, in particular the nature of user ratings received in each case (not necessarily comparable across the datasets). Figures 5.2 and 5.3 show the average score and score count distributions respectively, of sources [213, 71, 251]. Considering that the three scales are normalized to the same range, DPChallenge ratings are lower, on an average, which might reflect on the competitive nature. For the same reason, the number of ratings received per image are higher than the other two, which indicate that the averaged scores represent the consensus better.

We then look at the correlation between the number of ratings and the average score for each image, by plotting the tuple corresponding to each image, in Fig. 5.4.

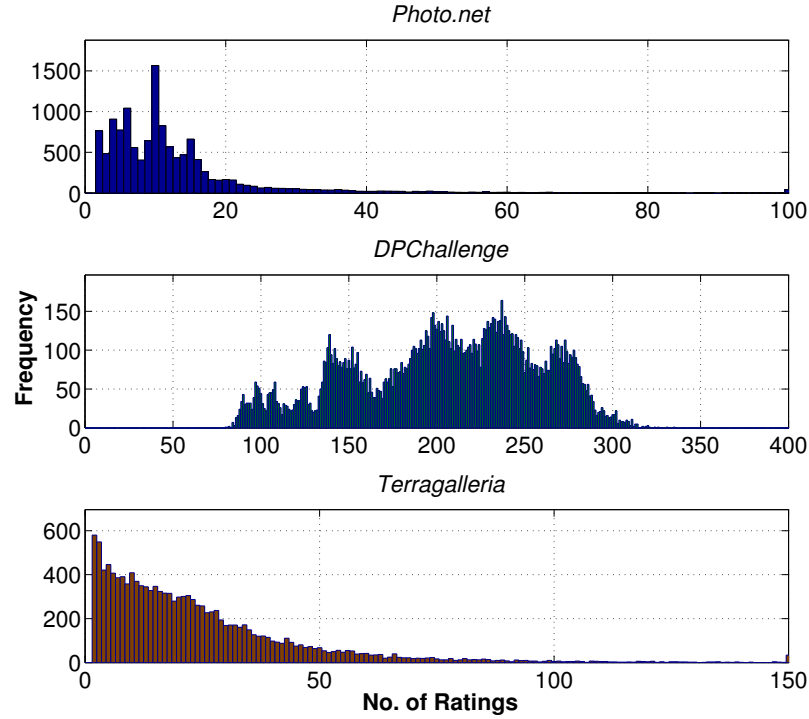


Figure 5.3. Distributions of number of photo ratings received per image.

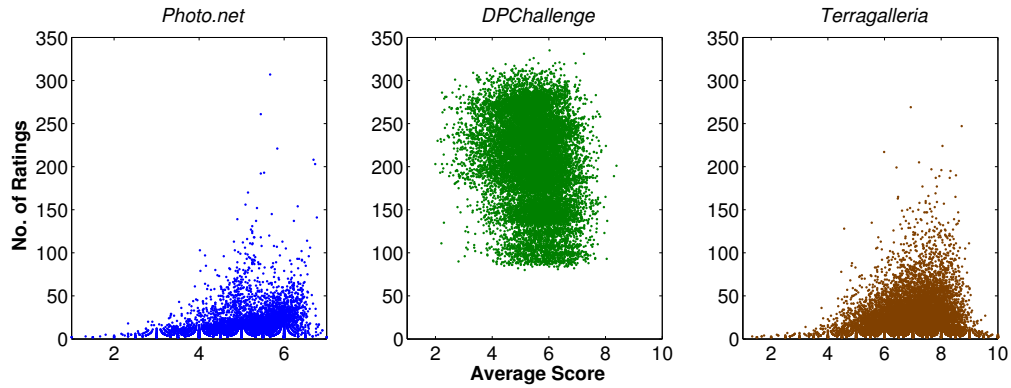


Figure 5.4. Correlation plot of (average score, number of ratings) pairs.

Considering uniform random samples, the graphs indicate that in Photo.net and Terragalleria more users rate higher quality photographs, while this skewness is less prominent in DPChallenge. This must be carefully considered when designing inference methods. Another point of interest is consensus, i.e., the extent of agreeability in rating, among users. Let n be the number of ratings given by users, a be their average, and x be the number of ratings within $a \pm 0.5$, with greater value

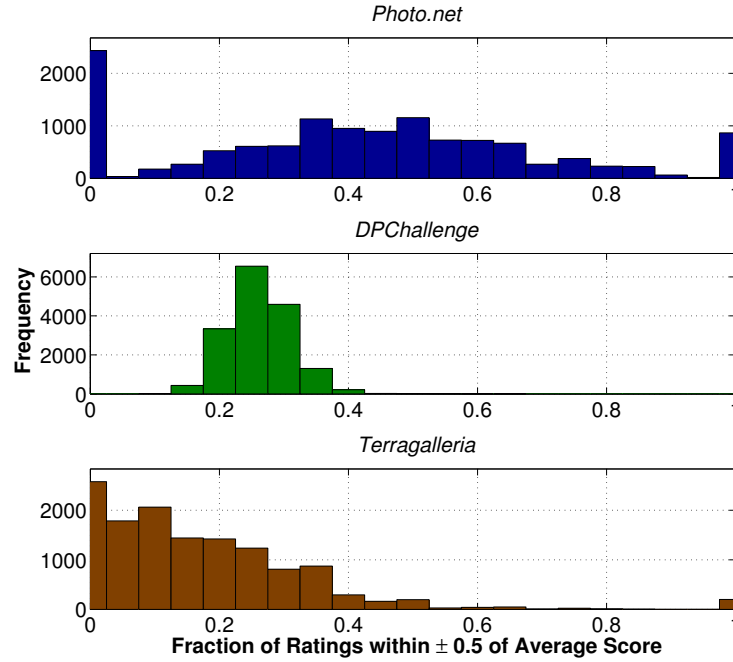


Figure 5.5. Distribution of the level of consensus among ratings.

indicating greater consensus. The distribution of x/n over all images is shown in Fig. 5.5, which roughly indicates that Photo.net has better consensus over the ratings than the other two.

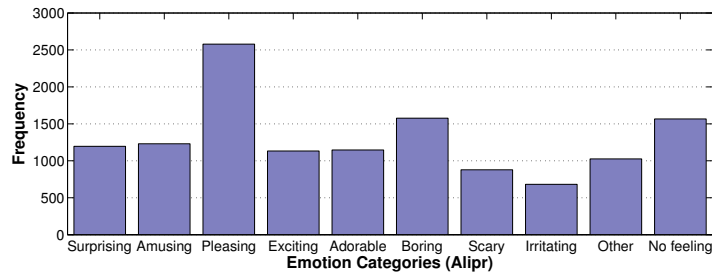


Figure 5.6. Distribution of emotion votes given to images (Alipr).

Finally, we plot the distribution of emotion votes for the dataset sampled from Alipr [3], where ‘pleasing’ may be related to high aesthetics or quality, while ‘boring’ or ‘no feeling’ may indicate otherwise. Despite over 13,000 votes, the number of them on a per-image basis is low. For higher reliability, we must wait till a greater number of votes are cast.

5.4 Visual Features for Photographic Aesthetics

Photography is the art or practice of taking and processing photographs [217]. Aesthetics in photography is how people usually characterize beauty in this form of art. There are various ways in which aesthetics is defined by different people. There exists no single consensus on what it exactly pertains to. The broad idea is that photographic images that are pleasing to the eyes are considered to be higher in terms of their aesthetic beauty. What pleases or displeases one person may be different from what pleases or displeases another person. While the average individual may simply be interested in how soothing a picture is to the eyes, a photographic artist may be looking at the composition of the picture, the use of colors and light, and any additional meanings conveyed by the picture. A professional photographer, on the other hand, may be wondering how difficult it may have been to take or to process a particular shot, the sharpness and the color contrast of the picture, or whether the “rules of thumb” in photography have been maintained. All these issues make the measurement of aesthetics in pictures or photographs extremely subjective. In spite of the ambiguous definition of aesthetics, we show in this chapter that there exist certain visual properties which make photographs, *in general*, more aesthetically beautiful than some others. Our results and findings could be of interest to the scientific community, as well as to the photographic art community and manufacturers for image capturing devices. Content analysis in photographic images has been studied by the multimedia and vision research community in the past decade. Today, several efficient region-based image retrieval engines are in use [177, 30, 278, 243]. Statistical modeling approaches have been proposed for automatic image annotation [9, 163]. Culturally significant pictures are being archived in digital libraries. Online photo sharing communities are becoming more and more common [2, 7, 88, 213]. In this age of digital picture explosion, it is critical to continuously develop intelligent systems for automatic image content analysis. The advantages of such systems can be reaped by the scientific community as well as common people.

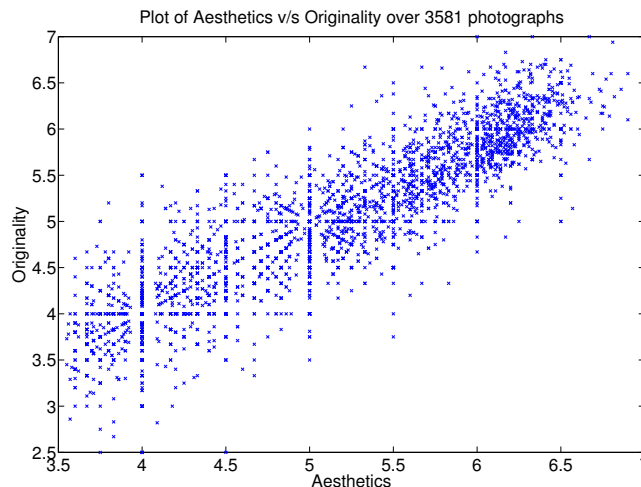


Figure 5.7. Correlation between the aesthetics and originality ratings for 3581 photographs obtained from Photo.net.

5.4.1 Photo.net: Community-based Photo Ratings

As discussed previously, a good data source for empirical modeling of aesthetics is the large on-line photo sharing community, *Photo.net*, started in 1997 by Philip Greenspun, then a researcher on online communities at MIT [213]. Primarily intended for photography enthusiasts, the Website attracts more than 400,000 registered members. Many amateur and professional photographers visit the site frequently, share photos, and rate and comment on photos taken by peers. There are more than one million photographs uploaded by these users for perusal by the community. Of interest to us is the fact that many of these photographs are peer-rated in terms of two qualities, namely *aesthetics* and *originality*. The scores are given in the range of one to seven, with a higher number indicating better rating.

This site acts as the main source of data for our computational aesthetics work. The reason we chose such an online community is that it provides photos which are rated by a relatively diverse group. This ensures generality in the ratings, averaged out over the entire spectrum of amateurs to serious professionals. While amateurs represent the general population, the professionals tend to spend more time on the technical details before rating the photographs. This is evident from the comments that are posted by peers on photographs, often in an attempt to justify their ratings. Because this is a photo sharing community, there can be some bias towards

the opinions of professional photographers over the general population, but this is not critical since opinions of professionals often reflect on what satisfies their customers on an average. Hence, we use these ratings as indicators of aesthetics in photography. We recommend the readers to peruse the aforementioned Website to get a better understanding of the data source. *One caveat:* The nature of any peer-rated community is such that it leads to unfair judgements under certain circumstances, and *Photo.net* is no exception, making the data fairly noisy.

We downloaded those pictures and their associated meta-data which were rated by at least two members of the community. In order not to over-distract the normal services provided by the site, we downloaded the data slowly and over a long-period of time for our research. For each image downloaded, we parsed the pages and gathered the following information: (1) average aesthetics score between 1.0 and 7.0, (2) average originality score between 1.0 and 7.0, (3) number of times viewed by members, and (4) number of peer ratings.

5.4.2 Aesthetics v/s Originality

By definition[217], *Aesthetics* means (1) “concerned with beauty and art and the understanding of beautiful things”, and (2) “made in an artistic way and beautiful to look at”. A more specific discussion on the definition of aesthetics can be found in [214]. As can be observed, no consensus was reached on the topic among the users, many of whom are professional photographers. *Originality* has a more specific definition of being something that is unique and rarely observed. The originality score given to some photographs can also be hard to interpret, because what seems original to some viewers may not be so for others. Depending on the experiences of the viewers, the originality scores for the same photo can vary considerably. Thus the originality score is subjective to a large extent as well. Even then, the reasons that hold for aesthetics ratings also hold for originality, making this data a fairly general representation of the concept of originality and hence safe to use for statistical learning purposes.

One of the first observations made on the gathered data was the strong correlation between the aesthetics and originality ratings for a given image. A plot of 3581 unique photograph ratings can be seen in Fig. 5.7. As can be seen, aesthetics



Figure 5.8. Aesthetics scores can be significantly influenced by the semantics. Loneliness is depicted using a person in this frame, though the area occupied by the person is very small. Average aesthetics score for these images are 6.0 out of 7 (left) and 6.61 out of 7 (right).

and originality ratings have approximately linear correlation with each other. This can be due to a number of factors. Many users quickly rate a batch of photos in a given day. They tend not to spend too much time trying to distinguish between these two parameters when judging a photo. They more often than not rate photographs based on a general impression. Typically, a very original concept leads to good aesthetic value, while beauty can often be characterized by originality in view angle, color, lighting, or composition. Also, because the ratings are averages over a number of people, disparity by individuals may not be reflected as high in the averages. Hence there is generally not much disparity in the average ratings. In fact, out of the 3581 randomly chosen photos, only about 1.1% have a disparity of more than 1.0 between average aesthetics and average originality, with a peak of 2.0.

As a result of this observation, we chose to limit the rest of our study to aesthetics ratings only, since the value of one can be approximated to the value of the other, and among the two, aesthetics has a rough definition that in principle depends somewhat less on the content or the semantics of the photograph, something that is very hard for present day machine intelligence to interpret accurately. Nonetheless, the strong dependence on originality ratings mean that aesthetics ratings are also largely influenced by the semantics. As a result, some visually similar photographs are rated very differently. For example in Fig. 5.8, loneliness is depicted using a man in each frame, increasing its appeal, while the lack of the person makes the photographs uninteresting and is likely causing poorer

ratings from peers. This makes the task of automatically determining aesthetics of photographs highly challenging.

5.4.3 Our Approach to Aesthetics Inference

Our desire is to take the first step in understanding what aspects of a photograph appeal to people, from a population and statistical stand-point. For this purpose, we aim to build (1) a classifier that can qualitatively distinguish between pictures of *high* and *low* aesthetic value, or (2) a regression model that can quantitatively predict the aesthetics score, both approaches relying on low-level visual features only. We define *high* or *low* in terms of predefined ranges of aesthetics scores.

There are reasons to believe that classification may be a more appropriate model than regression in tackling this problem. For one, the measures are highly subjective, and there are no agreed standards for rating. This may render absolute scores less meaningful. Again, ratings above or below certain thresholds on an average by a set of unique users generally reflect on the photograph’s quality. This way we also get around the problem of consistency where two identical photographs can be scored differently by different groups of people. However, it is more likely that both the group averages are within the same range and hence are treated fairly when posed as a classification problem.

On the other hand, the ‘ideal’ case is when a machine can replicate the task of robustly giving images aesthetics scores in the range of (1.0-7.0) the humans do. This is the regression formulation of the problem. Nevertheless, in this work we attempt both classification and regression models on the data. The possible benefits of building a *computational aesthetics* model can be summarized as follow: If the low-level image features alone can tell what range aesthetics ratings the image deserves, this can potentially be used by photographers to get a rough estimate of their shot composition quality, leading to adjustment in camera parameters or shot positioning for improved aesthetics. Camera manufacturers can incorporate a ‘suggested composition’ feature into their products. Alternatively, a content-based image retrieval system can use the aesthetics score to discriminate between visually similar images, giving greater priority to more pleasing query results. A reasonable solution to this problem can lead to a better understanding of human vision.

5.5 Feature Extraction

Experiences with photography lead us to believe in certain aspects as being critical to quality. This entire study is on such beliefs or hypotheses and their validation through numerical results. We treat each downloaded image separately and extract features from them. We use the following notation: The *RGB* data of each image is converted to *HSV* color space, producing two-dimensional matrices I_H , I_S , and I_V , each of dimension $X \times Y$. In photography and color psychology, color tones and saturation play important roles, and hence working in the *HSV* color space makes computation more convenient. For some features we extract information from objects within the photographs. An approximate way to find objects within images is segmentation, under the assumption that homogeneous regions correspond to objects. We use a fast segmentation method based on clustering. For this purpose the image is transformed into the *LUV* space, since in this space locally Euclidean distances model the perceived color change well. Using a fixed threshold for all the photographs, we use the *K*-Center algorithm to compute cluster centroids, treating the image pixels as a bag of vectors in *LUV* space. With these centroids as seeds, a *K*-means algorithm computes clusters. Following a connected component analysis, color-based segments are obtained. The 5 largest segments formed are retained and denoted as $\{s_1, \dots, s_5\}$. These clusters are used to compute *region-based features* as we shall discuss in Sec. 5.5.8.

We extracted 56 visual features for each image in an empirical fashion, based on (a) our own intuitions, (b) comments posted by peers on a large collection of high and low rated pictures, and (c) ease of interpretation of results. The feature set was carefully chosen but limited because our goal was mainly to study the trends or patterns, if any, that lead to higher or lower aesthetics ratings. If the goal was to only build a strong classifier or regression model, it would have made sense to generate exhaustive features and apply typical machine-learning techniques such as boosting. Without meaningful features it is difficult to make meaningful conclusions from the results. We refer to our features as *candidate features* and denote them as $\mathcal{F} = \{f_i | 1 \leq i \leq 56\}$ which are described as follows.

5.5.1 Exposure of Light

Measuring the brightness using a light meter and a gray card, controlling the exposure using the aperture and shutter speed settings, and darkroom printing with dodging and burning are basic skills for any professional photographer. Too much exposure (leading to brighter shots) often yields lower quality pictures. Those that are too dark are often also not appealing. Thus light exposure can often be a good discriminant between high and low quality photographs. Note that there are always exceptions to any ‘rules of thumb’. An over-exposed or under-exposed photograph under certain scenarios may yield very original and beautiful shots. Therefore it is prudent to not expect or depend too much on individual features. This holds for all features, since photographs in [213] are too diverse to be judged by a single parameter. Ideally, the use of light should be characterized as normal daylight, shooting into the sun, backlighting, shadow, night etc. We use the average pixel intensity to characterize the use of light:

$$f_1 = \frac{1}{XY} \sum_{x=0}^{X_1} \sum_{y=0}^{Y-1} I_V(x, y) \quad .$$

5.5.2 Colorfulness

We propose a fast and robust method to compute relative color distribution, distinguishing multi-colored images from monochromatic, *sepia*¹ or simply low contrast images. We use the Earth Mover’s Distance (EMD) [221], which is a measure of similarity between any two weighted distributions. We divide the *RGB* color space into 64 cubic blocks with four equal partitions along each dimension, taking each such cube as a sample point. Distribution D_1 is generated as the color distribution of a hypothetical image such that for each of 64 sample points, the frequency is $1/64$. Distribution D_2 is computed from the given image by finding the frequency of occurrence of color within each of the 64 cubes. The EMD measure requires that the pairwise distance between sampling points in the two distributions be supplied. Since the sampling points in both of them are identical, we compute the pairwise Euclidean distances between the geometric centers

¹<http://www.knaw.nl/ecpa/sepia/home.html>

c_i of each cube i , after conversion to LUV space. Thus the *colorfulness* measure f_2 is computed as follows: $f_2 = emd(D_1, D_2, \{d(a, b) \mid 0 \leq a, b \leq 63\})$, where $d(a, b) = \|\text{rgb2luv}(c_a) - \text{rgb2luv}(c_b)\|$.

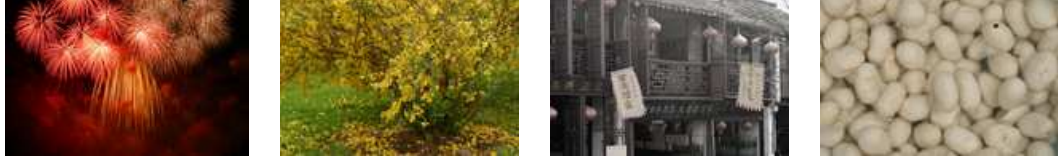


Figure 5.9. The proposed *colorfulness* measure, f_2 . The two photographs on the *left* have high values while the two on the *right* have low values.

The distribution D_1 can be interpreted as the *ideal* color distribution of a ‘colorful’ image. How similar the color distribution of an arbitrary image is to this one is a rough measure of how colorful that image is. Examples of images producing high and low values of f_2 are shown in Fig. 5.9.

5.5.3 Saturation and Hue

Saturation indicates chromatic purity. Pure colors in a photo tend to be more appealing than dull or impure ones. In natural out-door landscape photography, professionals use specialized film such as the *Fuji Velvia* to enhance the saturation to result in deeper blue sky, greener grass, more vivid flowers, etc. We compute the saturation indicator as the average saturation f_3 over the picture,

$$f_3 = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I_S(x, y) \ .$$

Hue is similarly computed averaged over I_H to get feature f_4 , though the interpretation of such a feature is not as clear as the former. This is because hue as defined in the HSV space corresponds to angles in a color wheel.

5.5.4 The Rule of Thirds

A very popular rule of thumb in photography is the *Rule of Thirds*. The rule can be considered as a sloppy approximation to the ‘golden ratio’ (about 0.618), a visualization proportion discovered by the ancient Greeks. It specifies that the

main element, or the center of interest, in a photograph should lie at one of the four intersections as shown in Fig. 5.10 (a). Browsing through a large number of professional photographs it was observed that most of those that follow this rule have the main object stretch from an intersection up to the center of the image. Also noticed was the fact that centers of interest, e.g., the eye of a man, were often placed aligned to one of the edges, on the inside. This implies that a large part of the main object often lies on the periphery or inside of the inner rectangle. Based on these observations, we computed the average hue as feature f_5 , with f_6 and f_7 being similarly computed for I_S and I_V respectively:

$$f_5 = \frac{9}{XY} \sum_{x=X/3}^{2X/3} \sum_{y=Y/3}^{2Y/3} I_H(x, y)$$

Although it may seem redundant to use as feature vectors the average saturation and intensity once for the whole image and once for the inner third, it must be noted that the latter may often pertain exclusively to the main object of interest within the photograph, and hence can potentially convey different kind of information.

5.5.5 Familiarity Measure

We humans learn to rate the aesthetics of pictures from the experience gathered by seeing other pictures. Our opinions are often governed by what we have seen in the past. Because of our curiosity, when we see something unusual or rare we perceive it in a way different from what we get to see on a regular basis. In order to capture this factor in human judgment of photography, we define a new measure of *familiarity* based on the integrated region matching (IRM) image distance [278]. The IRM distance computes image similarity by using color, texture and shape information from automatically segmented regions, and performing a robust region-based matching with other images. Primarily meant for image retrieval applications, we use it here to quantify familiarity. Given a pre-determined *anchor* database of images with a well-spread distribution of aesthetics scores, we retrieve the top K closest matches in it with the candidate image as query. Denoting IRM distances of the top matches for each image in decreasing order of rank

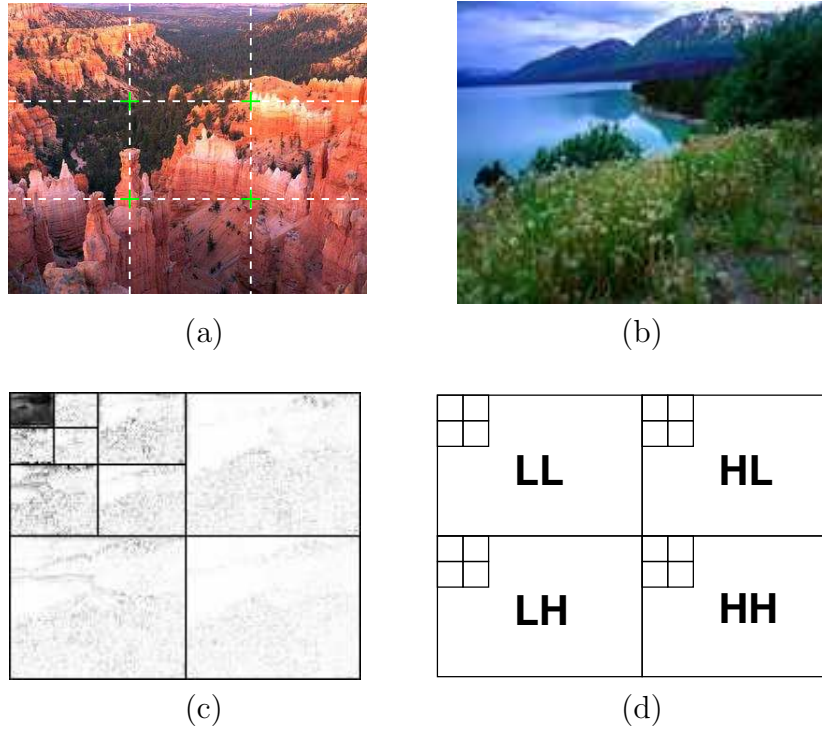


Figure 5.10. (a) The *rule of thirds* in photography: Imaginary lines cut the image horizontally and vertically each into three parts. Intersection points are chosen to place important parts of the composition instead of the center. (b)-(d) Daubechies wavelet transform. *Left:* Original image. *Middle:* Three-level transform, levels separated by borders. *Right:* Arrangement of three bands LH, HL and HH of the coefficients.

as $\{q(i) | 1 \leq i \leq K\}$. We compute f_8 and f_9 as

$$f_8 = \frac{1}{20} \sum_{i=1}^{20} q(i) , \quad f_9 = \frac{1}{100} \sum_{i=1}^{100} q(i) .$$

In effect, these measures should yield higher values for uncommon images (in terms of their composition). Two different scales of 20 and 100 top matches are used since they may potentially tell different stories about the uniqueness of the picture. While the former measures average similarity in a local neighborhood, the latter does so on a more global basis. Because of the strong correlation between aesthetics and originality, it is intuitive that a higher value of f_8 or f_9 corresponds to greater originality and hence we expect greater aesthetics score.

5.5.6 Wavelet-based Texture

Graininess or smoothness in a photograph can be interpreted in different ways. If as a whole it is grainy, one possibility is that the picture was taken with a grainy film or under high ISO settings. If as a whole it is smooth, the picture can be out-of-focus, in which case it is in general not pleasing to the eye. Graininess can also indicate the presence/absence and nature of *texture* within the image.

The use of texture is a composition skill in photography. One way to measure spatial smoothness in the image is to use Daubechies wavelet transform [63], which has often been used in the literature to characterize texture. We perform a *three-level* wavelet transform on all three color bands I_H , I_S and I_V . An example of such a transform on the intensity band is shown in Fig. 5.10 (b)-(c). The three levels of wavelet bands are arranged from top left to bottom right in the transformed image, and the four coefficients per level, LL , LH , HL , and HH are arranged as shown in Fig. 5.10 (d). Denoting the coefficients (except LL) in level i for the wavelet transform on hue image I_H as w_i^{hh} , w_i^{hl} and w_i^{lh} , $i = \{1, 2, 3\}$, we define features f_{10} , f_{11} and f_{12} as follows:

$$f_{i+9} = \frac{1}{S_i} \left\{ \sum_x \sum_y w_i^{hh}(x, y) + \sum_x \sum_y w_i^{hl}(x, y) + \sum_x \sum_y w_i^{lh}(x, y) \right\}$$

where $S_k = |w_i^{hh}| + |w_i^{hl}| + |w_i^{lh}|$ and $i = 1, 2, 3$. The corresponding wavelet features for saturation (I_S) and intensity (I_V) images are computed similarly to get f_{13} through f_{15} and f_{16} through f_{18} respectively. Three more wavelet features are derived. The sum of the average wavelet coefficients over all three frequency levels for each of H , S and V are taken to form three additional features: $f_{19} = \sum_{i=10}^{12} f_i$, $f_{20} = \sum_{i=13}^{15} f_i$, and $f_{21} = \sum_{i=16}^{18} f_i$.

5.5.7 Size and Aspect Ratio

The size of an image has a good chance of affecting the photo ratings. Although scaling is possible in digital and print media, the size presented initially must be agreeable to the content of the photograph. A more crucial parameter is the aspect ratio. It is well-known that 4 : 3 and 16 : 9 aspect ratios, which approximate the ‘golden ratio,’ are chosen as standards for television screens or 70mm movies, for

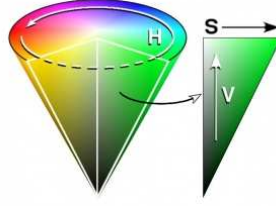


Figure 5.11. The HSV Color Space.

reasons related to viewing pleasure. The 35mm film used by most photographers has a ratio of 3 : 2 while larger formats include ratios like 7 : 6 and 5 : 4. While size feature $f_{22} = X + Y$, the aspect ratio feature $f_{23} = \frac{X}{Y}$.

5.5.8 Region Composition

Segmentation results in rough grouping of similar pixels, which often correspond to objects in the scene. We denote the set of pixels in the largest five connected components or *patches* formed by the segmentation process described before as $\{s_1, \dots, s_5\}$. The number of patches $t \leq 5$ which satisfy $|s_i| \geq \frac{XY}{100}$ denotes feature f_{24} . The number of color-based clusters formed by K -Means in the LUV space is feature f_{25} . These two features combine to measure how many distinct color *blobs* and how many disconnected significantly large regions are present.

We then compute the average H , S and V values for each of the top 5 patches as features f_{26} through f_{30} , f_{31} through f_{35} and f_{36} through f_{40} respectively. Features f_{41} through f_{45} store the relative size of each segment with respect to the image, and are computed as $f_{i+40} = |s_i|/(XY)$ where $i = 1, \dots, 5$.

The hue component of HSV is such that the colors that are 180° apart in the color circle (Fig. 5.11) are complimentary to each other, which means that they add up to ‘white’ color. These colors tend to look pleasing together. Based on this idea, we define two new features, f_{46} and f_{47} in the following manner, corresponding to *average color spread* around the wheel and *average complimentary colors* among the top 5 patch hues. These features are defined as

$$f_{46} = \sum_{i=1}^5 \sum_{j=1}^5 |h_i - h_j|, \quad f_{47} = \sum_{i=1}^5 \sum_{j=1}^5 l(|h_i - h_j|), \quad h_i = \sum_{(x,y) \in s_i} I_H(x, y)$$

where $l(k) = k$ if $k \leq 180^\circ$, $360^\circ - k$ if $k > 180^\circ$. Finally, the rough positions of each segment are stored as features f_{48} through f_{52} . We divide the image into 3 equal parts along horizontal and vertical directions, locate the block containing the centroid of each patch s_i , and set $f_{47+i} = (10r + c)$ where $(r, c) \in \{(1, 1), \dots, (3, 3)\}$ indicates the corresponding block starting with top-left.

5.5.9 Low Depth of Field Indicators

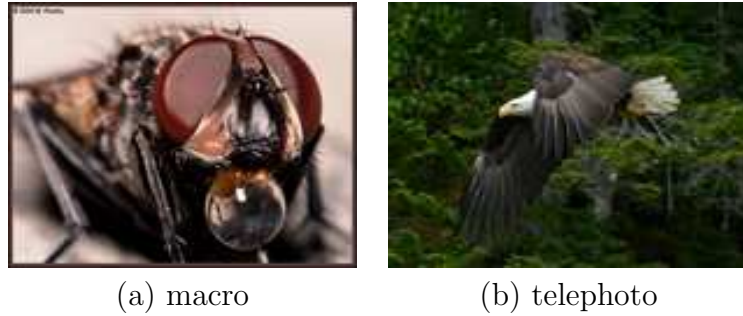


Figure 5.12. Aesthetics ratings are often higher for images with low depth of field. (a) 6.37 out of 7 (b) 6.25 out of 7

Pictures with a simplistic composition and a well-focused center of interest are sometimes more pleasing than pictures with many different objects (see Fig. 5.12). Professional photographers often reduce the depth of field (DOF) for shooting single objects by using larger aperture settings, macro lenses, or telephoto lenses. DOF is the range of distance from a camera that is acceptably sharp in the photograph. A typical camera is an optical system containing a lens and an image screen. The lens creates images in the plane of the image screen, which is *normally* parallel to the lens plane. Denote the focal length of the lens by f and its diameter by a . Denote the aperture f-stop number for this photo by p . Then $f = ap$. Suppose the image screen is at distance d from the lens and the object is at distance s from the lens. If the object is in focus, then the Gaussian thin lens law holds: $\frac{1}{s} + \frac{1}{d} = \frac{1}{f}$. A point closer or farther away from the lens than s is imaged as a circle rather than a point. On the photo, areas in the DOF are noticeably sharper.

By browsing the images and ratings, we noticed that a large number of low DOF photographs, e.g., insects, other small creatures, animals in motion, were given high ratings. One reason may be that these shots are difficult to take, since

it is hard to focus steadily on small and/or fast moving objects like insects and birds. A common feature is that they are taken either by *macro* or by telephoto lenses. We propose a novel method to detect low DOF and macro images. We divide the image into 16 equal rectangular blocks $\{M_1, \dots, M_{16}\}$, numbered in row-major order. Let $w_3 = \{w_3^{lh}, w_3^{hl}, w_3^{hh}\}$ denote the set of wavelet coefficients in the high-frequency (level 3 by the notation in Sec. 5.5.6) of the hue image I_H . The *low depth of field indicator* feature f_{53} for hue is computed as follows, with f_{54} and f_{55} being computed similarly for I_S and I_V respectively:

$$f_{53} = \frac{\sum_{(x,y) \in M_6 \cup M_7 \cup M_{10} \cup M_{11}} w_3(x, y)}{\sum_{i=1}^{16} \sum_{(x,y) \in M_i} w_3(x, y)}$$

The idea here is that the object of interest in a macro shot is usually near the center, where there is sharp focus, while the surrounding is usually out of focus due to low DOF. This essentially means that a large value of the low DOF indicator features tend to occur for macro and telephoto shots.

5.5.10 Shape Convexity

All of the previously discussed features were either related to color, composition, or texture. It is believed that shapes in a picture also influence the degree of aesthetic beauty perceived by humans. The challenge in designing a shape feature lies in the understanding of what kind of shape pleases humans, and whether any such measure generalizes well enough or not. As always, we hypothesize that convex shapes (perfect moon, well-shaped fruits, boxes, windows etc.) have an appeal (positive or negative) different from concave or highly irregular shapes. Let the image be segmented, as described before, and R patches $\{p_1, \dots, p_R\}$ are obtained such that $|p_k| \geq \frac{XY}{200}$. For each p_k , we compute its convex hull, denoted by $g(p_k)$. For a perfectly convex shape, $p_k \cap g(p_k) = p_k$, i.e. $\frac{|p_k|}{|g(p_k)|} = 1$. Allowing some room for irregularities of edge and error due to digitization, we define the *shape convexity* feature f_{56} as follows:

$$f_{56} = \frac{1}{XY} \left\{ \sum_{k=1}^R I\left(\frac{|p_k|}{|g(p_k)|} \geq 0.8\right) |p_k| \right\}$$



Figure 5.13. Demonstrating the *shape convexity* feature. *Left:* Original photograph. *Middle:* Three largest non-background segments shown in original color. *Right:* Exclusive regions of the *convex hull* generated for each segment are shown in white. The proportion of white regions determine the convexity value.

where $I(\cdot)$ is the indicator function. This feature can be interpreted as the fraction of the image covered by approximately convex-shaped homogeneous regions, ignoring the insignificant image regions. This feature is demonstrated in Fig. 5.13. Note that a critical factor here is the segmentation process, since we are characterizing shape by segments. Often, a perfectly convex object is split into concave or irregular parts, considerably reducing the reliability of this measure.

5.6 Feature Selection, Classification, Regression

A contribution of our work is the feature extraction process itself, since each of the features represent interesting aspects of photography regardless of how they aid in classification or regression. We now wish to select interesting features in order to (1) discover features that show correlation with community-based aesthetics scores, and (2) build a classification/regression model using a subset of strongly/weakly relevant features such that generalization performance is near optimal. Instead of using any regression model, we use a one-dimensional support vector machine (SVM) [265]. SVMs are essentially powerful binary classifiers that project the data space into higher dimensions where the two classes of points are linearly separable. Naturally, for one-dimensional data, they can be more flexible than a single threshold classifier.

For the 3581 images downloaded, all 56 features in \mathcal{F} were extracted and normalized to the $[0, 1]$ range to form the experimental data. Two classes of data are chosen, *high* containing samples with aesthetics scores greater than 5.8, and *low* with scores less than 4.2. Note that as mentioned before, only those images that

were rated by at least two unique members were used. The reason for choosing classes with a gap is that pictures with close lying aesthetic scores, e.g., 5.0 and 5.1 are not likely to have any distinguishing feature, and may merely be representing the noise in the whole peer-rating process. For all experiments we ensure equal priors by replicating data to generate equal number of samples per class. A total of 1664 samples is thus obtained, forming the basis for our classification experiments. We perform classification using the standard RBF Kernel ($\gamma = 3.7$, $cost = 1.0$) using the LibSVM package [23]. SVM is run 20 times per feature, randomly permuting the data-set each time, and using a 5-fold cross-validation (5-CV). The top 15 among the 56 features in terms of model accuracy are obtained. The stability of these single features as classifiers are also tested.

We then proceeded to build a classifier that can separate *low* from *high*. For this, we use SVM as well as the classification and regression trees (CART) algorithm, developed at Stanford and Berkeley [22]. While SVM is a powerful classifier, one limitation is that when there are too many irrelevant features in the data, the *generalization performance* tends to suffer. Hence the problem of feature selection continues to dwell. Feature selection for classification purposes is a well-studied topic [17], with some recent work related specifically to feature selection for SVMs. *Filter-based methods* and *wrapper-based methods* are two broad techniques for feature selection. While the former eliminates irrelevant features before training the classifier, the latter chooses features using the classifier itself as an integral part of the selection process. In this work, we combine these two methods to reduce computational complexity while obtaining features that yield good generalization performance: (1) The top 30 features in terms of their one-dimensional SVM performance methods are retained while the rest of the features are *filtered* out. (2) We use *forward selection*, a wrapper-based approach in which we start with an empty set of features and iteratively add one feature at a time that increases the 5-fold CV accuracy the most. We stop at 15 iterations (i.e. 15 features) and use this set to build the SVM-based classifier.

Although SVM produced very encouraging classification results, they were hard to interpret, except for the one-dimensional case. Classifiers that help understand the influence of different features directly are tree-based approaches such as CART. We used the recursive partitioning (RPART) implementation [254], developed at

Mayo Foundation, to build a two-class classification tree model for the same set of 1664 data samples.

Finally, we perform linear regression on polynomial terms of the features values to see if it is possible to directly predict the aesthetics scores in the 1 to 7 range from the feature vector. The quality of regression is usually measured in terms of the *residual sum-of-squares error* $R_{res}^2 = \frac{1}{N-1} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2$ where \hat{Y}_i is the predicted value of Y_i . Here Y being the aesthetics scores, in the worst case \bar{Y} is chosen every time without using the regression model, yielding $R_{res}^2 = \sigma^2$ (variance of Y). Hence, if the the independent variables explain something about Y , it must be that $R_{res} \leq \sigma^2$. For this part, all 3581 samples are used, and for each feature f_i , the polynomials $(f_i, f_i^2, f_i^3, f_i^{\frac{1}{3}}, \text{ and } f_i^{\frac{2}{3}})$ are used as independent variables.

5.7 Empirical Evaluation of Inference

For the one-dimensional SVM performed on individual features, the top 15 results obtained in decreasing order of 5-CV accuracy are as follows: $\{f_{31}, f_1, f_6, f_{15}, f_9, f_8, f_{32}, f_{10}, f_{55}, f_3, f_{36}, f_{16}, f_{54}, f_{48}, f_{22}\}$. The maximum classification rate achieved by any single feature was f_{31} with 59.3%. This is not surprising since one feature is not expected to distinguish between *high* and *low* aesthetics scores, but having accuracy greater than 54%, they act as weak classifiers and hence show some correlation with the aesthetics scores.

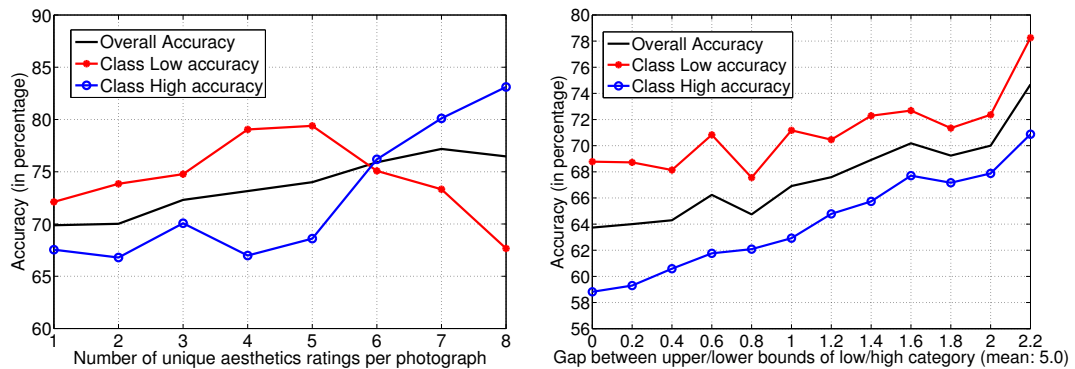


Figure 5.14. *Left:* Variation of 5 – CV SVM accuracy with the minimum number of unique ratings per picture. *Right:* Variation of 5 – CV SVM accuracy with inter-class gap δ .

Coming to the SVM results, the combined filter and wrapper method for feature selection yielded the following set of 15 features: $\{f_{31}, f_1, f_{54}, f_{28}, f_{43}, f_{25}, f_{22}, f_{17}, f_{15}, f_{20}, f_2, f_9, f_{21}, f_{23}, f_6\}$. The accuracy achieved with just these 15 features is 70.12%, with precision of detecting *high* class being 68.08%, and *low* class being 72.31%. Considering the nature of this problem, these classification results are indeed promising. The stability of these classification results in terms of number of ratings are then considered. Samples are chosen in such a way that each photo is rated by at least K unique users, K varying from 1 to 8, and the 5-CV accuracy and precision plotted, as shown in Fig. 5.14. It is observed that accuracy values show an upward trend with increasing number of unique ratings per sample, and stabilize somewhat when this value touches 5. This reflects on the peer-rating process - the inherent noise in this data gets averaged out as the number of ratings increase, converging towards a somewhat ‘fair’ score. We then experimented with how accuracy and precision varied with the gap in aesthetics ratings between the two classes *high* and *low*. So far we have considered ratings ≥ 5.8 as *high* and ≤ 4.2 as *low*. In general, considering that *ratings* $\geq 5.0 + \frac{\delta}{2}$, be (*high*) and *ratings* $\leq 5.0 - \frac{\delta}{2}$ be (*low*), we have based all classification experiments on $\delta = 1.6$. The value 5.0 is chosen as it is the *median* aesthetics rating over the 3581 samples. We now vary δ while keeping all other factors constant, and compute SVM accuracy and precision for each value. These results are plotted in Fig. 5.14. Not surprisingly, the accuracy increases as δ increases. This is accounted by the fact that as δ increases, so does the distinction between the two classes.

Figure 5.15 shows the CART decision tree obtained using the 56 visual features. In the figures, the decision nodes are denoted by squares while leaf nodes are denoted by circles. The decisions used at each split and the number of observations which fall in each node during the decision process, are also shown in the figures. Shaded nodes have a higher percentage of *low* class pictures, hence making them *low nodes*, while un-shaded nodes are those where the dominating class is *high*. The RPART implementation uses 5-CV to prune the tree to yield lowest risk. We used a 5-fold cross validation scheme. With *complexity parameter* governing the tree complexity set to 0.0036, the tree generated 61 splits, yielding an 85.9% model accuracy and a modest 62.3% 5-CV accuracy. More important than the accuracy, the tree provides us with a lot of information on how aesthetics can be related to

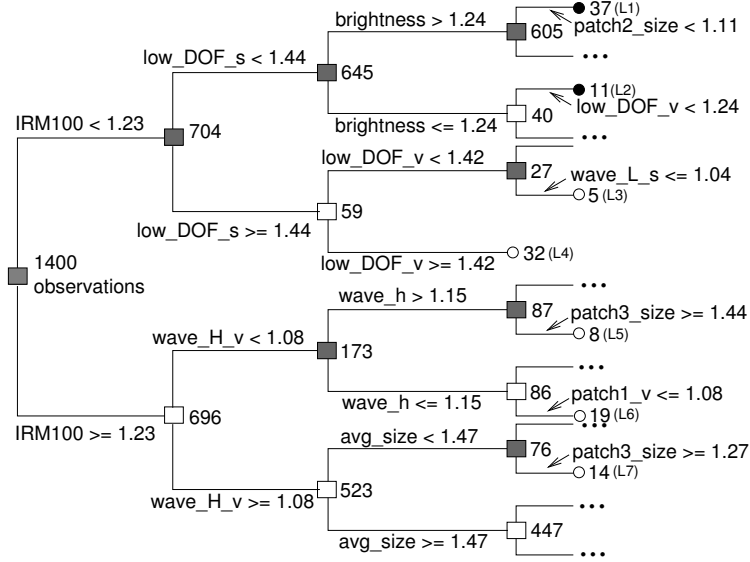


Figure 5.15. The CART tree obtained on the 56 visual features (partial view).

individual features. We do not have the space to include and discuss the entire tree. Let us discuss some interesting decision paths, in each tree, which support our choice of features. The features denoted by $IRM100$ (f_9), and the low DOF indicators for S and V components, respectively (denoted by low_DOF_s (f_{54}) and low_DOF_v (f_{55})), appear to play crucial roles in the decision process. The expected loss at L_3 and L_4 are 0% and 9%, respectively. A large numeric value of the low DOF indicators shows that the picture is focused on a central object of interest. As discussed before, taking such pictures requires professional expertise and hence high peer rating is not unexpected.

Finally, we report the regression results. The variance σ^2 of the aesthetics score over the 3581 samples is 0.69. With 5 polynomial terms for each of the 56, we achieved a residual sum-of-squares $R_{res}^2 = 0.5020$, which is a 28% reduction from the variance σ^2 . This score is not very high, but considering the challenge involved, this does suggest that visual features are able to predict human-rated aesthetics scores with some success. To ensure that this was actually demonstrating some correlation, we randomly permuted the aesthetics scores (breaking the correspondence with the features) and performed the same regression. This time, R_{res} is 0.65, clearly showing that the reduction in expected error was not merely by the over-fitting of a complex model.

5.8 Image Filtering via Aesthetics Inference

We now consider the application of aesthetics inference to real-world image management. The immense popularity of photo-sharing communities (e.g., Flickr, Photobucket, Photo.net) and social-networking platforms (e.g., Facebook, Myspace) has made it imperative to introduce novel media management capabilities, which in turn may help to stay competitive in these crowded markets. In the case of visual media management, areas such as content-based image classification and retrieval [242], automatic annotation [31, 163], and image watermarking [51] for rights management have been extensively studied. Complementing some of these techniques, our goal is to be able to automatically assess high-level visual quality (unlike low-level quality such as noise/quantization level), so as to facilitate *quality-based image management*. Among other things, it can help perform various kinds of image filtering, such as (a) *selecting* high-quality images from a collection for browsing, for front-page display, or as representatives, (b) *enhancing image search* by pushing images of higher quality up the ranks, or (c) *eliminating* low-quality images under space constraints (limited Web space, mobile device, etc.) or otherwise. Visual quality here can be based on criteria such as *aesthetics* (Photo.net, see Fig. 5.16) or *interestingness* (Flickr), and these can be either *personalized* (individuals treated separately), or *consensus-based* (scores averaged over population). A major deterrent to research in this direction has been the difficulty to precisely define their characteristics, and to relate them to low-level visual features. One way around this is to ignore philosophical/psychological aspects, and instead treat the problem as one of data-driven statistical inferencing, similar to user preference modeling in recommender systems [219].

As described previously in this chapter, recent work on aesthetics inference [56] has given hope that it may be possible to empirically learn to distinguish between images of low and high aesthetic value, especially at the extremes of the rating scale. A key result presented in that work is as follows. Using carefully chosen visual features followed by feature selection, a support vector machine (SVM) can distinguish between images rated > 5.8 and < 4.2 (on a 1-7 scale) with 70% accuracy and those rated ≥ 5.0 and < 5.0 with 64% accuracy, images being rated publicly by Photo.net users. There are two key concerns in the context of applica-

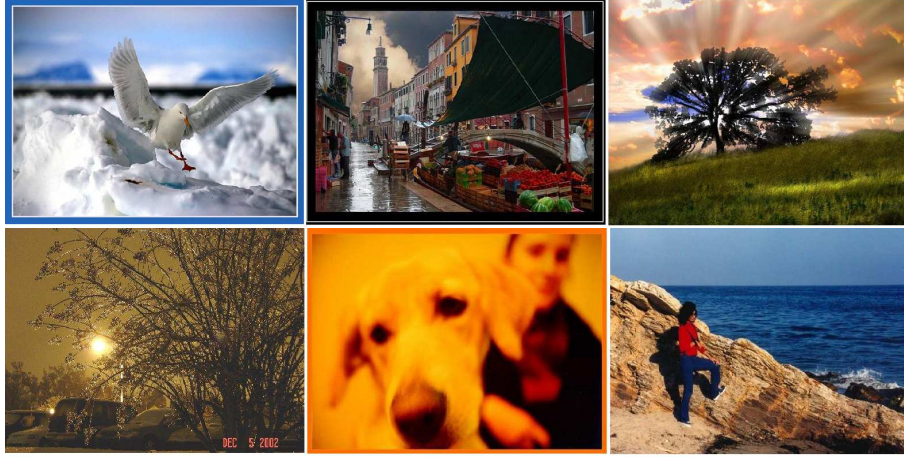


Figure 5.16. Example images from Photo.net where the consensus aesthetics score ≥ 6 (above), and ≤ 4 (below), on 1 – 7.

bility of these results. (1) A 64% accuracy in being able to distinguish ($\geq 5.0, < 5.0$) is not a strong-enough for real-world deployment in selecting high-quality pictures (if ≥ 5.0 implies high-quality, that is). (2) It is unclear how a 70% accuracy on a ($> 5.8, < 4.2$) question can be used to help photo management in any way. Here, we explore the use of the same set of visual features in image filtering, and conclude that despite the moderate classification accuracy, the extracted visual features can help develop very usable aesthetics-related applications. The specific contributions are: (A) Given a set of visual features known to be useful for visual quality, we propose a new approach to exploiting them for significantly improved accuracy in inferring quality. (B) We introduce a weighted learning procedure to account for the trust we have in each consensus score, in the training data, and empirically show consistent performance improvement with it. (C) We propose two new problems of interest that have direct applicability to image management in real-world settings. Our approach produces promising solutions to these problems.

5.9 Regression and Classification Models

Let us suppose that there are D visual features known (or hypothesized) to have correlation with visual quality (e.g., aesthetics, interestingness). An image I_k can thus be described by a feature vector $\vec{X}_k \in \mathbb{R}^D$, where we use the notation $X_k(d)$ to refer to component d of feature vector \vec{X}_k . For clarity of understanding, let

us assume that there exists a *true* measure q_k of consensus on the visual quality that is intrinsic to each I_k . Technically, we can think of this true consensus as the asymptotic average over the entire population, i.e., $q_k = \lim_{Q \rightarrow \infty} \frac{1}{Q} \sum_{i=1}^Q q_{k,i}$, where $q_{k,i}$ is the i^{th} rating received. This essentially formalizes the notion of ‘aesthetics in general’ presented in [56]. This measurement is expected to be useful to the average user, while for those ‘outliers’ whose tastes differ considerably from the average, a personalized score is more useful - a case that best motivates recommender systems with individual user models.

In reality, it is impractical to compute this true consensus score because it requires feedback over the entire population. Instead, items are typically scored by a small subset of the population, and what we get from averaging over this subset is an estimator for q_k . If $\{s_{k,1}, \dots, s_{k,n_k}\}$ is a set of scores provided by n_k unique users for I_k , then $\hat{q}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} s_{k,i}$, where \hat{q}_k is an estimator of q_k . In theory, as $n_k \rightarrow \infty$, $\hat{q}_k \rightarrow q_k$. Given a set of N training instances $\{(\vec{X}_1, \hat{q}_1), \dots, (\vec{X}_N, \hat{q}_N)\}$, our goal is to learn a model that can help predict quality from the content of unseen images.

5.9.1 Weighted Least Squares Regression

Regression is a direct attempt at learning to emulate human ratings of visual quality, which we use here owing to the fact that it is reported in [56] to have found some success. Here, we follow the past work by learning a least squares linear regressor on the predictor variables $X_k(1), \dots, X_k(D)$, where the dependent variable is the consensus score \hat{q}_k . We introduce *weights* to the regression process on account of the fact that \hat{q}_k are only estimates of the true consensus q_k , with less precise estimates being less trustable for learning tasks. From classical statistics, we know that the *standard error of mean*, given by $\frac{\sigma}{\sqrt{n}}$, decreases with increasing sample size n . Since \hat{q}_k is a mean estimator, we compute the weights w_k as a simple increasing function of sample size n_k ,

$$w_k = \frac{n_k}{n_k + 1}, \quad k = 1, \dots, N \quad (5.1)$$

where $\lim_{n_k \rightarrow \infty} w_k = 1$, $w_k \in [\frac{1}{2}, 1)$. The corresponding parameter estimate for squared loss is written as

$$\vec{\beta}^* = \arg \min_{\vec{\beta}} \frac{1}{N} \sum_{k=1}^N w_k \left(\hat{q}_k - \left(\beta(0) + \sum_{d=1}^D \beta(d) X_k(d) \right) \right)^2$$

Given a $\vec{\beta}^*$ estimated from training data, the predicted score for an unseen image I having feature vector X is given by

$$q^{pred} = \beta^*(0) + \sum_{d=1}^D \beta(d) X(d) \quad (5.2)$$

Because weighted regression is relatively less popular than its unweighted counterpart, we briefly state an elegant and efficient linear algebraic [99] estimation procedure, for the sake of completeness. Let us construct an $N \times (D + 1)$ matrix $\mathbf{X} = [\vec{\mathbf{1}} \ \mathbf{Z}^T]$ where $\vec{\mathbf{1}}$ is a N -component vector of ones, and $\mathbf{Z} = [\vec{X}_1 \ \cdots \ \vec{X}_N]$. Let \vec{q} be a $N \times 1$ column matrix (or vector) of the form $(\hat{q}_1 \cdots \hat{q}_N)^T$, and \mathbf{W} is an $N \times N$ diagonal matrix consisting of the weights, i.e., $\mathbf{W} = \text{diag}\{w_1, \dots, w_N\}$. In the unweighted case of linear regression, the parameter estimate is given by $\vec{\beta}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \vec{q} = \mathbf{X}^\dagger \vec{q}$, where \mathbf{X}^\dagger is the *pseudoinverse* in the case of linearly independent columns. The weighted linear least squares regression parameter set, on the other hand, is estimated as below:

$$\vec{\beta}^* = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \vec{q} \quad (5.3)$$

Letting $\mathbf{V} = \text{diag}\{\sqrt{w_1}, \dots, \sqrt{w_N}\}$, such that $\mathbf{W} = \mathbf{V}^T \mathbf{V} = \mathbf{V} \mathbf{V}^T$, we can re-write Eq. 5.3 in terms of pseudoinverse:

$$\begin{aligned} \vec{\beta}^* &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \vec{q} \\ &= ((\mathbf{V} \mathbf{X})^T (\mathbf{V} \mathbf{X}))^{-1} (\mathbf{V} \mathbf{X})^T \mathbf{V} \vec{q} \\ &= (\mathbf{V} \mathbf{X})^\dagger \mathbf{V} \vec{q} \end{aligned} \quad (5.4)$$

This form may lead to cost benefits. Note that the weighted learning process does not alter the inference step of Eq. 5.2.

5.9.2 Naive Bayes' Classification

The motivation for having a naive Bayes' classifier was to be able to complement the linear model with a probabilistic one, based on the hypothesis that they have non-overlapping performance advantages. The particular way of fusing regression and classification will become clearer shortly. For this, we assume that by some predetermined threshold, the (consensus) visual quality scores \hat{q}_k can be mapped to binary variables $\hat{h}_k \in \{-1, +1\}$. For simplification, we make a conditional independence assumption on each feature given the class, to get the following form of the naive Bayes' classifier:

$$Pr(H | X(1), \dots, X(D)) \propto Pr(H) \prod_{d=1}^D Pr(X(d) | H) \quad (5.5)$$

The inference for an image I_k with features \vec{X}_k involves a comparison of the form

$$\hat{h}_k = \arg \max_{h \in \{-1, +1\}} Pr(H = h) \prod_{d=1}^D Pr(X_k(d) | H = h) \quad (5.6)$$

The training process involves estimating $Pr(H)$ and $Pr(X(d)|H)$ for each d . The former is estimated as follows:

$$Pr(H = h) = \frac{1}{N} \sum_{i=1}^N \mathcal{I}(\hat{h}_i = h) \quad (5.7)$$

where $\mathcal{I}(\cdot)$ is the indicator function. For the latter, parametric distributions are estimated for each feature d given class. While mixture models seem appropriate for complicated features (e.g., neither too high nor too low brightness is preferred), here we model each of them using single component Gaussian distributions, i.e.,

$$X(d) | (H = h) \sim \mathcal{N}(\mu_{d,h}, \sigma_{d,h}), \quad \forall d, h, \quad (5.8)$$

where the Gaussian parameters $\mu_{d,h}$ and $\sigma_{d,h}$ are the mean and std. dev. of the feature value X_d over those training samples k that have $\hat{h}_k = h$. Performing weighted parameter estimation is possible here too, although in our experiments we restricted weighting learning to regression only.

5.10 Selection and Elimination Algorithms

In this section, we describe the algorithms that make use of the regression and classification models to select high-quality images and eliminate low-quality images from image collections.

5.10.1 Selecting High-quality Pictures

Equipped with the above two methods, we are now ready to describe our approach to selecting high-quality images. First we need a definition for ‘high-quality’. An image I_k is considered to be visually of high-quality if its estimated consensus score, as determined by a subset of the population, exceeds a predetermined threshold, i.e., $\hat{q}_k \geq HIGH$. Now, the task is to automatically select T high-quality images out of a collection of N images. Clearly, this problem is no longer one of classification, but that of retrieval. The goal is to have high *precision* in retrieving pictures, such that a large percentage of the T pictures selected are of high-quality. To achieve this, we perform the following:

1. A weighted regression model (Sec. 5.9.1) is learned on the training data.
2. A naive Bayes’ classifier (Sec. 5.9.2) is learned on training data, where the class labels \hat{h}_k are defined as

$$\hat{h}_k = \begin{cases} +1 & \text{if } \hat{q}_k \geq HIGH \\ -1 & \text{if } \hat{q}_k < HIGH \end{cases}$$

3. Given an unseen set of N test images, we get predict consensus scores $\{\hat{q}_1, \dots, \hat{q}_N\}$ using the weighted regression model, which we sort in *descending* order.
4. Using the naive Bayes’ classifier, we start from the top of the ranklist, selecting images for which the predicted class is +1, i.e., $\hat{h} = +1$, and $\frac{Pr(H=+1|X(1), \dots, X(D))}{Pr(H=-1|X(1), \dots, X(D))} > \theta$, until T of them have been selected. This filter applied to the ranked list therefore requires that only those images at the top of the ranked list that are also classified as high-quality by the naive

Bayes' (and convincingly so) are allowed to pass. For our experiments, we chose $\theta = 5$ arbitrarily and got satisfactory results.

5.10.2 Eliminating Low-quality Pictures

Here, we first need to define 'low-quality'. An image I_k is considered to be visually of low-quality if its consensus score is below a threshold, i.e., $\hat{q}_k \leq LOW$. Again, the task is to automatically filter out T low-quality images out of a collection of N images, as part of a space-saving strategy (e.g., presented to the user for deletion). The goal is to have high precision in eliminating low-quality pictures, with the added requirement that as few high-quality ones (defined by threshold $HIGH$) be eliminated in the process as possible. Thus, we wish to eliminate as many images having score $\leq LOW$ as possible, while keeping those with score $\geq HIGH$ low in count. Here, steps 1 and 2 of the procedure are same as before, while steps 3 and 4 differ as follows:

1. In Step 3, instead of sorting the predicted consensus scores in descending order, we do so in *ascending* order.
2. In Step 4, we start from the top of the ranklist, selecting images for which the predicted class is -1 (not +1, as before), by a margin. This acts as a two-fold filter: (a) low values for the regressed score ensure preference toward selecting low-quality pictures, and (b) a predicted class of -1 by the naive Bayes' classifier prevents those with $HIGH$ scores from being eliminated.

5.11 Empirical Evaluation of Search Refinement

All experiments are performed on the same dataset obtained from Photo.net that was used in [56], consisting of 3581 images, each rated publicly by one or more Photo.net users on a 1–7 scale, on two parameters, (a) aesthetics, and (b) originality. As before, we use the aesthetics score as a measure of quality. While individual scores are unavailable, we do have the average scores \hat{q}_k for each image I_k , and the no. of ratings n_k given to it. The score distribution in the 1–7 range, along with the distribution of the per-image number of ratings, is presented in Fig. 5.17.

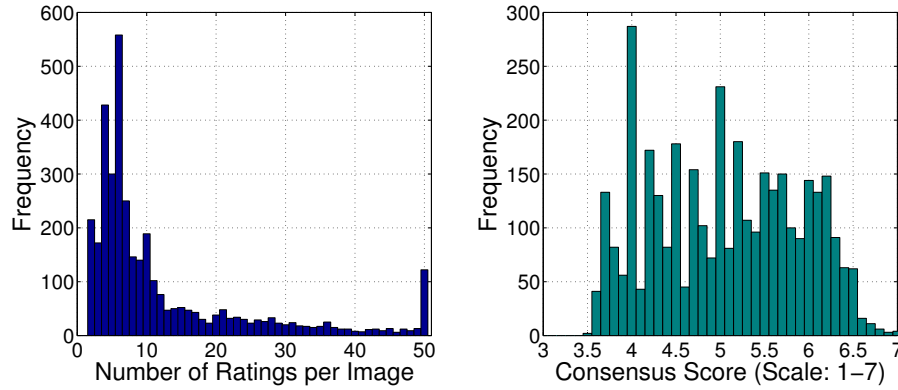


Figure 5.17. Distributions of no. of ratings (left) and scores (right) in Photo.net dataset.

Note that the lowest average score given to an image is 3.55, and that the number of ratings has a *heavy-tailed* distribution. The same 56 visual features extracted in [56] (which include measures for brightness, contrast, depth-of-field, saturation, shape convexity, region composition, etc.) are used here as well, but without any feature selection being performed. Furthermore, nonlinear powers of each of these features, namely their squares, cubes, and square-roots, are augmented with them to get $D = 224$ dimensional feature vectors describing each image.

5.11.1 Selecting High-quality Pictures

Using the procedure described in Sec. 5.10.1, we perform experiments for selection of high-quality images for different values of $HIGH$, ranging over 4.8 – 6.0 out of a possible 7, in intervals of 0.1. In each case, 1000 images are drawn uniformly at random from the 3581 images for testing, and the remaining are used for training the regressor and the classifier. The task here is to select $T = 5, 10$, and 20 images out of the pool of 1000 (other values of $T \leq 50$ showed similar trends), and measure the $precision = \frac{\#(\text{high-quality images selected})}{\#(\text{images selected})}$, where the denominator is a chosen T . We compare our approach with three baselines. First, we use only the regressor and not the subsequent classifier (named ‘Regression only’). Next we use an SVM, as originally used in [56], to do a ($< HIGH$, $\geq HIGH$) classification to get a fixed performance independent of T (named ‘SVM’), i.e., the SVM simply classifies each test image, and therefore regardless of the number of images (T)

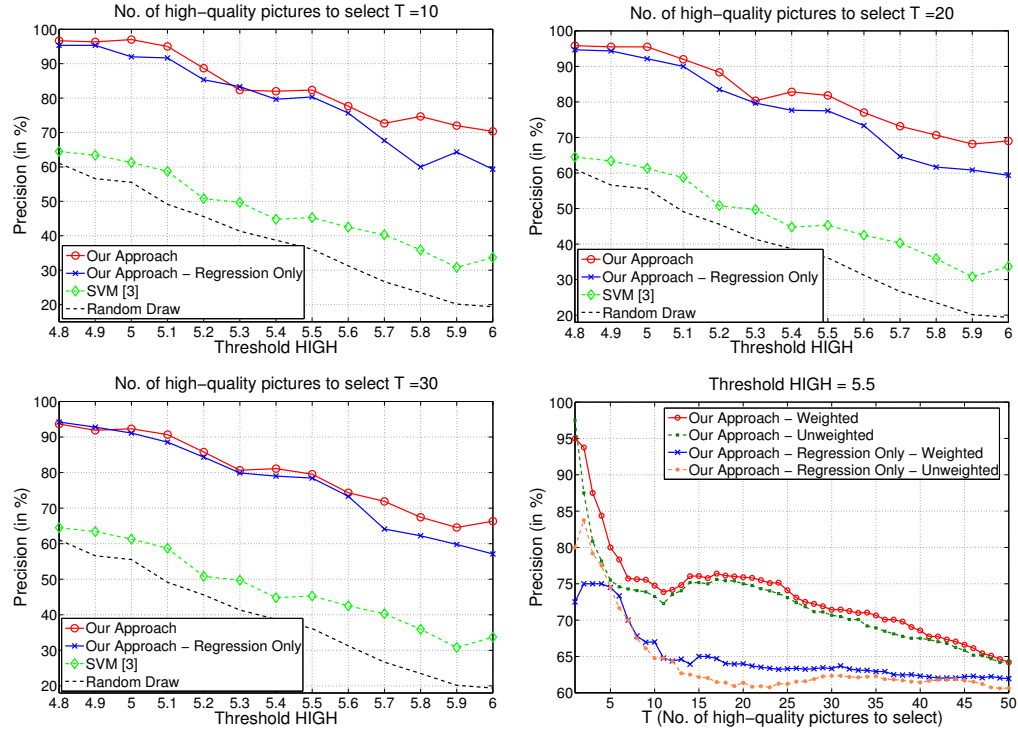


Figure 5.18. Precision in selecting high-quality images, shown here for three selection set sizes, $T = 10, 20$, and 30 . *Bottom-right:* Impact of using weighted model estimation vs. their unweighted counterparts, with $HIGH$ fixed and T varying.

to select, performance is always the same. Finally, as a worst-case bound on performance, we plot the precision achieved on picking any T images at random (named ‘Random Draw’). This is also an indicator of the proportion of the 1000 test images that actually are of high-quality on an average. Each plot in Fig. 5.18 are averages over 50 random test sets.

We notice that our performance far exceeds that of the baselines, and that combining the regressor with the naive Bayes’ in series pushes performance further, especially for larger values of $HIGH$ (since the naive Bayes’ classifier tends to identify high-quality pictures more precisely). For example, when $HIGH$ is set to 5.5, and $T = 20$ images are selected, on an average 82% are of high-quality when our approach is employed, in contrast to less than 50% using SVMs. For lower thresholds, the accuracy exceeds 95%. In the fourth graph (bottom-right), we note the improvement achieved by performing weighted regression instead of giving every sample equal importance. Performed over a range of $HIGH$ values, these

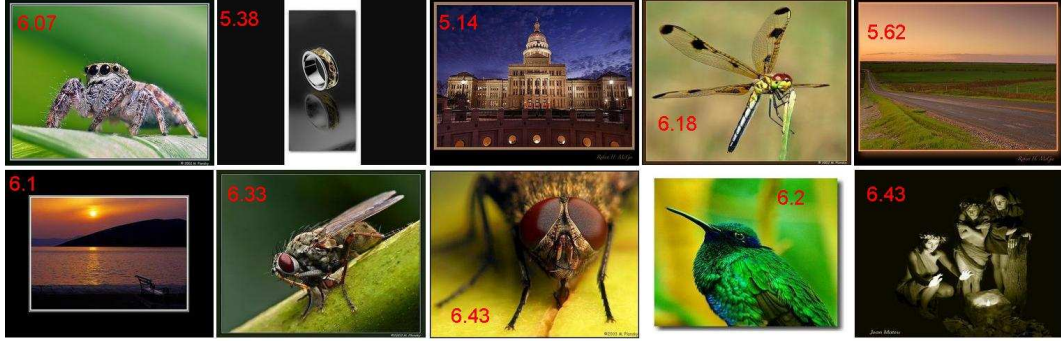


Figure 5.19. A sample instance of $T = 10$ images selected by our approach, for $HIGH = 5.5$. The actual consensus scores are shown in red, indicating an 80% precision in this case.

averaged results confirm our hypothesis about the role of ‘confidence’ in consensus modeling. For illustration, we present a sample instance of images selected by our approach for $T = 10$ and $HIGH = 5.5$, in Fig. 5.19, along with their ground-truth consensus scores.

5.11.2 Eliminating Low-quality Pictures

Here again, we apply the procedure presented in Sec. 5.10.2. The goal is to be able to eliminate T images such that a large fraction of them are of low-quality (defined by threshold LOW) while as few as possible images of high-quality (defined by threshold $HIGH$) get eliminated alongside. Experimental setup is same as the previous case, with 50 random test sets of 1000 images each. We experimented with various values of $T \leq 50$ with consistent performance. Here we present the cases of $T = 25$ and 50, fix $HIGH = 5.5$, while varying LOW from 3.8–5.0. Along with the metric $precision = \frac{\#(\text{low-quality images eliminated})}{\#(\text{images eliminated})}$, also computed in this case is $error = \frac{\#(\text{high-quality images eliminated})}{\#(\text{images eliminated})}$. Measurements over both these metrics, with varying LOW threshold, and in comparison with the ‘Regression Only’, ‘SVM’, and ‘Random Draw’, are presented in Fig. 5.20.

These results are very encouraging, as before. For example, it can be seen that when the threshold for low-quality is set to 4.5, and 50 images are chosen for elimination, our approach ensures $\sim 65\%$ of them to be of low-quality, with only $\sim 9\%$ to be of high-quality. At higher threshold values, precision exceeds 75%, while error remains roughly the same. In contrast, the corresponding SVM

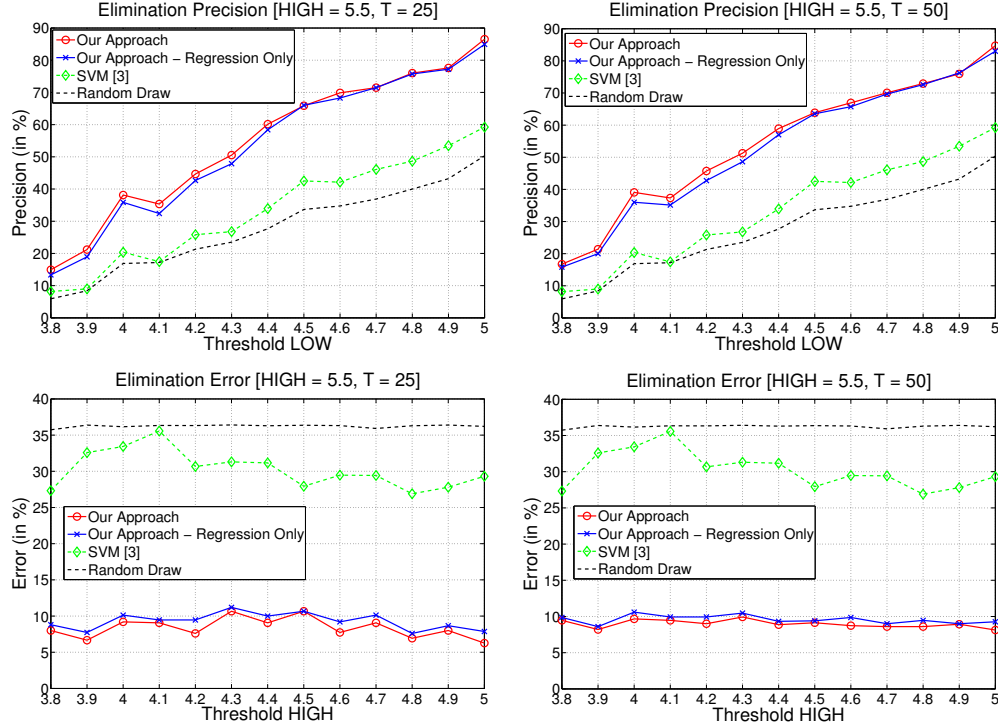


Figure 5.20. Above: Precision in eliminating low-quality images, shown here for two set sizes, namely $T = 25$ and 50. Below: The corresponding errors, made by eliminating high-quality images in the process.

figures are 43% and 28% respectively. We also note that the performance with using naive Bayes' in conjunction with regression does improve performance on both metrics, although not to the extent we see in high-quality picture selection. While not shown here, we found similar improvements as before with using the weighted methods over the unweighted ones.

5.12 Summary

In this chapter, we have explored the topic of data-driven inference of visual quality or 'aesthetic value' of images. Given the highly subjective nature of this problem, our focus was specifically on building data-driven models for aesthetics inference. Owing to minimal prior art, the topic is first explored in great detail, presenting definitions, scope, problems of interest, and datasets available for training. Then, methods for extracting a number of high-level visual features, presumed to have

correlation with aesthetics, are presented. Through feature selection and machine learning, an aesthetics inference model is trained and found to perform moderately on real-world data. The aesthetics-correlated visual features are then used in an image filtering application involving selecting and eliminating images at the high and low extremes of the aesthetics scale respectively, using a novel statistical model. Experimentally, we find this approach to work well in both forms of visual quality based image filtering. The proposed model is thus a favorable candidate for the task of image search result filtering.

Exploiting the Semantic Gap: Designing CAPTCHAs Using Image Search Metrics

Robust image understanding remains an open problem. The gap between human and computational ability to recognizing visual content has been termed by Smeulders et al. [242] as the *semantic gap*. A key area of research that would greatly benefit from the narrowing of this gap is content-based image retrieval (CBIR). Over more than a decade, attempts have been made to build tools and systems that can retrieve images (from repositories) that are semantically similar to query images, which have enjoyed moderate success [58, 242]. While the inability to bridge the semantic gap highlights the limitations of the state-of-the-art in image content analysis, we see in it an opportunity for *system security*. This, and any task that humans are better at performing than the best computational means, can be treated as an ‘automated Turing test’ [274, 261] that tells humans and computers apart. Typically referred to as HIP (Human Interactive Proof) or CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) [25], they help reduce e-mail spam, stop automated blog and forum responses, save resources, and prevent denial-of-service (DoS) attacks on Web servers [194], among others. In general, DoS attacks involve generating a large number of automated (machine) requests to one or more network devices (e.g., servers) for resources in some form, with the goal of overwhelming them and pre-

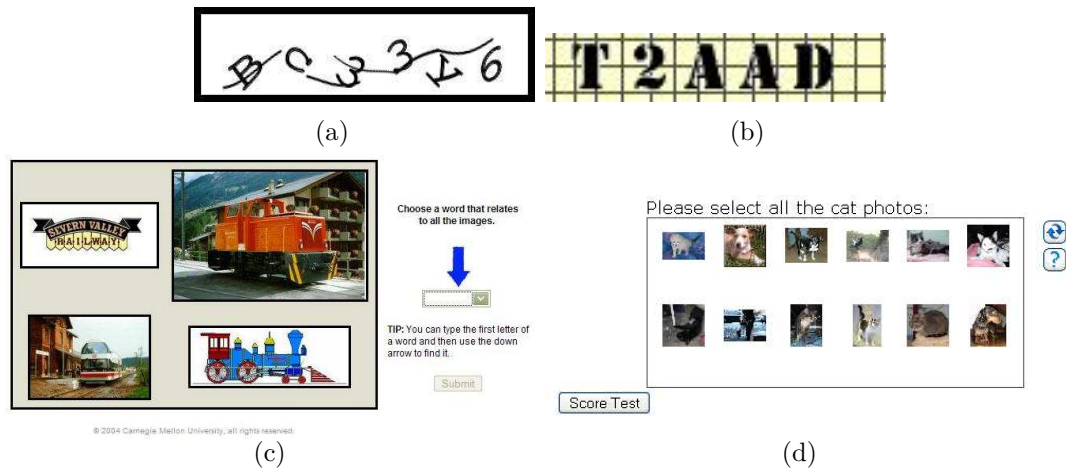


Figure 6.1. Sample CAPTCHAs proposed or in real-world use. (a)-(b) Text-based CAPTCHAs in public use. (c) Image-based CAPTCHA proposed by CMU’s Captcha Project. User is asked to choose an appropriate label from a list. (d) Asirra [76] presents pictures of cats and dogs and asks users to select all the cats.

venting legitimate (human) users from getting their service. In a distributed DoS, multiple machines are compromised and used for coordinated automated attacks, making it hard to detect and block the attack sources. To prevent such forms of attacks and save resources, the servers or other network devices can require CAPTCHA solutions to accompany each request, thus forcing human intervention, and hence, in the very least, reducing the intensity of the attacks. Because CAPTCHAs can potentially play a very critical role in Web security, it is imperative that the design and implementation of CAPTCHAs be relatively foolproof.

There has been sizable research on designing as well as *breaking* CAPTCHAs. In both these efforts, computing research stands to benefit. A better CAPTCHA design means greater security for computing systems, and the breaking of an existing CAPTCHA usually means the advancement of artificial intelligence (AI). While text-based CAPTCHAs have been traditionally used in real-world applications (Yahoo! Mail Sign up, PayPal Sign up, Ticketmaster search, Blogger Comment posting, etc.), their vulnerability has been repeatedly shown by computer vision researchers [195, 252, 38, 196], reporting over 90% success rate. Among the earliest commercial ones, the Yahoo! CAPTCHA has also been reportedly compromised, with a success rate of 35% [241], allowing e-mail accounts to be opened automatically, and encouraging e-mail spam.

In principle, there exist many hard AI problems that can replace text-based CAPTCHAs, but in order to have general appeal and accessibility, recognition of image content has been an oft-suggested alternative [274, 46, 61, 76, 226]. While automatic image recognition is usually considered to be a much harder problem than text recognition (which is a reason for it to be suggested as an alternative to text CAPTCHAs), it has also enjoyed moderate success as part of computer vision research. This implies that a straightforward replacement of text with images may subject it to similar risks of being ‘broken’ by image recognition techniques. Techniques such as near-duplicate image matching [139], content-based image retrieval [242], and real-time automatic image annotation [166] are all potential attack tools for an adversary. One approach that can potentially make it harder for automated attack while maintaining recognizability by humans is *systematic distortion*. A brief mention of the use of distortions in the context of image CAPTCHAs has been made in the literature [46], but this has not been followed up by any study or implementation. Furthermore, while there have been ample studies on the algorithmic ability to handle noisy signals (occlusion, low light, clutter, noise), most often to test robustness of recognition methods, their behavior under strong artificial distortions has been rarely studied systematically.

In this work, we explore the use of systematic image distortion in designing CAPTCHAs, for inclusion in our experimental system called IMAGINATION. We compare human and machine recognizability of images under distortion based on extensive user studies and image matching algorithms respectively. The criteria for a distortion to be eligible for CAPTCHA design are that when applied, they

1. make it difficult for algorithmic recognition, and
2. have minor effect on recognizability by humans.

Formally, let \mathcal{H} denote a representative set of humans, and let \mathcal{M} denote one particular algorithm of demonstrated image recognition capability. We introduce a *recognizability function* $\rho_X(I)$ to indicate whether image I has been correctly recognized by X or not. Thus, $\rho_{\mathcal{H}}(I)$ and $\rho_{\mathcal{M}}(I)$ are human and machine recognizabilities respectively, and we refer to $|\rho_{\mathcal{H}}(I) - \rho_{\mathcal{M}}(I)|$ as the *recognizability gap* with respect to image I . This image can be visually distorted to varying degrees. We

define a distortion function $\delta_y(\cdot)$ that can be applied to a natural image, the degree of distortion being abstractly represented by parameter y . This study focuses on analyzing (a) recognizability, and (b) recognizability gap, of distorted images $\delta_y(I)$, over a large number of natural images. The following are of interest:

- Current state-of-the-art in image recognition typically test and report results on undistorted natural images, and on minor distortions. The ‘breaking’ of an image CAPTCHA, in the absence of distortion, is therefore roughly as likely as the performance of these image recognition techniques.
- On application of a distortion, the image recognition performance is expected to degrade. There has been no comprehensive study on the effect of various artificial distortions on image recognizability.
- Distortion also affects human recognizability of images. It is safe to assume, though, that humans are relatively more resilient to distortion; they can ‘see through’ clutter and fill in the missing pieces, owing to their power of imagination.
- In CAPTCHA design, the goal is to evade recognition by machines while being easily recognizable by humans. It is therefore important to be able to figure out the types and strengths of distortion on images that keep human recognizability high while significantly affecting machine recognizability.

While the primary aim of this work is the systematic design of a security mechanism, the results from the study (See, e.g., Figs. 6.7, 6.8, 6.9, 6.10, and Table 6.3) also reveal to us some of the shortcomings of image matching algorithms, i.e., how the application of certain distortions makes it difficult for even state-of-the-art image matching methods to pair up distorted images with their originals. Furthermore, through large-scale user studies, we are also made aware of the kinds of distortions that make image recognition difficult for humans. These peripheral observations may find use in other research domains.

A Note on CAPTCHA-based Security: Besides specific attempts to break CAPTCHAs by solving hard AI problems, in the recent times, adversaries have used a method which greatly undermines their strength: using humans to solve them. As reported recently [104], humans are being used to solve them, either in

a well-organized manner commercially (in low labor cost regions), or by the use of games and other methods whereby humans are unaware that their responses are being used for malicious purposes. These attempts make it futile to create harder AI problems, because in principle, a CAPTCHA should be solvable by virtually all humans, regardless of their intent. Nonetheless, CAPTCHAs are and will continue to remain deployed until alternate, unbreakable, human identity verification methods become practical. Till then, they should, at the very least, serve to impede the intensity of human-guided breaking of CAPTCHAs. Our work continues the mission of designing CAPTCHAs resilient to *automated* attacks. A key strategy involved in preventing automated attacks is to incorporate random distortions as much as possible, effectively making the space of CAPTCHA problems infinite, thus rendering any attempt to build a dictionary of answers infeasible. Our approach is based on such a strategy.

The rest of this chapter is arranged as follows. In Sec. 6.1, we discuss the metrics for measurement of recognizability under distortion for both humans and machines, and potential candidate distortions that can affect recognizability. In Sec. 6.2, we describe our experimental system IMAGINATION, which we then compare comprehensively with existing CAPTCHAs, in Sec. 6.3. Experimental results on the effect of distortions on human and machine recognizability are presented in Sec. 6.4. We conclude in Sec. 6.5.

6.1 Image Recognizability Under Distortion

Let us assume that we have a collection of natural images, each with a dominant subject, such that given a set of options (say 15), choosing a label is unambiguous. We first define machine/human recognizability concretely, and then discuss distortions that can potentially satisfy the CAPTCHA requirements.

6.1.1 Algorithmic Recognizability

Algorithms that attempt to perform image recognition under distortion can be viewed from two different angles here. First, they can be thought of as methods that potential *adversaries* may employ in order to break image CAPTCHAs. Sec-

ond, they can be considered as intelligent vision systems. Because the images in question can be widely varying and be part of a large image repository, content-based image retrieval (CBIR) systems [242] seem apt. Essentially a memory-based method of attack, the assumption is that the adversary has access to the original (undistorted) images (which happens to be a requirement [25] of CAPTCHAs) for matching with the distorted image presented. While our experiments focus on image matching algorithms, other types of algorithms also seem plausible attack strategies. *Near-duplicate detection* [139], which focus on finding marginally modified/distorted copyrighted images, also seems to be a good choice here. This is part of our future work. *Automatic image annotation* and *scene recognition* techniques [58] also have potential, but given the current state-of-the-art, these methods are very unlikely to do better than direct image-to-image matching.

Recognition of a distorted image $\delta_y(I)$ is thus achieved as follows: Let the adversary have at hand the entire database \mathcal{X} of possible images, i.e., $\forall I, I \in \mathcal{X}$. We can think of the image retrieval algorithm as a function that takes in a pair of images and produces a distance measure $g(I_1, I_2)$ (which hopefully correlates well with their semantic distance). Define a rank function

$$rank_g(I_1, I_2, \mathcal{X}) = \text{Rank of } I_1 \text{ w.r.t. } I_2 \text{ in } \mathcal{X} \text{ using } g(\cdot, \cdot) \quad (6.1)$$

We relax the criteria for machine recognizability, treating image I_1 as recognizable if $rank_g(I_1, I_2, \mathcal{X})$ is within the top K ranks. This is done since the adversary, being a machine, can iterate over a small set K of images quickly to produce a successful attack. Thus, we define *average machine recognizability* under distortion $\delta_y(\cdot)$, where machine in this case is an image retrieval system modeled as $g(\cdot, \cdot)$, as

$$\overline{\rho}_g(\delta_y) = \frac{1}{|\mathcal{X}|} \sum_{I \in \mathcal{X}} \mathcal{I}(rank_g(I, \delta_y(I), \mathcal{X}) \leq K) \quad (6.2)$$

where $\mathcal{I}(\cdot)$ is the indicator function. For our experiments, we consider a very simple image similarity metric, and two well-known and widely used image retrieval systems that use different low-level image representation and compute pairwise image distance in different ways. First, we use the simplest possible image similarity metric; the average of the norm of the pixel-wise difference (PWD) between

the two images. Given two images, the larger image is first scaled to the smaller one to match its dimensions (In our experiments, all test images are of the same dimensions). If the two images are I and I' , then

$$pwd(I, I') = \frac{1}{|I|} \sum_{x,y} \sum_{c \in \{R,G,B\}} (I_c(x, y) - I'_c(x, y))^2 \quad (6.3)$$

where $|I|$ here denotes the total number of pixels in the image. This measure clearly lacks robustness, and is expected to be sensitive even to very small distortions. Second, we employ the Earth Mover's Distance (EMD) [221] (which is essentially the earlier proposed Mallow's Distance [181]) based on global color features and a robust, true distance metric. Finally, we employ the more recent IRM distance which forms the backbone of the SIMPLicity system [278]. This distance performs region segmentation and takes into consideration color, texture, and shape of regions, going on to compute a robust distance between a variable number of region descriptors across a pair of images. In these two cases, color similarity is computed in the CIE-LAB and CIE-LUV spaces respectively, thus adding to their robustness to chromatic distortions. Both methods, while being fairly distinct, have been independently shown to yield good retrieval performance under distortion. The generic distance function $g(\cdot, \cdot)$ is specifically denoted here as $pwd(\cdot, \cdot)$, $emd(\cdot, \cdot)$, and $irm(\cdot, \cdot)$ respectively. Thus, under distortion $\delta_y(\cdot)$, we denote their average recognizability by $\overline{\rho_{pwd}}(\delta_y)$, $\overline{\rho_{emd}}(\delta_y)$, and $\overline{\rho_{irm}}(\delta_y)$ respectively.

6.1.2 Human Recognizability

We measure human recognizability under distortion using a controlled user study. An image I is sampled from \mathcal{X} , subjected to distortion $\delta_y(\cdot)$, and then presented to a user, along with a set of 15 word choices, one of which is unambiguously an appropriate label. The user choice, made from the word list, is recorded alongside the particular image category and distortion type. Since it is difficult to get user responses for each distortion type over all images \mathcal{X} , we measure the average recognizability for a given distortion using the following. If $\mathcal{U}(\delta_y)$ is the set of all

images presented to users subjected to $\delta_y(\cdot)$,

$$\overline{\rho_{\mathcal{H}}}(\delta_y) = \frac{1}{|\mathcal{U}(\delta_y)|} \sum_{I \in \mathcal{U}(\delta_y)} \mathcal{I}(I \text{ is correctly recognized}) \quad (6.4)$$

where \mathcal{I} is the indicator function. The implicit assumptions made here, under which the term $\overline{\rho_{\mathcal{H}}}(\delta_y)$ is comparable to $\overline{\rho_{emd}}(\delta_y)$ or $\overline{\rho_{irm}}(\delta_y)$ is that (a) all users independently assess recognizability of a distorted image (since they are presented privately, one at a time), and (b) with sufficient number of responses, the average recognizability measures converge to their true value.

Assessing Recognizability with User Study: The user study we use in order to measure what we term as the average human recognizability $\overline{\rho_{\mathcal{H}}}(\delta_y)$ under distortion δ_y , is only one of many ways to assess the ability of humans to recognize images in clutter. This metric is designed specifically to assess the usability of CAPTCHAs, and may not reflect on general human vision. Furthermore, the study simply asks users to choose one appropriate image label from a list of 15 words, and recognizability is measured as the fraction of times the various users made the correct choice. While correct selection may mean that the user recognized the object in the image correctly, it could also mean that it was the only choice perceived to be correct, by elimination of choices (i.e., best among many poor matches), or even a random draw from a reduced set of potential matches. Furthermore, using the averaged responses over multiple users could mean that the CAPTCHA may still be unusable by some fraction of the population. While it is very difficult to assess true recognizability, our metric serves the purpose it is used for: the ability of users to pick one correct label from a list of choices, given a distorted image, and hence we use these averaged values in the CAPTCHA design. Furthermore, the user study consists of roughly the same number of responses from over 250 random users, making the average recognizability metric fairly representative. Later in Sec. 6.4, we will see that there is sufficient room for relaxing the intensity of distortions so as to ensure high recognizability for most users, without compromising on security.

Table 6.1. Some features and distortions that affect their extraction

Feature	Affected by	Not Affected by
Local Color	Quantization, Dithering, Luminance, Noise	Cut/rescale
Color Histogram	Luminance, Noise, Cut/rescale	Quantization, Dithering
Texture	Quantization, Dithering, Noise	Luminance, Cut/rescale
Edges	Noise, Dithering	Quantization, Luminance, Cut/rescale
Segmentation & Shape	Dithering, Noise, Quantization	Luminance, Cut/rescale
Interest Points	Noise, Dithering, Quantization	Luminance, Cut/rescale

6.1.3 Candidate Distortions

We look at distortion candidates that are relevant in designing image CAPTCHAs. With the exception of the requirement that the distortion should obfuscate machine vision more than human vision, the space of possible distortions $\delta_y(\cdot)$ is unlimited. Any choice of distortion gets further support if simple filtering or other pre-processing steps are ineffective in undoing the distortion. Furthermore, we avoid non-linear transformations on the images so as to retain basic shape information, which can severely affect human recognizability. For the same reason we do not use other images or templates to distort an image. Pseudo-randomly generated distortions are particularly useful here, as with text CAPTCHAs.

For the purpose of making it harder for machine recognition to undo the effect of distortion, we need to also consider the approaches taken in computer vision for this task. In the literature, the fundamental step in generic recognition tasks has been *low-level feature extraction* from the images [242, 58]. In fact, this is the only part of the recognition process that we have the power to affect. The subsequent steps typically involve deriving mid to high level features representations from them, performing pair-wise image feature matching, matching them to learned models, etc. Because of their dependence on low-level features, we expect them to weaken or fail when feature extraction is negatively affected. Some of the fundamental features and the corresponding distortions (describe below) that typically affect

their extraction, are presented in Table 6.1. For each feature, we consider only well-established extraction methodologies (e.g., SIFT [172] for interest point detection) when deciding which distortions affect them.

We formalize the notion of image distortions as follows. Suppose we have a set of fundamental or ‘atomic’ distortion types (denoted δ), e.g., adjustment of image luminance, quantization of colors, dithering, or addition of noise. These distortions are parameterized (parameter denoted y), so a particular distortion is completely specified by (type, parameter) tuples, denoted δ_y . The set of possible distortions Δ , which is countably infinite if parameter y is discrete, is formalized as follows:

- Atomic distortions $\{Quantize_y(\cdot), Dither_y(\cdot), \dots\} \in \Delta$.
- If $\delta_y(\cdot)$ and $\delta'_y(\cdot) \in \Delta$, then $\delta_y(\delta'_y(\cdot))$ and $\delta'_y(\delta_y(\cdot)) \in \Delta$.

Put in plain words, any combination of an atomic distortion (applied in a specific order) is a new distortion. Here, we list the atomic distortions (and their parametrization) that we considered for this study.

- **Luminance:** Being one of the fundamental global properties of images, we seek to adjust it. Increasing and decreasing ambient light within an image is expected to affect recognizability. A scale factor parameter controls this in the following way. The RGB components of each pixel are scaled by scale factor, such that the average luminance over the entire image is also scaled by this scale factor. Too much or too little brightness are both expected to affect recognizability.
- **Color Quantization:** Instead of allowing the full color range, we quantize the color space for image representation. For each image, we transform pixels from RGB to CIE-LUV color space. The resultant color points, represented in \mathbb{R}^3 space, are subject to k -means clustering with k -center initialization [124]. A parameter controls the number of color clusters generated by the k -means algorithm. All colors are then mapped to this reduced set of colors. A lower number of color clusters translates to loss of information and hence lower recognizability.
- **Dithering:** Similar to half-toning of the printing industry, color dithering is a digital equivalent that uses a few colors to produce the illusion of color

depth. This is a particularly attractive distortion method here, since it affects low-level feature extraction (on which machine recognition is dependent) while having, by design, minimal effect on human vision. Straightforward application of dithering is, however, ineffective for this purpose since a simple *mean filter* can restore much of the original image. Instead, we randomly partition the image in the following two ways:

- Multiple random orthogonal partitions.
- Image segments, generated using k -means clustering with k -center initialization on color, followed by connected component labeling.

In either case, for each such partition, we randomly select y colors (being the parameter for this distortion) and use them to dither that region. This leaves a segment-wise dithering effect on the image, which is difficult to undo. We expect automatic image segmentation to be particularly affected. Distortion tends to have a more severe effect on recognizability at lower values of y .

- **Cutting and Re-scaling:** For machine recognition methods that rely on pixel-to-pixel correspondence based matching, scaling and translation helps making them ineffective. We simply take a portion of one of the four sides of the image, cut out between 10 – 20% from the edge (chosen at random), and re-scale the remainder to bring it back to the original image dimensions. This is rarely disruptive to human recognition, since items of interest occupy the central region in our image set. On the other hand, it breaks the pixel correspondence. Which side to cut is also selected at random.
- **Line and Curve Noise:** Addition of pixel-wide noise to images is typically reversible by median filtering, unless very large quantities are added, in which case human recognizability also drops. Instead, we add stronger noise elements on to the image, at random. In particular, thick lines, sinusoids, and higher-order curves are added. Technically, we do not set the color of these lines and curves to zero; instead, to make detection and removal harder, we reduce the RGB components of each such line or curve by a randomly drawn factor, giving the illusion of being dark but not necessarily zero. The density of noisy lines and curves are controlled by parameter y . Lines and sinusoids

are generated orthogonal to each axis, spaced by density parameter y . For higher order curves, y specifies the number of them to be added, each added at random positions and orientations.

These distortions are by no means exhaustive, as mentioned before. However, they are hand-picked to be representative of distortions that are potentially good candidates. We experimented with each of them individually, and their simultaneous application on images to produce *composite distortions*. None of the atomic distortions by themselves yielded results promising enough to satisfy the requirements. Hence composite distortions were the only way out. We give specific details of the composite distortions that proved effective for CAPTCHA design, in the results section (Sec. 6.4).

6.2 Experimental System: IMAGINATION

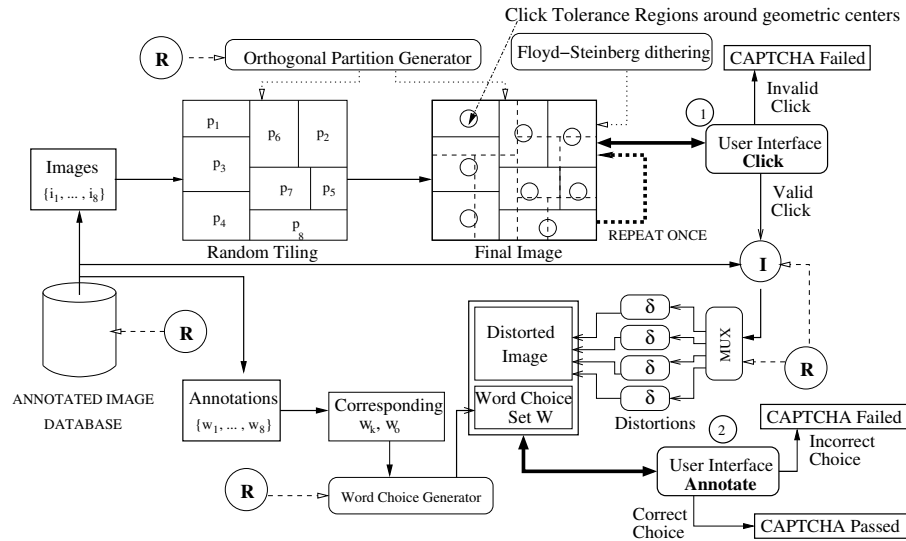


Figure 6.2. Architecture of the IMAGINATION system. The circled ‘R’ components represent randomizations.

So as to put the implications of the distortion experiments into perspective, we first briefly describe our experimental system IMAGINATION¹ (IMAGE Generation

¹A working version of IMAGINATION is at <http://alipr.com/captcha/>.

for INternet AuthenticaTION). The nomenclature is inspired by the fact that the system’s success inherently depends on the imagination power of humans, to help them ‘see through’ distortion and fill in the ‘gaps’ introduced by distortion.

The overall system architecture of our system is shown in Fig. 6.2. Assume the availability of an image repository \mathcal{R} , each labeled with an appropriate word, and an *orthogonal partition generator* that randomly breaks up a rectangle of a given dimension into 8 partitions. The system generates a tiled image, dithers it to make automatic boundary detection hard, and asks the user to select near the center of one of the images. This is the **click** step. On success, an image is randomly sampled, distorted by one of four methods and appropriate parameterizations (discussed in detail in Sec. 6.4), and presented to the user along with a list of word choices, for labeling. This is the **annotation** step. These two steps are detailed below:

- **Click:** A single image is created on-the-fly by sampling 8 images from \mathcal{R} and tiling them according to a randomly generated orthogonal partition. This image is then similarly partitioned twice over. Each time, and for each partition, 18 colors are chosen at random from the RGB space and are used to dither that partition using the two-stage Floyd-Steinberg error-diffusion algorithm [89]. The two rounds of dithering are employed to ensure that there is increased ambiguity in image borders (more candidate ‘edges’), and to make it much more difficult to infer the original layout. An example of such an image is shown in Fig. 6.3. What the user needs to do is select near the physical center of any one of the 8 images. Upon successfully clicking within a tolerance radius r of one of the 8 image centers, the user is allowed to proceed. Otherwise, authentication is considered failed.
- **Annotate:** Here, an image is sampled from \mathcal{R} , a distortion type and strength is chosen (from among those that satisfy the requirements - we find this out experimentally, as described in Sec. 6.4), applied to the image and presented to the user along with an unambiguous choice of 15 words (generated automatically). A sample screenshot is presented in Fig. 6.4. If the user fails in image recognition, authentication is immediately considered failed and re-start from step 1 is necessary.

These two click-annotate steps are repeated once more for added security. The convenience of this interface lies in the fact that no typing is necessary. Authentication is completed using essentially four mouse clicks. The word choices can be translated automatically to other languages if needed.

Word Choice Generator: The word choice generator creates an unambiguous list of 15 words, inclusive of the correct label, in a very simple manner. For this, we make use of a WordNet-based [190] word similarity measure proposed by Leacock and Chodorow [153]. The 14 incorrect choices are generated by sampling from the word pool, avoiding any one that is too similar semantically (determined by a threshold on similarity) to the correct label. A more elaborate strategy was proposed in [61], but we found that for limited pools of words, this simpler strategy was equally effective.

Orthogonal Partition Generator: Optimal rectangle packing (within a larger rectangle), with minimum possible waste of space, is an NP-complete problem. Approximate solutions to this problem have been attempted before, such as in recent work of R.E. Korf [147]. However, waste of space is not an issue for us, nor are rectangles to pack rigid, i.e., linear stretching is allowed. Our approach is as follows.

The full rectangular area is first partitioned vertically or horizontally (chosen randomly) into two equal rectangles. The sub-rectangles so formed are further partitioned recursively, strictly alternating between horizontal and vertical. The point of partition is sampled uniformly at random along a given length. We stop when the required number of sub-rectangles (8 in our experiments) are formed. In Appendix B, we explain in greater detail this process of generating partitions. It is important that the adversary be unable to take advantage of any non-uniformity in the way the partitions are generated. In other words, the center coordinates of sub-rectangles so formed should be drawn from a jointly uniform distribution, such that within the plausible region that an image center may lie, every point is equally probable. In Appendix B, we show that the way we generate the partitions guarantees this uniformity.

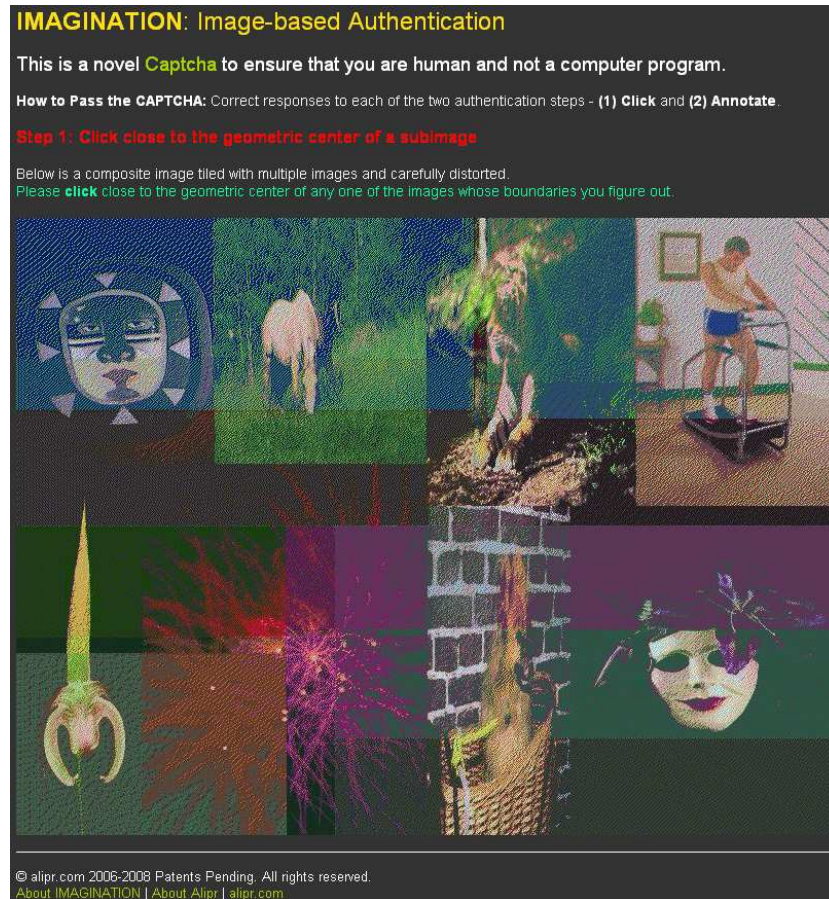


Figure 6.3. Screenshot of the **Click** step of authentication in the IMAGINATION system. The tiled image is randomly partitioned orthogonally and dithered using different color sets, to make it harder for automated identification of the image boundaries. The user must click near the center of one of the images to get past this step.



Figure 6.4. Two screenshots of the **Annotate** step in the IMAGINATION system, where a distorted image is presented, and the user must select an appropriate label from a list of choices.

6.2.1 Vulnerability of ‘Click’ Stage to Attack

In the experiments (Sec. 6.4), we primarily analyze vulnerability of the ‘Annotate’ stage to automatic image recognition. If the adversary has full access to the image

dataset, she should be able to use brute force to align one of the eight images within the tiled image, and determine its center. However, this will be an extremely expensive operation.

For an attack endeavor to be successful, it is essential to be able to find the corner coordinates of one of the 8 images. Exact match with an image in the database is made challenging by the following factors:

- After tiling the images, the full image is re-partitioned and dithered twice (described previously), fragmenting the color composition of each image.
- The images are scaled to fit the generated partitions, not maintaining aspect ratio.

Because partial matches do not reveal the corner coordinates, the only way to get at them is to do a brute-force search over the tiled image, at various scales. This is still subject to the fact that given a perfect alignment, there exists an image similarity metric, which despite the dithering, decisively reveals the match. To get a pessimistic estimate of the threat (from the designer point of view), let us assume the adversary does have such a metric. Let us consider a tiled image of size 800×600 , and that the adversary is attempting to match with one of the 8 images whose top-left corner coincides with the tiled image (Φ_{111} in Appendix B). Its bottom-right corner could be placed anywhere from $(0,0)$ to $(X/2, Y)$, which in this case is $(400, 600)$. Again, assume that the adversary can skip 5 pixels in each dimension without missing the exact match. In our database, there are 1050 images. The number of image matches to be attempted is $400/5 \times 600/5 \times 1050 = 10080000$. Assuming that the matching metric takes $100\mu s$ for each image pair, it will take 1008 seconds, or about 17 minutes, to find one pair of corners.

While this analysis clearly indicates that the brute-force image matching approach to automatically solving the ‘Click’ stage is infeasible, there are some caveats with respect to commercial implementations and actual attempts at breaking them. First, the number of images in the database should be far more than 1050. Second, given the heavy multi-stage distortion applied to the tiled image, a metric for image matching may not always reveal the perfect alignment positions. Third, a more robust image matching algorithm may be necessary, which is

likely to take more than $100\mu\text{s}$ to process, which would mean that attack in this manner will be more expensive than estimated here. On the other hand, more robust methods used in subimage matching [139] may be able to significantly reduce brute-force search. However, such methods are only robust to minimal distortions and limited, aspect-ratio-preserving rescaling of images.

6.2.2 Overall Success Rate of Random Attack

The size of the tiled image in the click stage is fixed at 800×600 . The choice of the tolerance radius r is an important trade-off between ease of use and threat of random attacks. In Sec. 6.4, we empirically show the impact of this choice on users. For now, let us assume that $r = 25$ is a reasonable choice, which corresponds very roughly to one-tenth the width of each contained image. Assuming that we are able to produce dithering and distortions that make it no easier to attack than by random guess, the success rate is approximately $(\frac{8\pi r^2}{800 \times 600} \frac{1}{15})^2$ (see Appendix B), or about 1 in 210,312. which can be considered quite costly for opening one e-mail account, for example. The tiled image, the word choices, and the final distorted image together take about 1 second to generate. For faster processing, a large set of distorted images over varied parameter settings can be pre-generated.

6.3 Comparison with Existing CAPTCHAs

Before quantifying the efficacy of the click-annotate steps of the proposed IMAGINATION system, we draw qualitative comparison with existing CAPTCHA systems, both in public-domain existence and proposed in research publications. Reiterating that their purpose is to authenticate users as human without being disruptive or time-consuming, the basis for comparison among CAPTCHAs broadly includes (a) vulnerability to attacks, and (b) user-friendliness. Vulnerability, leading to the failure of such systems, can be due to one of the following:

1. The AI problem posed is fundamentally solved;
2. Use of cheap labor to solve the problem; and
3. Problematic implementation.

While (3) can be avoided by foolproof design and rigorous testing, and (2) cannot be avoided in principle, neither of them depend on the nature of the CAPTCHA. Our comparison, therefore, focuses on (1) which is the availability of tools to solve the AI problem posed. User-friendliness of such systems can be attributed to the following characteristics:

1. Time taken to solve a problem;
2. Chances of human failure;
3. Culture/language/educational bias; and
4. Accessibility (blind, deaf).

In the case of user-friendliness of a CAPTCHA, all these factors are important, so we consider each of them in the ensuing comparison. In order to make the comparisons concise, we group CAPTCHAs into broad classes, as follows:

- **Text based:** Text characters, typically in English, distorted in various ways (Fig. 6.1.a).
- **Generic image based:** Images of easily recognizable objects, shown to be labeled (Fig. 6.1.c).
- **Speciality image based:** Image classes easily distinguished by humans, hard for machines (Fig. 6.1.d).
- **Knowledge based:** Questions in a language, which require ‘common sense’ responses.
- **Audio based:** For people with vision problems, audio clips are presented for recognition.

Our IMAGINATION system falls roughly within the category of ‘generic image based’, but with some vital differences. In Table 6.2, we compare it with the various classes of CAPTCHAs. We observe that for a majority of factors, our system is favorable compared to the rest. Furthermore, our system has so much randomness that it is not possible to encounter same or similar problems repeatedly, ruling out ‘answer collection’ as an attack strategy. Note that blindness poses a challenge to

Table 6.2. Qualitative comparison of our system with other CAPTCHAs

CAPTCHA Classes:	Text based	Generic image based	Speciality image based	Knowledge based	Our System
<i>Examples</i>	EZ-Gimpy	ESP-PIX [25]	Asirra [76], ARTiFACIAL [226]	NoSpam! [270]	-
<i>Automated Solutions</i>	Yes - [195, 196, 252]	Yes - exact or approx. match [278, 58, 242], Automatic Annotation [166]	Yes - Asirra [98, 86]	Likely	Unlikely (by design)
<i>Randomness</i>	Moderate	Low	Moderate	Moderate	High (multi-stage)
<i>Adversary advantage with dataset access</i>	Word dictionary to prune guesses - [196, 195]	Exact or approx. match with image collection	Exact match - speciality DB like Petfinder [227]	Common-sense datasets, e.g. Cyc [154] helps answer questions	Unlikely (tested assuming dataset is available)
<i>Input modality</i>	Keyboard	Mouse	Mouse	Keyboard	Mouse
<i>Time taken to solve</i>	Quick	Quick	Moderate	Quick	Quick
<i>Chances of human failure</i>	Medium (distortion-attack tradeoff)	Low (no distortion)	Medium (distortion-attack tradeoff)	High (knowledge-dependent)	Medium (distortion-attack tradeoff)
<i>Language bias</i>	Medium (must recognize letters)	Low (via automatic translations)	Low	High (must comprehend sentences)	Low (via automatic translations)
<i>Educational bias</i>	Low	Low	Low	High (knowledge-dependent)	Low

all visual CAPTCHAs. While the ‘audio based’ systems primarily serve to solve this issue, they are somewhat orthogonal in design, and hence are not compared. In the following sections, we also make a more detailed comparison of IMAGINATION with text and image based CAPTCHAs.

6.3.1 Comparison with Text-based CAPTCHAs

Text-based systems, such as the ones shown in Fig. 6.1 a. and b., remain arguably the most widely deployed forms of CAPTCHAs. The AI challenge involved is essentially optical character recognition (OCR), with the additional challenge that the characters are randomly distorted. Thanks to years of research in OCR, hand-writing recognition, and computer vision, a number of research articles, such as [195, 196, 252], have shown that such CAPTCHAs can be solved or ‘broken’, with reportedly over 90% success rate. This is a key motivation for exploring alternate paradigms. The AI challenge posed by our IMAGINATION system is that of image recognition, which is arguably [274] a much harder problem to solve than OCR. Therefore, in principle, IMAGINATION is likely to be more resilient to automated attacks. Furthermore, text-based CAPTCHAs require typing of letters in a given language, say English, and its internationalization may require considerable effort which includes regenerating the CAPTCHA images. With IMAGINATION, the ‘click’ stage is language-independent, and for the ‘annotate’ stage, any standard language translator software can map the options from English to another language.

6.3.2 Comparison with Other Image-based CAPTCHAs

Within the paradigm of image-based CAPTCHAs, there are distinct examples, such as simple image recognition CAPTCHAs [46] which present users with undistorted generic images to be labeled using the provided word lists, Asirra [76], which present images of 12 cats and dogs and users are required to identify the cats among them, and ARTiFACIAL [226] which generates facial images and asks users to pinpoint facial features in them. Problems with presenting undistorted images, or arbitrarily distorted images, in these systems, are

- It is a security requirement of CAPTCHA systems to make data publicly available, in this case the set of labeled images used. Given this, a straightforward pixel-by-pixel matching algorithm should be sufficient to answer the image labeling question in such CAPTCHAs.
- If the dataset is indeed not made available, even then there is a problem.

Real-time automatic image annotation systems such as Alipr [166], or other object recognition systems [58, 242], may be able to tag images at a moderate level of accuracy. This level will then translate into the success rate of attacks, if they are used by the adversary. For the more specialized image CAPTCHAs like Asirra, work on cat detection [86, 98] can go a long way in undermining their effectiveness.

- In case the images are distorted arbitrarily, approximate matching algorithms [278, 58] may be sufficient to match them to their originals, and hence obtain the results.

Our proposed IMAGINATION system has the advantage of posing a hard AI problem (image recognition) while avoiding the pitfalls of the other image recognition based systems. Distortions are applied so that presented images cannot be matched exactly to the image dataset, and these distortions are generated in a controlled manner such that approximate matching methods cannot be successfully applied. While researchers in computer vision have had success in image recognition for specific images classes, such as cats [86], recognition of generic image classes is considered a challenging open problem that is unlikely to be solved in the near future. Because our system works with an unrestricted range of image categories, the threat of this AI problem getting solved sometime soon is extremely low.

6.4 Experimental Results

Large scale experiments were conducted using our publicly available IMAGINATION system², as well as our internal testbed. We obtained empirical results for both the **click** and the **annotate** steps, based on actual usage. We describe the setup and results below.

6.4.1 Stage 1: Click

The main variable component of this stage is the choice of r , the radius of tolerance around each image center that is considered a valid click. We therefore wanted to

²<http://alipr.com/captcha/>

see whether valid human users were able to click near the geometric centers or not, and if so, how near or far they clicked.

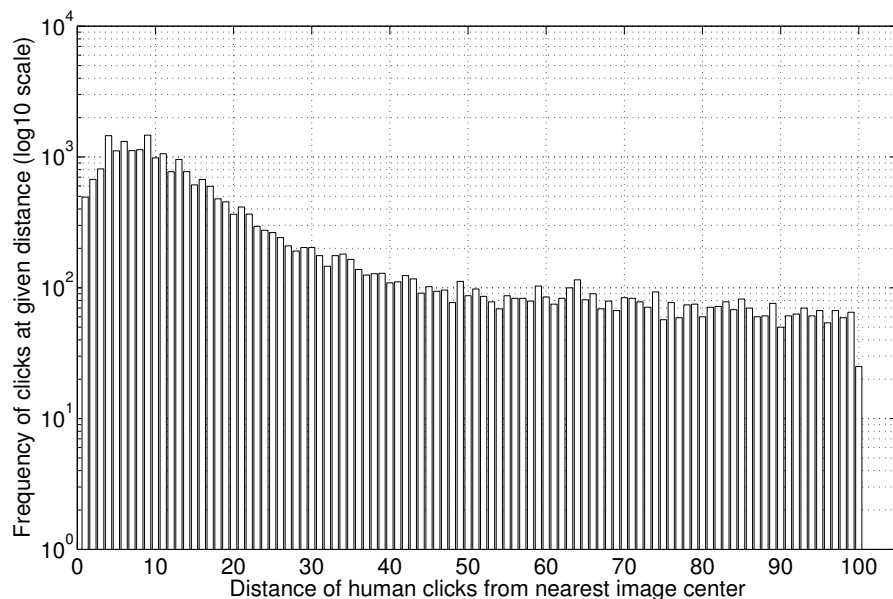


Figure 6.5. Plot of distribution of distances of user clicks from the nearest image centers within the composite images. This distribution (26,152 points), plotted in base-10 log scale, shows that a majority of the clicks are within a short distance of a center.

For this, we used the click data obtained from visitors to our public demo. Since there was evidence that a fraction of users attempted automated attacks on the system by randomly generating coordinates and trying them out, we had to denoise the data such that the majority of the clicks were genuine attempts at succeeding in the task. To achieve this, we randomly picked a single click data for each unique IP address. This way, multiple automated clicks from one machine will have been eliminated from consideration. After denoising the data, we had 26,152 data points corresponding to as many unique IP addresses.

The distribution of the distance of the human clicks from their nearest image centers is shown in Fig. 6.5. We see that a majority of the clicks are in the vicinity of a valid image center, adding to the confidence that this step of the CAPTCHA is a reasonable one for humans. In order to choose a tolerance radius r , useful reference graphs are those plotted in Fig. 6.6. Here we see the empirical distribution of success rates over this set of user clicks, as r is varied. Assuming an attack strategy

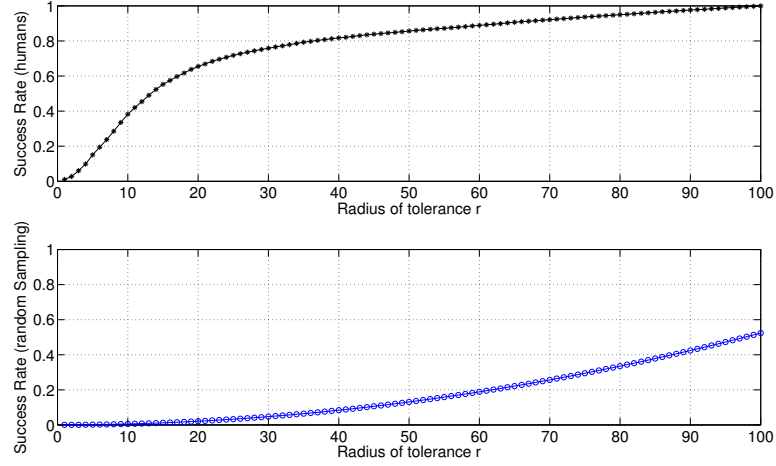


Figure 6.6. Variation of success rates by human (above) and automated randomly sampled clicks (below) with varying tolerance radius r . This graph helps to choose r given a desired trade-off between human ease of use and insulation from attacks.

that involves uniformly sampling at random a pair of coordinates to artificially click at, its success rates are only dependent on r , and these are plotted below. Together, these two plots help determine a value that gives desirable security as well as usability. We can see that 25, which leads to a 70% user success rate and very low random attack success rate, is one good choice and hence is currently used in the IMAGINATION demo. The two outcomes of this experiment were (a) the verification of plausibility of this step, and (b) the selection of a desirable value for parameter r .

6.4.2 Stage 2: Annotate

The experiments related to the annotate step consisted of distorting images and measuring human and machine recognizability, over a set of 1050 Corel images covering 35 easily identifiable categories. Machine recognizabilities was based on the similarity measures PWD, EMD, and IRM (detailed in Sec. 6.1). Human recognizability was measured based on a user study consisting of over 250 individuals, receiving over 4700 responses. The user study consisted of presenting distorted images and a list of 15 words to each user (See Fig. 6.4), allowing them to select an appropriate label, or choose ‘I cannot recognise’ (enabled only during

experimentation). Recognition is considered failed if the latter is chosen, or if an incorrect label is chosen. The following summarizes recognizabilities of humans and machines, and their *recognizability gap*.

6.4.2.1 Atomic Distortions

We first analyzed results obtained from the application of atomic distortions on images. In particular, the effect of luminance adjustment, noise addition, color quantization, and dithering, each in isolation, were studied. For the latter two distortions, cut/rescale was also applied for comparison. These results are presented in Figures 6.7, 6.8, 6.9, and 6.10 respectively. Dithering here is based on orthogonal block partitioning. In each case, the range of values for which human recognizability exceeds 0.9 are shown within Magenta colored dashed lines. They help understand how human and machine recognizabilities contrast.

When pixel correspondence is unaffected, the pixel-wise distance (PWD) performed quite well. However, with the cut/rescale addition, this correspondence is broken and we see significant degradation of PWD's performance (Fig. 6.9 and 6.10). In general IRM shows well-balanced performance, making it a good general-purpose attack tool. Note also that in all these atomic distortion cases, the range where human recognizability is high, at least one of the machine-based methods show high recognizability as well. From this observation, we conclude that any one atomic distortion, does not provide the requisite security from attacks while still being able to maintain human recognizability. This leads us to searching the space of composite distortions. Nonetheless, the results of atomic distortion give a clear insights and help build intuitions about how to combine them effectively.

6.4.2.2 Composite Distortions

An exhaustive search for composite distortions is prohibitively expensive. One may be able to think of algorithmic means to arrive at a composite distortion that satisfies the image CAPTCHA requirements. For example, if atomic distortions are considered analogous to features in a learning problem, then forward-backward selection [18] seems to be an appropriate choice, adding and removing atomic distortions (ordered), testing recognizability, and stopping on satisfactory performance.

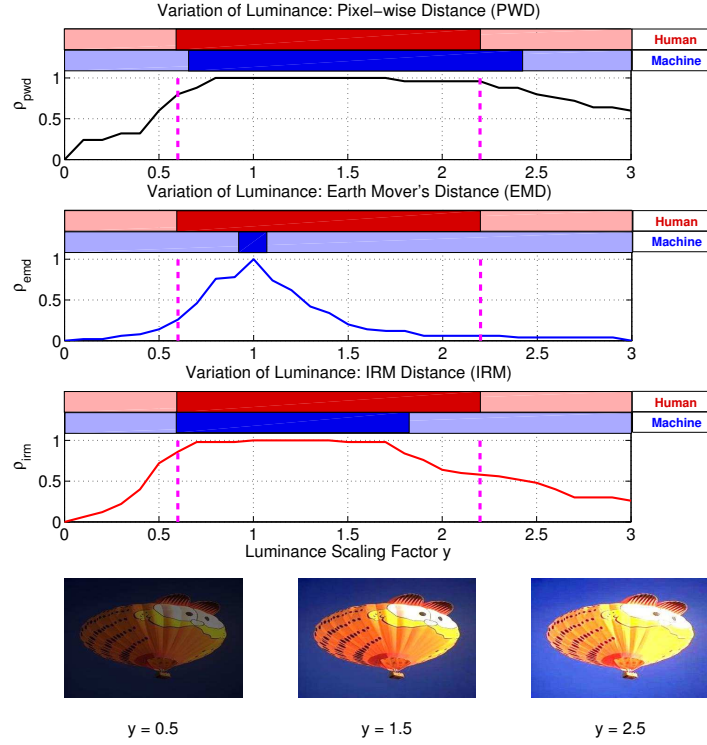


Figure 6.7. Variation of average machine recognizability with change in luminance scaling factor. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$.

Bottlenecks to systematic search for acceptable composite distortions are:

- **Search space is large:** Not only are there many possible atomic distortions, they are also parameterized. Each add/remove step need also iterate over the possible parameter values. For each distortion-parameter pair, machine recognizability needs to be measure over multiple test images.
- **Humans in the loop:** The search space being so large, what is even more problematic is measuring human recognizability at each step. This step would require feedback from multiple users over a multitude of images.
- **Lack of Analytical Solution:** It is difficult to formulate it theoretically as an optimization problem, without which analytical solutions are not possible.

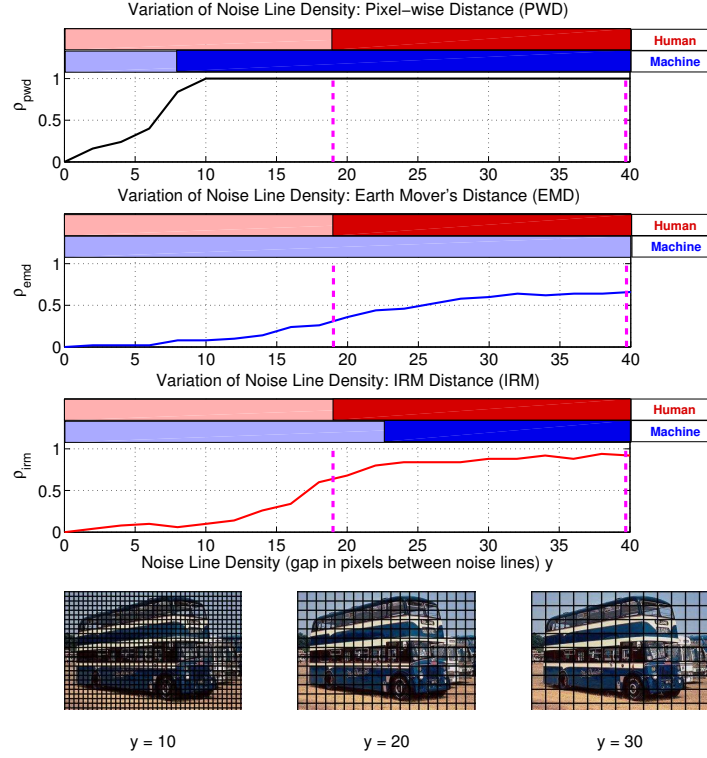


Figure 6.8. Variation of average machine recognizability with change in density of noisy lines added, represented in pixels specifying the gap between consecutive lines. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$.

Instead, we heuristically selected permutations of the atomic distortions and experimented with them. Based on preliminary investigation, four composite distortions seemed particularly attractive, and we conducted large-scale experimentation on them.

Detailed description of each of the four chosen composite distortions are presented in Table 6.3, along with the corresponding experimental results. Each of them are controlled by parameters DITHERPAR, which controls the extent of dithering, and DENSEPAR, which controls the density of noise elements added. To better visualize the recognizability gap as well as make the problem harder, the three types of machine recognition are combined together in the following way. If

Table 6.3. Four Distortions that are part of the IMAGINATION System

Distortion Steps	Human Recognizability	Machine Recognizability (PWD+EMD+IRM)
<ol style="list-style-type: none"> 1. Do k-center/k-means segmentation ($k=15$). 2. Use cluster centroids to quantize image. 3. Create image partitioning using the Orthogonal Partition Generator. 4. Dither each <i>block</i> with DITHERPAR randomly drawn colors. 5. Draw DENSEPAR lines parallel to each axis, <i>randomly spaced</i>. 6. Do 10-20% cut/rescale on randomly chosen side. 		
<ol style="list-style-type: none"> 1. Do k-center/k-means segmentation ($k=15$). 2. Use cluster centroids to quantize image. 3. Create image partitioning using the Orthogonal Partition Generator. 4. Dither each <i>block</i> with DITHERPAR randomly drawn colors. 5. Draw DENSEPAR lines parallel to each axis, <i>equally spaced</i>. 6. Do 10-20% cut/rescale on randomly chosen side. 		
<ol style="list-style-type: none"> 1. Do k-center/k-means segmentation ($k=15$). 2. Use cluster centroids to quantize image. 3. Create image partitioning using the Orthogonal Partition Generator. 4. Dither each <i>block</i> with DITHERPAR randomly drawn colors. 5. Draw DENSEPAR <i>third-order curves</i>, 1-3 pixels thick, <i>randomly positioned</i>. 6. Do 10-20% cut/rescale on randomly chosen side. 		
<ol style="list-style-type: none"> 1. Do k-center/k-means segmentation ($k=15$). 2. Use cluster centroids to quantize image. 3. Perform connected component labeling to get image segments. 4. Dither each <i>segment</i> with DITHERPAR random colors. 5. Draw DENSEPAR <i>sinusoids</i> with axes parallel to each axis, <i>randomly spaced</i>. 6. Do 10-20% cut/rescale on randomly chosen side. 		

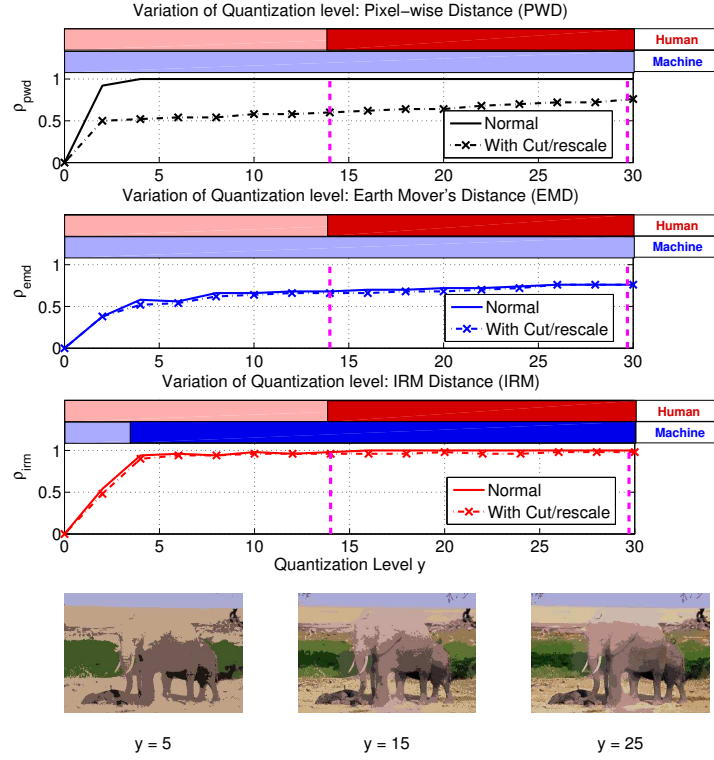


Figure 6.9. Variation of average machine recognizability with change in quantization level, specified in terms of the number of color clusters generated and (centroids) used for mapping. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$, and we show the cut/rescale case here.

any one of PWD, EMD, or IRM recognizes an image, it is considered as successful machine recognition. We find that for a limited range of parameter values in each of them, human recognizability is high (exceeds 0.9) while machine recognizability is low (below 0.1). These distortion type and parameter value/range combinations are appropriate for inclusion into our experimental system IMAGINATION. The few cases where machine recognizability exceeds human recognizability are also interesting and worth exploring, but that is beyond the scope of this work.

To further the investigation and help design the IMAGINATION system better, we studied the trends of human recognizability from the user responses. Figure 6.11 presented the variation of recognizability with parameter values across all four

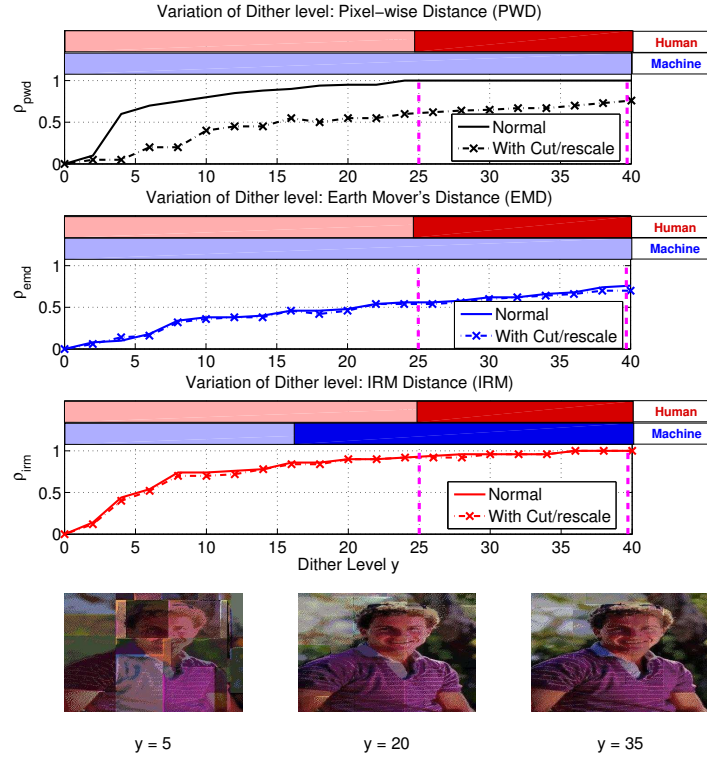


Figure 6.10. Variation of average machine recognizability with change in dithering level, specified in terms of the number of colors available for dithering each partition. Human recognizability is high within the Magenta lines. The red and blue regions above the graphs show ranges (and overlaps) of human and machine recognizability. Dark and light shades indicate ‘high’ and ‘low’ recognizability respectively. Machine recognizability is considered ‘high’ for $\rho \geq 0.8$, and we show the cut/rescale case here.

distortion types, revealing the general trend associated with DITHERPAR and DENSEPAR regardless of the distortion type. More specifically, a greater number of dithering colors tend to help humans recognize image content better, while greater quantities of noise hinder their recognition. Figure 6.12 reveals yet another aspect of the recognition process, namely the average human recognizability per concept, taken over varying distortion type and strength. As can be seen, some concepts (e.g., parade, vegetable) are inherently harder to identify than others.

The results we presented here are over-optimistic from the point of view of attacks. This is because human recognizability only involves identifying the entity and not ‘matching’ any specific pair of images. If we increase the number of

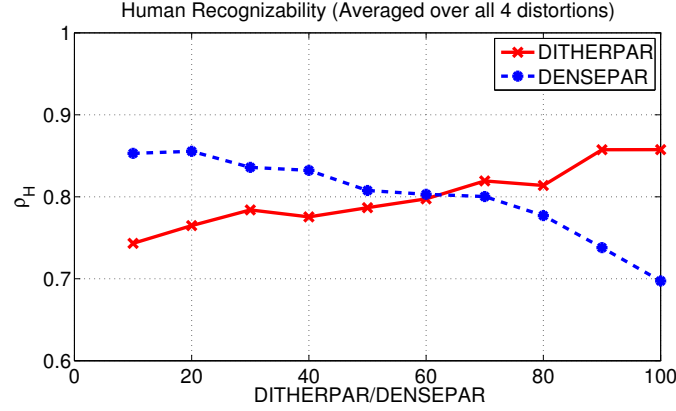


Figure 6.11. Overall variation of human recognizability with dithering parameter DITHERPAR and noise density parameter DENSEPAR, taken across all four composite distortion methods.

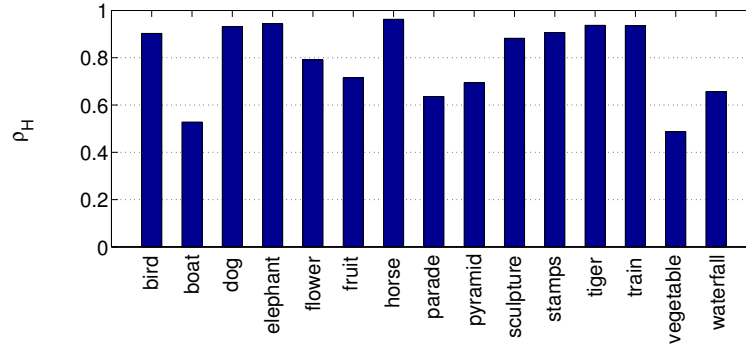


Figure 6.12. Overall variation of human recognizability with the image concept, taken across all four composite distortions and their parameter values; 15 most frequently sampled concepts are shown here.

images in the repository \mathcal{R} , machine recognizability is bound to suffer, while human recognizability should remain at about the same level as reported here. A real-world system implementation will have many more than 1050 in its repository, and will thus be more secure. Also note that with a 15 word choice list, the distortions never need to reduce machine recognizability to less than $1/15$, since randomly selecting a word without even considering the image would yield a $1/15$ chance.

6.5 Summary

In this chapter, we have presented a novel way to distinguish humans from machines by an image recognition test, one that has far-reaching implications in Web secu-

rity. The key point is that image recognition, especially under missing or pseudo information, is still largely unsolved, and this *semantic gap* can be *exploited* for the purpose of building better CAPTCHA systems than the vulnerable text-based CAPTCHAs that are in use today. We have explored the space of systematic distortions as a means of making automated image matching and recognition a very hard AI problem. Without on-the-fly distortion, and with the original images publicly available, image recognition by matching is a trivial task. We have learned that atomic distortions are largely ineffective in reducing machine-based attacks, but when multiple atomic distortions combine, their effect significantly reduce machine recognizability.

Our study, while in no way encompassing the entire space of distortions (or algorithms that can recognize under distortion), presents one way to understand the effects of distortion on the recognizability of images in general, and more specifically to help design image CAPTCHA systems. Furthermore, it attempts to expose the weaknesses of low-level feature extraction to very simple artificial distortions. As a bi-product, an understanding of the difference in recognizability of algorithms and humans under similar conditions also provides an opportunity for better feature extraction design.

Conclusions and Future Research Directions

In this dissertation, I have explored novel ways to improve content-based image search and automatic tagging with the help of statistical learning. My broad goal was two-fold, (1) to contribute toward making the content-driven search paradigm for images more accessible to the masses by removing some existing hurdles, and (2) to enhance the search experience through innovative new approaches.

The former has been achieved with new models and algorithms for automatic tagging which push the state-of-the-art in accuracy and efficiency. I have shown that these automatically generated tags can be very effectively used in a number of realistic image search modalities. Further, real-world usage requires adaptability to changes. There has been demonstrated success in developing algorithms for adapting to changing context, changing personalities (personalization) and changes over time. Much of these topics, while arguably critical to the core of content-based image search, have not been studied prior to this work.

For the latter, I have proposed to infer visual quality from image content, aimed at enhancing the image search experience. For this, I have presented learning approaches for inferring aesthetic value of images using novel visual features correlated with aesthetics, and empirically found them to be promising. The use of aesthetics inference in selecting high-quality images from search result pools, or eliminating low-quality ones, has been experimentally shown to be very effective. Given the negligible prior art, these contributions are among the first on the topic.

As a peripheral contribution, I have explored the use of image search techniques in designing CAPTCHAs, which are hard AI problems aimed at distinguishing humans from machines based on their response. On one hand, this dissertation pushes the state-of-the-art in the hard AI problem of image semantics recognition by proposing improved algorithms for image search and tagging. On the other hand, it assumes that the frontiers will continue to be pushed and eventually make this hard AI problem easy to solve, rendering image recognition based CAPTCHAs unusable. Based on this assumption, I use state-of-the-art image search metrics to design image CAPTCHAs so as to make them more attack resistant, should the adversary use such techniques to launch attacks. The topic of systematic design of image CAPTCHAs has not been previously explored.

7.1 Future Research Directions

While a number of novel contributions are made in this dissertation, they are in no way complete solutions. The problems I deal with are relatively open-ended, with a lot of scope for incremental improvement, or even for adopting radically new approaches. Here, I summarize potential directions that can be explored further.

- With the proposed statistical models, automatic image tagging accuracy and efficiency has improved to unprecedented levels. However, on an absolute scale, these numbers are still low, especially on the accuracy front. While it is hard to measure true recall, precision can be potentially pushed to a much higher level with better algorithms. A higher precision would mean greater reliability on the tags generated, and hence improved image search results.
- The proposed meta-learning algorithms for adapting image tagging to different kinds of changes shows promising results. However, it depends on a moderately performing underlying annotation algorithm. I have experimented with two such state-of-the-art annotation algorithm, but many others have been proposed in the past. It remains to be seen if the meta-learning approach is robust to a wide range of annotation algorithms or not. This can only be made possible by experimenting with a large number of previously proposed annotation systems, using a set of real-world datasets.

- While the proposed approach to aesthetics inference is one of the first attempts of its kind, empirical performance is found to be only moderately good. There is a lot of scope to innovate in the visual feature extraction front. With better, possibly more sophisticated features, a statistical model has a greater chance of finding correlations, in the feature space, to aesthetics. On the other hand, the statistical modeling part can improve as well, taking into consideration the fact that aesthetics is highly subjective. Eventually, to robustly handle the issue of subjectivity, models should be built on a per-person basis, on condition that sufficient data is available. Furthermore, subjectivity is such that it changes over time, so a personalized model of aesthetics should ideally adapt over time as well.
- The use of image search techniques to design robust CAPTCHAs is a novel attempt at harnessing multimedia search technology for network security. Yet, such a design cannot be guaranteed to be foolproof because it makes assumptions about the future, i.e., what tools an adversary might use to attack image CAPTCHAs. This is clearly a limitation with all forms of CAPTCHAs. The best possible strategy is therefore to ensure that all currently available attack tools be tested with. To achieve this, a large-scale study involving various relevant image analysis and computer vision techniques can be conducted. While this can be a tedious exercise, it is the only way to claim strong security guarantees within the current context.

7.2 The Future of Image Search

The most popular image search tools currently available are the Web-based search engines such as those of Yahoo! and Google, which rely heavily on surrounding text to generate results. Content based image search is still not widely deployed for public use, barring a few research efforts, e.g., Alipr [3]. One issue with deployment is that image processing is computationally expensive, which makes rapid querying and updating of the index of billions of images challenging. There is a trade-off between sophistication and speed of the image recognition algorithms. With rapidly growing computing power, this may become less of an issue in the future.

Another issue with large-scale use of content-based search is the barrier with using images as query modalities. It is arguably easier to type in search phrases with a keyboard than to find an image from a stored location which describes user intent well. There are a few ways in which researchers can potentially make content-based image search more accessible. Some of these are already in place, and others are in the offing. But none of these are under wide-scale deployment.

- **Annotation driven search:** This dissertation explored algorithms for annotation driven image search, as a way to make search more accessible. The idea is to not change query modality at all, i.e., to stick to text queries. Instead, in the background, the images are automatically tagged using content-based approaches. This will increase the coverage of tagged images, and potentially remove noisy tags extracted from surrounding text.
- **Cell phone driven search:** With the widespread use of camera-equipped cell phones, it is now easy to take a picture on the go. This allows the user intent to be caught on camera more easily, as and when something is observed. With new Web-enabled cell phones like the iPhone 3G, it is possible to send the query image right away to a server, and get back results instantaneously. With the general market-share of cell phone based search increasing rapidly, this image search modality has great growth potential.
- **Sub-image search:** If we ignore for a moment the issue with specifying image examples as queries, there is still the problem of conveying user intent with a full image, which may contain various entities. In order to be more specific with user queries, the search interface can allow the selection of one or more sub-regions. If we consider each sub-region to be a word, and allow the query to span multiple images, this is then equivalent to specifying text queries with multiple keywords. The idea of sub-image search has been around for a while, but with the availability of touch-screen cell phones and other computing devices, acceptable public use seems more reasonable now.

Many years have passed since the days of QBIC [87], and we still do not have large-scale deployment of content-based image search. The hope is that we are reaching a stage in computer software and hardware advancement where this will soon become reality.

Trends and Impact of Image Retrieval Research

As part of an effort to understand the field of content-based image retrieval (CBIR) and automatic annotation better, we have compiled research trends in image retrieval using Google Scholar’s search tool and its computed citation scores. First, we present recent publication trends related to image retrieval research. Graphs for publication counts and citation scores have been generated for (1) sub-fields of image retrieval, and (2) venues/journals relevant to image retrieval research. Next, we note that image retrieval has likely caused quite a few otherwise-unrelated fields of research to be brought closer together than in the past. To help understand this better, we present an analysis of the impact that various research communities have had on each other in the process of finding solutions to the core problems in image retrieval.

A.1 Publication Trends

We analyze recent publication trends in content-based image retrieval and annotation via two exercises, with Google Scholar as aid. The first of these is an analysis of which venues and journals have carried the most image retrieval related work and what the impact is, and which sub-topics generated the most publication count and impact in the last five years. The second one involves generating subtopic-wise time-series capturing trends in publication over the last eleven years.

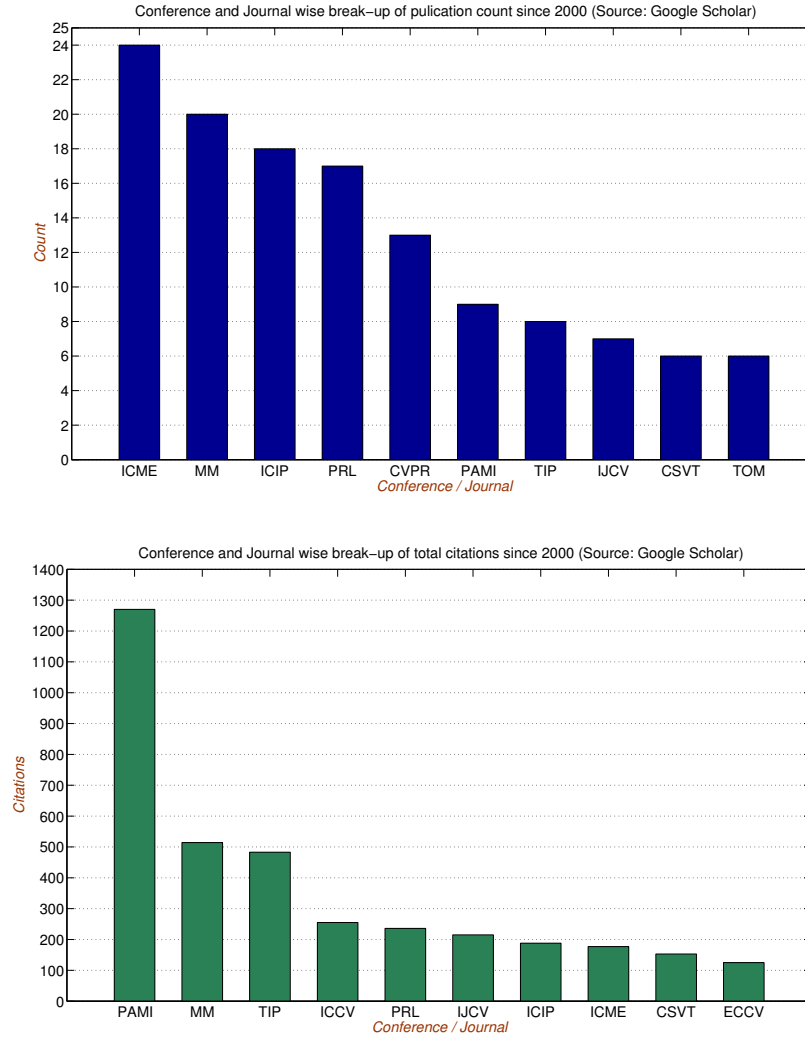


Figure A.1. Conference-wise and journal-wise publication statistics on topics closely related to image retrieval, year 2000 onwards. *Top:* Publication counts. *Bottom:* Total citations.

We query on the phrase “image OR images OR picture OR pictures OR content-based OR indexing OR ‘relevance feedback’ OR annotation ”, year 2000 onwards, for publications in the journals - IEEE T. Pattern Analysis and Machine Intelligence (PAMI), IEEE T. Image Processing (TIP), IEEE T. Circuits and Systems for Video Technology (CSVT), IEEE T. Multimedia (TOM), J. Machine Learning Research (JMLR), International J. Computer Vision (IJCV), Pattern Recognition Letters (PRL), and ACM Computing Surveys (SURV) and conferences - IEEE Computer Vision and Pattern Recognition (CVPR), International Conference on

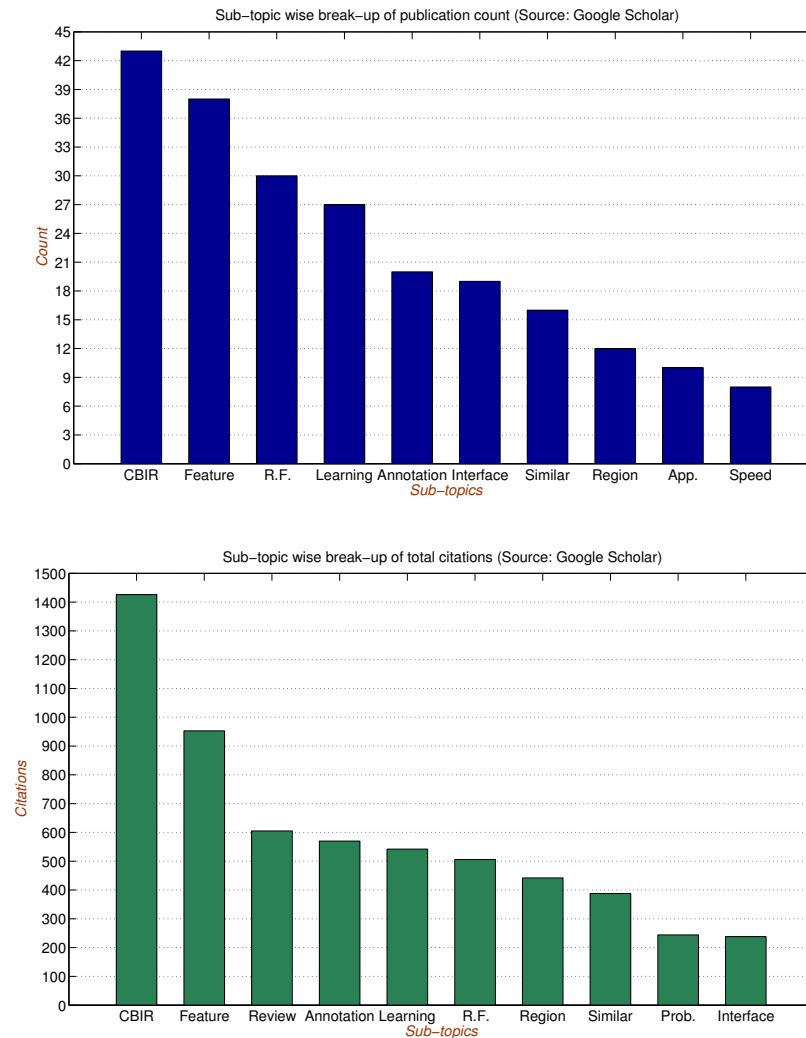


Figure A.2. Publication statistics on sub-topics of image retrieval, 2000 onwards. *Top:* Publication Counts. *Bottom:* Total citations. *Abbreviations:* *Feature* - Feature Extraction, *R.F.* - Relevance Feedback, *Similar* - Image similarity measures, *Region* - Region based approaches, *App.* - Applications, *Prob.* - Probabilistic approaches, *Speed* - Speed and other performance enhancements.

Computer Vision (ICCV), European Conference on Computer Vision (ECCV), IEEE International Conference on Image Processing (ICIP), ACM Multimedia (MM), ACM SIG Information Retrieval (IR), and ACM Human Factors in Computing Systems (CHI). Relevant papers among the top 100 results in each of these searches are used for the study. Google Scholar presents results roughly in decreasing order of citations (again, only rough approximations to the actual numbers).

Limiting search to the top few papers translates to reporting statistics on work with noticeable impact. We gathered statistics on two parameters, (1) publishing venue/journal, and (2) sub-topics of interest. These trends are reported in terms of (a) number of papers, and (b) total number of citations. Plots of these scores are presented in Fig. A.1 and Fig. A.2. Note that the tabulation is not mutually exclusive (i.e. one paper can have contributions in multiple sub-topics such as ‘Learning’ and ‘Region’, and hence are counted under both headings), neither is it exhaustive or scientifically precise (Google’s citation values may not be accurate). Nevertheless, these plots convey general trends in the relative impact of scholarly work. Readers are advised to use discretion when interpreting these results.

For the second experiment, we query Google Scholar for the phrase ‘image retrieval’ for each year from 1995 to 2005, and note the publication count, say x . We then add a phrase corresponding to a CBIR-related technique, e.g., relevance feedback, and note the publication count again, say y . For each year and for each phrase, we take the ratio y/x representing the fraction of relevant publications. The time-series plot for eight such phrases, over the eleven years, can be seen in Fig. A.3.

A.2 Scientific Impact on Other Communities

The list of references in our comprehensive survey [58] is probably a good way to understand how diverse image retrieval as a field is. There are at least 30 different well-known journals or proceedings where CBIR-related publications can be found, spanning at least eight different fields. In order to quantify this impact, we conduct a study. All the CBIR-related papers in our survey are analyzed in the following manner. Let a set of CBIR-related fields be denoted as $\mathbf{F} = \{\textit{Multimedia (MM)}, \textit{Information Retrieval (IR)}, \textit{Digital Libraries/ World Wide Web (DL)}, \textit{Human-Computer Interaction (HCI)}, \textit{Language Processing (LN)}, \textit{Artificial Intelligence (including ML) (AI)}, \textit{Computer Vision (CV)}\}$. Note the overlap among these fields, even though we treat them as distinct and non-overlapping for the sake of analysis. For each paper, we note what the core contribution is, including any new technique being introduced. For each such contribution, the core field it is associated with, $a \in \mathbf{F}$, is noted. For example, a paper that proposed

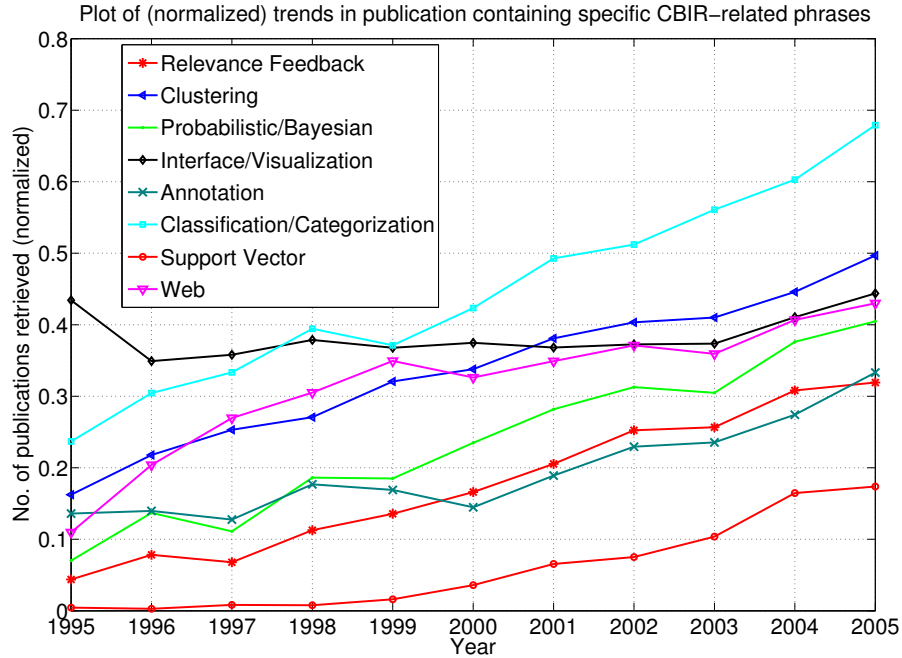


Figure A.3. Normalized trends in publications containing ‘image retrieval’ and corresponding phrases, as indexed by Google Scholar. Counts are normalized by the number of papers having ‘image retrieval’ for the particular year.

a spectral clustering based technique for computing image similarity is counted under both *CV* and *AI*. Now, given the journal/venue where the paper was published, we note the field $b \in \mathbf{F}$ which it caters to, e.g., ACM SIGIR is counted under *IR* and ACM MIR Workshop is counted under both *IR* and *MM*. Over the 170 papers, we count the publication count and the Google Scholar citations for each $a \rightarrow b$ pair, $a \neq b$. The 7×7 matrices so formed ($|\mathbf{F}| = 7$) for count and citations are represented as directed graphs, as shown in Fig. A.4. The thickness represents the publication or citation count, normalized by the maximum in the respective tables. Edges less than 5% of the maximum are not shown.

The basic idea behind constructing such graphs is to analyze how CBIR induces interests of one field of researchers in another field. A few trends are quite clear from the graphs. Most of the *MM*, *CV* and *AI* related work (i.e. CBIR research whose content falls into these categories) has been published in *IR* venues and received high citations. At the same time, *AI* related work published in *CV* venues has generated considerable impact. We view this as a side-effect of CBIR research

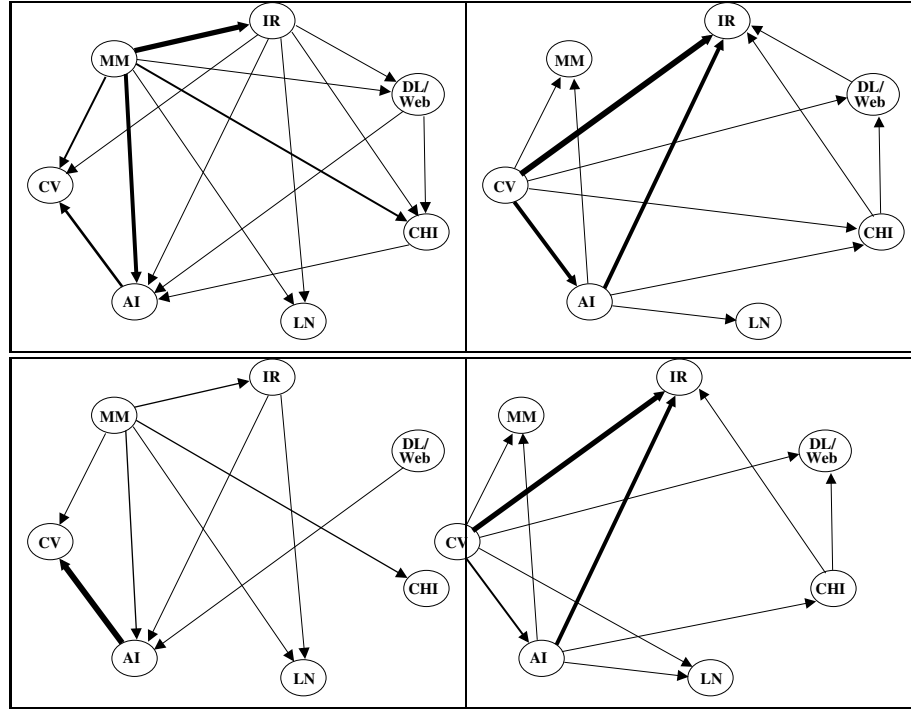


Figure A.4. [Acronyms: MM := Multimedia, IR := Information Retrieval, DL := Digital Libraries/ World Wide Web, HCI := Human-Computer Interaction, LN := Language Processing, AI := Artificial Intelligence, and CV := Computer Vision]. Directed graphs representing inter-field impact induced by CBIR-related publications. An edge $a \rightarrow b$ implies publications at venue/journal concerning field b , having content concerning field a . We show oppositely directed edges between pairs of nodes, wherever significant, in the left and right graphs. *Top:* Edge thicknesses represent (relative) **publication count**. *Bottom:* Edge thicknesses represent (relative) **citations** as reported by Google Scholar.

resulting in marriage of fields, communities, and ideas. But then again, there is little evidence of any mutual influence or benefits between the CV and CHI communities brought about by CBIR research.

Orthogonal Partition Generation

Here we show how the composite images are generated and how the approach leads to uniformly distributed placement of image centers. The significance of uniform distribution is that even if the adversary was aware of the algorithm for generating the composite images, she would not be able to improve chance of successful attack to better than random. In other words, if the algorithm generated constituent image positions non-uniformly, the adversary may predict image centers more often around regions of higher density, thereby increasing success chance even without any image processing.

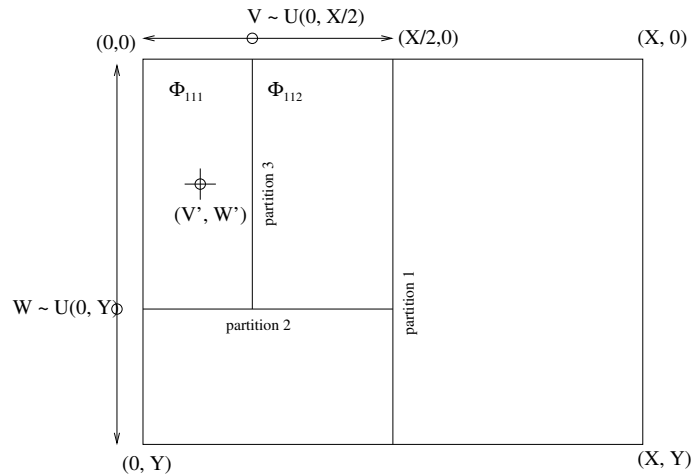


Figure B.1. Steps to orthogonal partition generation, to create 8 rectangular sub-regions for image tiling.

Let the composite image have dimensions $X \times Y$, and be denoted by Φ . Let

the *uniform distribution* over an $[a, b]$ range be denoted by $U(a, b)$. To achieve uniformity in partitioning Φ to generate 8 sub-images, the following algorithm is employed.

- Partition Φ along the center, either horizontally or vertically (randomly chosen), at $X/2$ or $Y/2$ respectively, to get Φ_1 and Φ_2 respectively.
- Recursively partition Φ_1 and Φ_2 further. Here we explain the case of horizontally split left-side rectangle Φ_1 , as shown in Fig. B.1. Other cases are similar.
 - Sample $V \sim U(0, Y)$ and partition Φ_1 vertically along V , to generate two more sub-rectangles Φ_{11} and Φ_{12} .
 - Sample $W \sim U(0, X/2)$ and partition the upper sub-rectangle Φ_{11} horizontally along W , to further generate sub-rectangles Φ_{111} and Φ_{112} .
- In a similar way for the remaining cases, we end up with 8 partitions Φ_{111} , Φ_{112} , Φ_{121} , Φ_{122} , Φ_{211} , Φ_{212} , Φ_{221} , and Φ_{222} .

Let us now analyze the probability distribution of the center of sub-rectangle Φ_{111} , with analysis of the other sub-rectangles being similar. The top-left corner of Φ_{111} is at $(0, 0)$. The bottom edge row is at W which is drawn uniformly at random over $[0, Y]$. Similarly, the right edge column is at V , which is drawn uniformly at random over $[0, X/2]$. Furthermore, V and W are drawn conditionally independent of each other. Therefore, the joint p.d.f. $f(v, w)$ of the random vector (V, W) is given by

$$f(v, w) = f_V(v)f_W(w) = \frac{1}{Y} \frac{2}{X} = \frac{2}{XY}$$

where $f_V(v)$ and $f_W(w)$ are the marginal densities. Furthermore, if a variable Z is drawn uniformly from $[0, c]$, having p.d.f. $\frac{1}{c}$, a new variable $T = Z/2$ is distributed uniformly over $[0, c/2]$ and has p.d.f. $\frac{2}{c}$. Therefore, given that the rectangle Φ_{111} spans $(0, 0)$ to (V, W) , its center (V', W') is located at $(V/2, W/2)$, is uniformly distributed, and has a joint p.d.f. $f(V', W')$ given by

$$f(v', w') = f_{V'}(v')f_{W'}(w') = \frac{1}{2}f_V(v)\frac{1}{2}f_W(w) = \frac{8}{XY}$$

since V' and W' are conditionally independent of each other.

In other words, for a composite image Φ of size $X \times Y$, the center (V, W) of the sub-rectangle Φ_{111} (and similarly for other sub-rectangles), generated by the algorithm above, can lie anywhere in the $(0, 0) - (X/4, Y/2)$ region with equal probability. Therefore, without analyzing the image content, given a single shot at clicking near the center of Φ_{111} (or any other sub-rectangle), the adversary's success probability is approximately $\frac{8\pi r^2}{XY}$, where r is the tolerance radius.

Bibliography

- [1] P. Aigrain, H. Zhang, and D. Petkovic. Content-based representation and retrieval of visual media: A review of the state-of-the-art. *Multimedia Tools and Applications*, 3(3):179–202, 1996.
- [2] Airlines.net, 2006. <http://www.airliners.net>.
- [3] Alipr, 2006. <http://www.alipr.com>.
- [4] J. Amores, N. Sebe, and P. Radeva. Fast spatial pattern discovery integrating boosting with constellations of contextual descriptors. In *Proc. IEEE CVPR*, 2005.
- [5] J. Amores, N. Sebe, and P. Radeva. Boosting the distance estimation: Application to the k-nearest neighbor classifier. *Pattern Recognition Letters*, 27(3):201–209, 2006.
- [6] J. Amores, N. Sebe, P. Radeva, T. Gevers, and A. Smeulders. Boosting contextual information in content-based image retrieval. In *Proc. MIR Workshop, ACM Multimedia*, 2004.
- [7] ARTStor.org, 2006. <http://www.artstor.org>.
- [8] A. Bar-hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *J. Machine Learning Research*, 6:937–965, 2005.
- [9] K. Barnard, P. Duygulu, D. Forsyth, N. deFreitas, D.M. Blei, and M.I. Jordan. Matching words and pictures. *J. Machine Learning Research*, 3:1107–1135, 2003.
- [10] I. Bartolini, P. Ciaccia, and M. Patella. Warp: Accurate retrieval of shapes using phase of fourier descriptors and time warping distance. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(1):142–147, 2005.

- [11] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.
- [12] S. Berretti and A. Del Bimbo. Modeling spatial relationships between 3d objects. In *Proc. IEEE ICPR*, 2006.
- [13] S. Berretti, A. Del Bimbo, and P. Pala. Retrieval by shape similarity with perceptual distance and effective indexing. *IEEE Trans. Multimedia*, 2(4):225–239, 2000.
- [14] S. Berretti, A. Del Bimbo, and E. Vicario. Weighted walkthroughs between extended entities for retrieval by spatial arrangement. *IEEE Trans. Multimedia*, 5(1):52–70, 2003.
- [15] D. M. Blei and M. I. Jordan. Modeling annotated data. In *Proc. ACM SIGIR*, 2003.
- [16] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *J. Machine Learning Research*, 3:993–1022, 2003.
- [17] A.L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.
- [18] A.L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.
- [19] C. Bohm, S. Berchtold, and D. A. Keim. Searching in high-dimensional space index structures for improving the performance of multimedia databases. *ACM Computing Surveys*, 33(3):322–373, 2001.
- [20] G. Bouchard and B. Triggs. Hierarchical part-based visual object categorization. In *Proc. IEEE CVPR*, 2005.
- [21] C.A. Bouman. Cluster: An unsupervised algorithm for modeling gaussian mixtures, 2006. <http://www.ece.purdue.edu/~bouman/>.
- [22] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, 1983.
- [23] C. c. Chang and C. j. Lin. Libsvm : A library for SVM, 2006. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>.
- [24] D. Cai, X. He, Z. Li, W. Y. Ma, and J. R. Wen. Hierarchical clustering of www image search results using visual, textual and link information. In *Proc. ACM Multimedia*, 2004.

- [25] Captcha.net, 2006. <http://www.captcha.net>.
- [26] J. Carballido-Gamio, S. Belongie, and S. Majumdar. Normalized cuts in 3-d for spinal MRI segmentation. *IEEE Trans. Medical Imaging*, 23(1):36–44, 2004.
- [27] G. Carneiro, A.B. Chan, P.J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29(3):394–410, 2007.
- [28] G. Carneiro and D. Lowe. Sparse flexible models of local features. In *Proc. ECCV*, 2006.
- [29] G. Carneiro and N. Vasconcelos. Minimum bayes error features for visual recognition by sequential feature selection and extraction. In *Proc. Canadian Conf. Computer and Robot Vision*, 2005.
- [30] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Color and texture-based image segmentation using em and its application to image querying and classification. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.
- [31] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.
- [32] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.
- [33] G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. In *Proc. NIPS*, 2001.
- [34] E. Y. Chang, K. Goh, G. Sychay, and G. Wu. CBSA: Content-based soft annotation for multimodal image retrieval using bayes point machines. *IEEE Trans. Circuits and Systems for Video Technology*, 13(1):26–38, 2003.
- [35] S.-F. Chang, J.R. Smith, M. Beigi, and A. Benitez. Visual information retrieval from large distributed online repositories. *Communications of the ACM*, 40(12):63–71, 1997.
- [36] S.K. Chang, Q.Y. Shi, and C.W. Yan. Iconic indexing by 2-d strings. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(3):413–427, 1987.
- [37] S.K. Chang, C.W. Yan, D.C. Dimitroff, and T. Arndt. An intelligent image database system. *IEEE Trans. Software Engineering*, 14(5):681–688, 1988.

- [38] K. Chellapilla and P.Y. Simard. Using machine learning to break visual human interaction proofs (HIPs). In *Proc. NIPS*, 2004.
- [39] C.-C. Chen, H. Wactlar, J. Z. Wang, and K. Kiernan. Digital imagery for significant cultural and historical materials - an emerging research field bridging people, culture, and technologies. *Intl. J. on Digital Libraries*, 5(4):275–286, 2005.
- [40] J. Chen, T.N. Pappas, A. Mojsilovic, and B. Rogowitz. Adaptive image segmentation based on color and texture. In *Proc. IEEE ICIP*, 2002.
- [41] Y. Chen and J. Z. Wang. A region-based fuzzy feature matching approach to content-based image retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(9):1252–1267, 2002.
- [42] Y. Chen and J. Z. Wang. Image categorization by learning and reasoning with regions. *J. Machine Learning Research*, 5:913–939, 2004.
- [43] Y. Chen, J. Z. Wang, and R. Krovetz. CLUE: Cluster-based retrieval of images by unsupervised learning. *IEEE Trans. Image Processing*, 14(8):1187–1201, 2005.
- [44] Y. Chen, X. Zhou, and T. S. Huang. One-class SVM for learning in image retrieval. In *Proc. IEEE ICIP*, 2002.
- [45] M. Chew and J. D. Tygar. Image recognition CAPTCHAs. In *Proc. Information Security Conf.*, 2004.
- [46] M. Chew and J.D. Tygar. Image recognition CAPTCHAs. In *Proc. ISC*, 2004.
- [47] P. Ciaccia, M. Patella, and P. Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *Proc. VLDB*, 1997.
- [48] CNN. Computer decodes mona lisa’s smile. *CNN - Technology*, 12/16/2005, 2005.
- [49] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.
- [50] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos. The bayesian image retrieval system, pichunter: Theory, implementation, and psychophysical experiments. *IEEE Trans. Image Processing*, 9(1):20–37, 2000.

- [51] I.J. Cox, J. Kilian, F.T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Trans. Image Processing*, 6(12):1673–1687, 1997.
- [52] A. Csillaghy, H. Hinterberger, and A.O. Benz. Content based image retrieval in astronomy. *Information Retrieval*, 3(3):229–241, 2000.
- [53] I. Dagan, L. Lee, and F.C.N. Pereira. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 34(1-3):43–69, 1999.
- [54] C. Dagli and T. S. Huang. A framework for grid-based image retrieval. In *Proc. IEEE ICPR*, 2004.
- [55] R. Datta, W. Ge, J. Li, and J.Z. Wang. Toward bridging the annotation-retrieval gap in image search. *IEEE MultiMedia*, 14(3):24–35, 2007.
- [56] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *Proc. ECCV*, 2006.
- [57] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Tagging over time: Real-world image annotation by lightweight meta-learning. In *Proc. ACM Multimedia*, 2007.
- [58] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2):1–60, 2008.
- [59] R. Datta, J. Li, and J. Z. Wang. Content-based image retrieval - a survey on the approaches and trends of the new age. In *Proc. MIR Workshop, ACM Multimedia*, 2005.
- [60] R. Datta, J. Li, and J. Z. Wang. IMAGINATION: A robust image-based captcha generation system. In *Proc. ACM Multimedia*, 2005.
- [61] R. Datta, J. Li, and J.Z. Wang. Imagination: A robust image-based captcha generation system. In *Proc. ACM Multimedia*, 2005.
- [62] R. Datta, J. Li, and J.Z. Wang. Learning the consensus on visual quality for next-generation image management. In *Proc. ACM Multimedia*, 2007.
- [63] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, Philadelphia, 1992.
- [64] V. de Silva and J. Tenenbaum. Global versus local methods in nonlinear dimensionality reduction. In *Proc. NIPS*, 2003.
- [65] A. Del Bimbo and P. Pala. Visual image retrieval by elastic matching of user sketches. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(2):121–132, 1997.

- [66] Y. Deng and B. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.
- [67] Y. Deng, B. S. Manjunath, C. Kenney, M. S. Moore, and H. Shin. An efficient color representation for image retrieval. *IEEE Trans. Image Processing*, 10(1):140–147, 2001.
- [68] Discovery. Digital pics ‘read’ by computer. *Discovery News*, 11/09/2006, 2006.
- [69] M. N. Do and M. Vetterli. Wavelet-based texture retrieval using generalized gaussian density and kullback-leibler distance. *IEEE Trans. Image Processing*, 11(2):146–158, 2002.
- [70] A. Dong and B. Bhanu. Active concept learning for image retrieval in dynamic databases. In *Proc. IEEE ICCV*, 2003.
- [71] DPChallenge, 2006. <http://www.dpchallenge.com>.
- [72] Y. Du and J. Z. Wang. A scalable integrated region-based image retrieval system. In *Proc. IEEE ICIP*, 2001.
- [73] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins. Visualizing tags over time. *ACM Transactions on the Web*, 1(2), 2007.
- [74] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proc. ECCV*, 2002.
- [75] Y. Eishenthal, G. Dror, and E. Ruppim. Facial attractiveness: Beauty and the machine. *Neural Computation*, 18(1):119–142, 2006.
- [76] J. Elson, J.R. Douceur, J. Howell, and J. Saul. Asirra: A CAPTCHA that exploits interest-aligned manual image categorization. In *Proc. ACM CCS*, 2007.
- [77] A.M. Eskicioglu and P.S. Fisher. Image quality measures and their performance. *IEEE Trans. Communications*, 45(12):2959–2965, 1995.
- [78] R. Fagin. Combining fuzzy information from multiple systems. In *Proc. PODS*, 1997.
- [79] Y. Fang and D. Geman. Experiments in mental face retrieval. In *Proc. Audio and Video-based Biometric Person Authentication*, 2005.

- [80] Y. Fang, D. Geman, and N. Boujemaa. An interactive system for mental face retrieval. In *Proc. MIR Workshop, ACM Multimedia*, 2005.
- [81] H. Feng, R. Shi, and T. S. Chua. A bootstrapping framework for annotating and retrieving www images. In *Proc. ACM Multimedia*, 2004.
- [82] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proc. IEEE CVPR*, 2004.
- [83] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. IEEE CVPR*, 2003.
- [84] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *Proc. IEEE CVPR*, 2005.
- [85] G.D. Finlayson. Color in perspective. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(10):1034–1038, 1996.
- [86] F. Fleuret and D. Geman. Stationary features and cat detection. *J. Machine Learning Research*, 9:2549–2578, 2008.
- [87] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, 1995.
- [88] Flickr, 2006. <http://www.flickr.com>.
- [89] R.W. Floyd and L. Steinberg. An adaptive algorithm for spatial grey scale. In *Proc. Society of Information Display*, 1976.
- [90] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proc. ICML*, 1996.
- [91] B. Gao, T.-Y. Liu, T. Qin, X. Zheng, Q.-S. Cheng, and W.-Y. Ma. Web image clustering by consistent utilization of visual features and surrounding texts. In *Proc. ACM Multimedia*, 2005.
- [92] Y. Gao, J. Fan, H. Luo, X. Xue, and R. Jain. Automatic image annotation by incorporating feature hierarchy and boosting to scale up SVM classifiers. In *Proc. ACM Multimedia*, 2006.
- [93] N. Garg and I. Weber. Personalized tag suggestion for flickr. In *Proc. WWW*, 2008.
- [94] T. Gevers and A.W.M. Smeulders. Pictoseek: Combining color and shape invariant features for image retrieval. *IEEE Trans. Image Processing*, 9(1):102–119, 2000.

- [95] GlobalMemoryNet, 2006. <http://www.memorynet.org>.
- [96] K.-S. Goh, E. Y. Chang, and K.-T. Cheng. SVM binary classifier ensembles for image classification. In *Proc. ACM CIKM*, 2001.
- [97] K.-S. Goh, E. Y. Chang, and W.-C. Lai. Multimodal concept-dependent active learning for image retrieval. In *Proc. ACM Multimedia*, 2004.
- [98] P. Golle. Machine learning attacks against the asirra captcha. In *Proc. ACM CCS*, 2008.
- [99] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, 1983.
- [100] Google Scholar, 2006. <http://scholar.google.com>.
- [101] S. Gordon, H. Greenspan, and J. Goldberger. Applying the information bottleneck principle to unsupervised clustering of discrete and continuous image representations. In *Proc. IEEE ICCV*, 2003.
- [102] V. Gouet and N. Boujemaa. On the robustness of color points of interest for image retrieval. In *Proc. IEEE ICIP*, 2002.
- [103] K. Grauman and T. Darrell. Efficient image matching with distributions of local invariant features. In *Proc. IEEE CVPR*, 2005.
- [104] Guardian. How captcha was foiled: Are you a man or a mouse?, 2008. <http://www.guardian.co.uk/technology/2008/aug/28/internet.captcha>.
- [105] A. Gupta and R. Jain. Visual information retrieval. *Communications of the ACM*, 40(5):70–79, 1997.
- [106] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Machine Learning Research*, 3:1157–1182, 2003.
- [107] E. Hadjidemetriou, M. D. Grossberg, and S. K. Nayar. Multiresolution histograms and their use for recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(7):831–847, 2004.
- [108] J. Han, K. N. Ngan, M. Li, and H.-J. Zhang. A memory learning framework for effective image retrieval. *IEEE Trans. Image Processing*, 14(4):511–524, 2005.
- [109] R.M. Haralick. Statistical and structural approaches to texture. *Proc. IEEE*, 67(5):786–804, 1979.
- [110] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2001.

- [111] A. G. Hauptmann and M. G. Christel. Successful approaches in the TREC video retrieval evaluations. In *Proc. ACM Multimedia*, 2004.
- [112] J. He, M. Li, H.-J. Zhang, H. Tong, and C. Zhang. Manifold-ranking based image retrieval. In *Proc. ACM Multimedia*, 2004.
- [113] J. He, M. Li, H.-J. Zhang, H. Tong, and C. Zhang. Mean version space: a new active learning method for content-based image retrieval. In *Proc. MIR Workshop, ACM Multimedia*, 2004.
- [114] X. He. Incremental semi-supervised subspace learning for image retrieval. In *Proc. ACM Multimedia*, 2004.
- [115] X. He, W.-Y. Ma, and H.-J. Zhang. Learning an image manifold for retrieval. In *Proc. ACM Multimedia*, 2004.
- [116] T.K. Ho, J.J. Hull, and S.N. Srihari. Decision combination in multiple classifier systems. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(1):66–75, 1994.
- [117] C.-H. Hoi and M. R. Lyu. Group-based relevance feedback with support vector machine ensembles. In *Proc. IEEE ICPR*, 2004.
- [118] C.-H. Hoi and M. R. Lyu. A novel log-based relevance feedback technique in content-based image retrieval. In *Proc. ACM Multimedia*, 2004.
- [119] D. Hoiem, R. Sukthankar, H. Schneiderman, and L. Huston. Object-based image retrieval using the statistical structure of images. In *Proc. IEEE CVPR*, 2004.
- [120] Jing Huang, S. Ravi Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. Spatial color indexing and applications. *Intl. J. Computer Vision*, 35(3):245–268, 1999.
- [121] D. P. Huijsmans and N. Sebe. How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(2):245–251, 2005.
- [122] Q. Iqbal and J. K. Aggarwal. Retrieval by classification of images containing large manmade objects using perceptual grouping. *Pattern Recognition J.*, 35(7):1463–1479, 2002.
- [123] A. Jaimes, K. Omura, T. Nagamine, and K. Hirata. Memory cues for meeting video retrieval. In *Proc. CARPE Workshop, ACM Multimedia*, 2004.
- [124] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

- [125] A.K. Jain and F. Farrokhnia. Unsupervised texture segmentation using gabor filters. In *Proc. Intl. Conf. Systems, Man and Cybernetics*, 1990.
- [126] B. J. Jansen, A. Spink, and J. Pedersen. An analysis of multimedia searching on Altavista. In *Proc. MIR Workshop, ACM Multimedia*, 2003.
- [127] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proc. ACM SIGIR*, 2003.
- [128] S. Jeong, C. S. Won, and R.M. Gray. Image retrieval using color histograms generated by gauss mixture vector quantization. *Computer Vision and Image Understanding*, 9(1-3):44-66, 2004.
- [129] R. Jin, J. Y. Chai, and L. Si. Effective automatic image annotation via a coherent language model and active learning. In *Proc. ACM Multimedia*, 2004.
- [130] R. Jin and A.G. Hauptmann. Using a probabilistic source model for comparing images. In *Proc. IEEE ICIP*, 2002.
- [131] Y. Jin, L. Khan, L. Wang, and M. Awad. Image annotations by combining multiple evidence & wordnet. In *Proc. ACM Multimedia*, 2005.
- [132] F. Jing, M. Li, H.-J. Zhang, and B. Zhang. An efficient and effective region-based image retrieval framework. *IEEE Trans. Image Processing*, 13(5):699-709, 2004.
- [133] F. Jing, M. Li, H.-J. Zhang, and B. Zhang. Relevance feedback in region-based image retrieval. *IEEE Trans. Circuits and Systems for Video Technology*, 14(5):672-681, 2004.
- [134] F. Jing, M. Li, H. J. Zhang, and B. Zhang. A unified framework for image retrieval using keyword and visual features. *IEEE Trans. Image Processing*, 14(6), 2005.
- [135] F. Jing, C. Wang, Y. Yao, K. Deng, L. Zhang, and W. Y. Ma. IGroup: Web image search results clustering. In *Proc. ACM Multimedia*, 2006.
- [136] D. Joshi, R. Datta, Z. Zhuang, W.P. Weiss, M. Friedenberg, J.Z. Wang, and J. Li. Paragrab: A comprehensive architecture for web image management and multimodal querying, 2006.
- [137] D. Joshi, M. Naphade, and A. Natsev. A greedy performance driven algorithm for decision fusion learning. In *IEEE ICIP*, 2007.
- [138] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate detection and subimage retrieval. In *Proc. ACM Multimedia*, 2004.

- [139] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate detection and subimage retrieval. In *Proc. ACM Multimedia*, 2004.
- [140] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *Proc. IEEE CVPR*, 2006.
- [141] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *Proc. IEEE CVPR*, 2006.
- [142] M. L. Kherfi, D. Ziou, and A. Bernardi. Image retrieval from the world wide web: Issues, techniques, and systems. *ACM Computing Surveys*, 36(1):35–67, 2004.
- [143] D.-H. Kim and C.-W. Chung. Qcluster: Relevance feedback using adaptive clustering for content based image retrieval. In *Proc. ACM SIGMOD*, 2003.
- [144] Y. S. Kim, W. N. Street, and F. Menczer. Feature selection in unsupervised learning via evolutionary search. In *Proc. ACM SIGKDD*, 2000.
- [145] B. Ko and H. Byun. Integrated region-based image retrieval using region’s spatial relationships. In *Proc. IEEE ICPR*, 2002.
- [146] J.Z. Kolter and M.A. Maloof. Dynamic weighted majority: An ensemble method for drifting concepts. *JMLR*, 8:2755–2790, 2007.
- [147] R.E. Korf. Optimal rectangle packing: New results. In *Proc. ICAPS*, 2004.
- [148] J. Laaksonen, M. Koskela, S. Laakso, and E. Oja. Self-organizing maps as a relevance feedback technique in content-based image retrieval. *Pattern Analysis and Applications*, 4:140–152, 2001.
- [149] J. Laaksonen, M. Koskela, and E. Oja. PicSOM: Self-organizing image retrieval with MPEG-7 content descriptors. *IEEE Trans. Neural Networks*, 13(4):841–853, 2002.
- [150] L. J. Latecki and R. Lakamper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10):1185–1190, 2000.
- [151] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *Proc. NIPS*, 2003.
- [152] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *Proc. IEEE ICCV*, 2003.
- [153] C. Leacock and M. Chodorow. Combining local context and wordnet similarity for word sense identification. *C. Fellbaum, Ed., WordNet: An Electronic Lexical Database*, pages 265–283, 1998.

- [154] D.B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Comm. of the ACM*, 38(11):33–38, 1995.
- [155] M. Lesk. How much information is there in the world?, 1997. <http://www.lesk.com/mlesk/ksg97/ksg.html>.
- [156] E. Levina and P. Bickel. The earth mover’s distance is the mallows distance: Some insights from statistics. In *Proc. IEEE ICCV*, 2001.
- [157] M. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-based multimedia information retrieval: State-of-the-art and challenges. *ACM Trans. Multimedia Computing, Communication, and Applications*, 2(1):1–19, 2006.
- [158] B. Li, K.-S. Goh, and E. Y. Chang. Confidence-based dynamic ensemble for image annotation and semantics discovery. In *Proc. ACM Multimedia*, 2003.
- [159] J. Li. A mutual semantic endorsement approach to image retrieval and context provision. In *Proc. MIR Workshop, ACM Multimedia*, 2005.
- [160] J. Li. Two-scale image retrieval with significant meta-information feedback. In *Proc. ACM Multimedia*, 2005.
- [161] J. Li, R. M. Gray, and R. A. Olshen. Multiresolution image classification by hierarchical modeling with two dimensional hidden markov models. *IEEE Trans. Information Theory*, 46(5):1826–1841, 2000.
- [162] J. Li, A. Najmi, and R. M. Gray. Image classification by a two dimensional hidden markov model. *IEEE Trans. Signal Processing*, 48(2):527–533, 2000.
- [163] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.
- [164] J. Li and J. Z. Wang. Studying digital imagery of ancient paintings by mixtures of stochastic models. *IEEE Trans. Image Processing*, 13(3):340–353, 2004.
- [165] J. Li, J. Z. Wang, and G. Wiederhold. IRM: Integrated region matching for image retrieval. In *Proc. ACM Multimedia*, 2000.
- [166] J. Li and J.Z. Wang. Real-time computerized annotation of pictures. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(6):985–1002, 2008.
- [167] X. Li, L. Chen, L. Zhang, F. Lin, and W. Y. Ma. Image annotation by large-scale content-based image retrieval. In *Proc. ACM Multimedia*, 2006.

- [168] Y. Li, L.G. Shaprio, and J.A. Bilmes. Generative/discriminative learning algorithm for image classification. In *Proc. ICCV*, 2005.
- [169] Z.-W. Li, X. Xie, H. Liu, X. Tang, M. Li, and W.-Y. Ma. Intuitive and effective interfaces for www image search engines. In *Proc. ACM Multimedia*, 2004.
- [170] Y.-Yu Lin, T.-L. Liu, and H.-T. Chen. Semantic manifold learning for image retrieval. In *Proc. ACM Multimedia*, 2005.
- [171] W. Liu and X. Tang. Learning an image-word embedding for image auto-annotation on the nonlinear latent space. In *Proc. ACM Multimedia*, 2005.
- [172] D.G. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, 1999.
- [173] Y. Lu, C. Hu, X. Zhu, H.J. Zhang, and Q. Yang. A unified framework for semantics and feature based relevance feedback in image retrieval systems. In *Proc. ACM Multimedia*, 2000.
- [174] P. Lyman and H. Varian. How much information?, 2003. <http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/>.
- [175] W.-Y. Ma and B.S. Manjunath. Texture thesaurus for browsing large aerial photographs. *J. American Society for Information Science*, 49(7):633–648, 1998.
- [176] W.Y. Ma and B.S. Manjunath. Netra: A toolbox for navigating large image databases. In *Proc. IEEE ICIP*, 1997.
- [177] W.Y. Ma and B.S. Manjunath. Netra: A toolbox for navigating large image databases. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 7(3):184–198, 1999.
- [178] J. Malik, S. Belongie, T. K. Leung, and J. Shi. Contour and texture analysis for image segmentation. *Intl. J. Computer Vision*, 43(1):7–27, 2001.
- [179] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *J. Optical Society of America A*, 7(5):923–932, 1990.
- [180] C. L. Mallows. A note on asymptotic joint normality. *Annals of Mathematical Statistics*, 43(2):508–515, 1972.
- [181] C.L. Mallows. A note on asymptotic joint normality. *Annals of Mathematical Statistics*, 43(2):508–515, 1972.

- [182] M.A. Maloof and R.S. Michalski. Incremental learning with partial instance memory. *Artificial Intelligence*, 154:95–126, 2004.
- [183] B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Trans. Circuits and Systems for Video Technology*, 11(6):703–715, 2001.
- [184] B.S. Manjunath and W.-Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.
- [185] J. R. Mathiassen, A. Skavhaug, and K. Bo. Texture similarity measure using kullback-leibler divergence between gamma distributions. In *Proc. ECCV*, 2002.
- [186] G. McLachlan and D. Peel. *Finite Mixture Models*. Wiley-Interscience, 2000.
- [187] R. Mehrotra and J. E. Gary. Similar-shape retrieval in shape data management. *IEEE Computer*, 28(9):57–62, 1995.
- [188] K. Mikolajczk and C. Schmid. A performance evaluation of local descriptors. In *Proc. IEEE CVPR*, 2003.
- [189] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *Intl. J. Computer Vision*, 60(1):63–86, 2004.
- [190] G. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [191] P. Mitra, C.A. Murthy, and S.K. Pal. Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(3):301–312, 2002.
- [192] F. Mokhtarian. Silhouette-based isolated object recognition through curvature scale space. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(5):539–544, 1995.
- [193] F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *Proc. ACM Multimedia*, 2003.
- [194] W.G. Morein, A. Stavrou, D.L. Cook, V. Mishra Keromytis, and D. Rubenstein. Using graphic turing tests to counter automated ddos attacks against web servers. In *Proc. ACM CCS*, 2003.
- [195] G. Mori and J. Malik. Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA. In *Proc. IEEE CVPR*, 2003.

- [196] G. Moy, N. Jones, C. Harkless, and R. Potter. Distortion estimation techniques in solving visual CAPTCHAs. In *Proc. IEEE CVPR*, 2004.
- [197] S. Mukherjea, K. Hirata, and Y. Hara. Amore: A world wide web image retrieval engine. In *Proc. WWW*, 1999.
- [198] H. Muller, N. Michoux, D. Bandon, and A. Geissbuhler. A review of content-based image retrieval systems in medical applications - clinical benefits and future directions. *Intl. J. Medical Informatics*, 73(1):1–23, 2004.
- [199] H. Muller, T. Pun, and D. Squire. Learning from user behavior in image retrieval: Application of market basket analysis. *Intl. J. Computer Vision*, 56(1/2):65–77, 2004.
- [200] T. Nagamine, A. Jaimes, K. Omura, and K. Hirata. A visuospatial memory cue system for meeting video retrieval. In *Proc. ACM Multimedia (Demonstration)*, 2004.
- [201] M. Nakazato, C. Dagli, and T.S. Huang. Evaluating group-based relevance feedback for content-based image retrieval. In *Proc. IEEE ICIP*, 2003.
- [202] A. Natsev, M. R. Naphade, and J. Tesic. Learning the semantics of multimedia queries and concepts from a small number of examples. In *Proc. ACM Multimedia*, 2005.
- [203] A. Natsev, R. Rastogi, and K. Shim. Walrus: A similarity retrieval algorithm for image databases. *IEEE Trans. Knowledge and Data Engineering*, 16(3):301–316, 2004.
- [204] A. Natsev and J.R. Smith. A study of image retrieval by anchoring. In *Proc. IEEE ICME*, 2002.
- [205] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui. Physics-motivated features for distinguishing photographic images and computer graphics. In *Proc. ACM Multimedia*, 2005.
- [206] Hot or Not, 2006. <http://www.hotornot.com>.
- [207] T. H. Painter, J. Dozier, D. A. Roberts, R. E. Davis, and R. O. Green. Retrieval of subpixel snow-covered area and grain size from imaging spectrometer data. *Remote Sensing of Environment*, 85(1):64–77, 2003.
- [208] N. Panda and E. Y. Chang. Efficient top-k hyperplane query processing for multimedia information retrieval. In *Proc. ACM Multimedia*, 2006.
- [209] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. In *Proc. SPIE*, 1994.

- [210] G. Petraglia, M. Sebillo, M. Tucci, and G. Tortora. Virtual images for similarity retrieval in image databases. *IEEE Trans. Knowledge and Data Engineering*, 13(6):951–967, 2001.
- [211] E. G. M. Petrakis, A. Diplaros, and E. Milios. Matching and retrieval of distorted and occluded shapes using dynamic programming. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.
- [212] E.G.M. Petrakis and A. Faloutsos. Similarity searching in medical image databases. *IEEE Trans. Knowledge and Data Engineering*, 9(3):435–447, 1997.
- [213] Photo.net, 2006. <http://photo.net>.
- [214] Photo.net. Rating system, 2006. <http://www.photo.net/gallery/photocritique/standards/>.
- [215] M. Pi, M. K. Mandal, and A. Basu. Image retrieval based on histogram of fractal parameters. *IEEE Trans. Multimedia*, 7(4):597–605, 2005.
- [216] J. Portilla and E.P. Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *Intl. J. Computer Vision*, 40(1):49–71, 2000.
- [217] Oxford University Press. Oxford advanced learner’s dictionary, 2006.
- [218] T. Quack, U. Monich, L. Thiele, and B. S. Manjunath. Cortina: A system for largescale, content-based web image retrieval. In *Proc. ACM Multimedia*, 2004.
- [219] P. Resnick and H.R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [220] N. C. Rowe. Marie-4: A high-recall, self-improving web crawler that finds images using captions. *IEEE Intelligent Systems*, 17(4):8–14, 2002.
- [221] Y. Rubner, C. Tomasi, and L.J. Guibas. The earth mover’s distance as a metric for image retrieval. *Intl. J. Computer Vision*, 40(2):99–121, 2000.
- [222] Y. Rui and T. S. Huang. Optimizing learning in image retrieval. In *Proc. IEEE CVPR*, 2000.
- [223] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool in interactive content-based image retrieval. *IEEE Trans. Circuits and Systems for Video Technology*, 8(5):644–655, 1998.

- [224] Y. Rui, T.S. Huang, and S.-F. Chang. Image retrieval: Current techniques, promising directions and open issues. *J. Visual Communication and Image Representation*, 10(1):39–62, 1999.
- [225] Y. Rui, T.S. Huang, and S. Mehrotra. Content-based image retrieval with relevance feedback in mars. In *Proc. IEEE ICIP*, 1997.
- [226] Y. Rui and Z. Liu. ARTiFACIAL: Automated reverse turing test using facial features. In *Proc. ACM Multimedia*, 2003.
- [227] J. Saul. Petfinder, 2009. <http://www.petfinder.com>.
- [228] B. Le Saux and N. Boujemaa. Unsupervised robust clustering for image database categorization. In *Proc. IEEE ICPR*, 2002.
- [229] F. Schaffalitzky and A. Zisserman. Viewpoint invariant texture matching and wide baseline stereo. In *Proc. IEEE ICCV*, 2001.
- [230] J.C. Schlimmer and R.H. Granger. Beyond incremental processing: Tracking concept drift. In *Proc. AAAI*, 1986.
- [231] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.
- [232] M. Schroder, H. Rehrauer, K. Seidel, and M. Datcu. Interactive learning and probabilistic retrieval in remote sensing image archives. *IEEE Trans. Geoscience and Remote Sensing*, 38(5):2288–2298, 2000.
- [233] ScientificAmerican. Computers get the picture. *Steve Mirsky - Scientific American 60-second World of Science*, 11/06/2006, 2006.
- [234] N. Sebe, M. S. Lew, and D. P. Huijsmans. Toward improved ranking metrics. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(10):1132–1141, 2000.
- [235] N. Sebe, M. S. Lew, X. Zhou, T. S. Huang, and E. Bakker. The state of the art in image and video retrieval. In *Proc. CIVR*, 2003.
- [236] H.R. Sheikh, A.C. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *IEEE Trans. Image Processing*, 14(11):1918–1927, 2005.
- [237] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- [238] B. Sigurbjornsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *Proc. WWW*, 2008.

- [239] D. Silver, G. Bakir, K. Bennett, R. Caruana, M. Pontil, S. Russell, and P. Tadepalli. Inductive transfer: 10 years later. In *Intl. Workshop at NIPS*, 2005.
- [240] Slashdot. Searching by image instead of keywords, 2005. <http://slashdot.org/articles/05/05/04/2239224.shtml>.
- [241] Slashdot. Yahoo CAPTCHA hacked, 2008. <http://it.slashdot.org/it/08/01/30/0037254.shtml>.
- [242] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, , and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.
- [243] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, , and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.
- [244] J.R. Smith and S.-F. Chang. Integrated spatial and feature image query. *IEEE Trans. Knowledge and Data Engineering*, 9(3):435–447, 1997.
- [245] J.R. Smith and S.-F. Chang. Visualeek: a fully automated content-based image query system. In *Proc. ACM Multimedia*, 1997.
- [246] B. Smolka, M. Szczepanski, R. Lukac, and A. N. Venetsanopoulos. Robust color image retrieval for the world wide web. In *Proc. IEEE ICASSP*, 2004.
- [247] C. G. M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, 2005.
- [248] Z. Su, H.-J. Zhang, S. Li, and S. Ma. Relevance feedback in content-based image retrieval: Bayesian framework, feature subspaces, and progressive learning. *IEEE Trans. Image Processing*, 12(8):924–937, 2003.
- [249] M.J. Swain and B.H. Ballard. Color indexing. *Intl. J. Computer Vision*, 7(1):11–32, 1991.
- [250] D.L. Swets and J. Weng. Using discriminant eigenfeatures for image retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(8):831–836, 1996.
- [251] Terragalleria, 2006. <http://terrageria.com>.
- [252] A. Thayananthan, B. Stenger, P.H.S. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *Proc. IEEE CVPR*, 2003.

- [253] C. Theoharatos, N. A. Laskaris, G. Economou, and S. Fotopoulos. A generic scheme for color image retrieval based on the multivariate wald-wolfowitz test. *IEEE Trans. Knowledge and Data Engineering*, 17(6):808–819, 2005.
- [254] T.M. Therneau and E.J. Atkinson. An introduction to recursive partitioning using rpart routines. In *Technical Report, Mayo Foundation*, 1997.
- [255] Q. Tian, N. Sebe, M. S. Lew, E. Loupias, and T. S. Huang. Image retrieval using wavelet-based salient points. *J. Electronic Imaging*, 10(4):835–849, 2001.
- [256] K. Tieu and P. Viola. Boosting image retrieval. *Intl. J. Computer Vision*, 56(1/2):17–36, 2004.
- [257] N. Tishby, F.C. Pereira, and W. Bialek. The information botfleneck method. In *Proc. Allerton Conf. Communication and Computation*, 1999.
- [258] H. Tong, M. Li, H. Zhang, J. He, and C. Zhang. Classification of digital photos taken by photographers or home users. In *Proc. Pacific Rim Conference on Multimedia*, 2004.
- [259] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *Proc. ACM Multimedia*, 2001.
- [260] Z. Tu and S.-C. Zhu. Image segmentation by data-driven markov chain monte carlo. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5):657–673, 2002.
- [261] A. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.
- [262] T. Tuytelaars and L. van Gool. Content-based image retrieval based on local affinity invariant regions. In *Proc. VISUAL*, 1999.
- [263] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Trans. Image Processing*, 4(11):1549–1560, 1995.
- [264] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H.-J. Zhang. Image classification for content-based indexing. *IEEE Trans. Image Processing*, 10(1):117–130, 2001.
- [265] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.
- [266] N. Vasconcelos. On the efficient evaluation of probabilistic similarity functions for image retrieval. *IEEE Trans. Information Theory*, 50(7):1482–1496, 2004.

- [267] N. Vasconcelos and A. Lippman. Learning from user feedback in image retrieval systems. In *Proc. NIPS*, 2000.
- [268] N. Vasconcelos and A. Lippman. A probabilistic architecture for content-based image retrieval. In *Proc. IEEE CVPR*, 2000.
- [269] N. Vasconcelos and A. Lippman. A multiresolution manifold distance for invariant image similarity. *IEEE Trans. Multimedia*, 7(1):127–142, 2005.
- [270] VBulletin. Nospam! an alternative to captcha images, 2008. <http://www.vbulletin.org/forum/showthread.php?t=124828>.
- [271] A. Velivelli, C.-W. Ngo, and T. S. Huang. Detection of documentary scene changes by audio-visual fusion. In *Proc. CIVR*, 2004.
- [272] R. Vilalta and Y. Drissi. A perspective view and survey of meta-learning. *AI Review*, 18(2):77–95, 2002.
- [273] T. Volkmer, J. R. Smith, and A. Natsev. A web-based system for collaborative annotation of large image and video collections: An evaluation and user study. In *Proc. ACM Multimedia*, 2005.
- [274] L. von Ahn, M. Blum, and J. Langford. Telling humans and computers apart (automatically) or how lazy cryptographers do ai. *Communications of the ACM*, 47(2):57–60, 2004.
- [275] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang. Image annotation refinement using random walk with restarts. In *Proc. ACM Multimedia*, 2006.
- [276] J. Z. Wang, N. Boujemaa, A. Del Bimbo, D. Geman, A. Hauptmann, and J. Tesic. Diversity in multimedia information retrieval research. In *Proc. MIR Workshop, ACM Multimedia*, 2006.
- [277] J. Z. Wang, J. Li, R. M. Gray, and G. Wiederhold. Unsupervised multiresolution segmentation for images with low depth of field. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(1):85–90, 2001.
- [278] J.Z. Wang, J. Li, and G. Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.
- [279] J.Z. Wang, G. Wiederhold, O. Firschein, and S.X. Wei. Content-based image indexing and searching using daubechies’ wavelets. *Intl. J. Digital Libraries*, 1(4):311–328, 1998.
- [280] X.-J. Wang, W.-Y. Ma, Q.-C. He, and X. Li. Grouping web image search result. In *Proc. ACM Multimedia*, 2004.

- [281] X. J. Wang, W. Y. Ma, G. R. Xue, and X. Li. Multi-model similarity propagation and its application for web image retrieval. In *Proc. ACM Multimedia*, 2004.
- [282] X.-J. Wang, L. Zhang, X. Li, and W.-Y. Ma. Annotating images by mining image search results. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(11):1919–1932, 2008.
- [283] Y. H. Wang. Image indexing and similarity retrieval based on spatial relationship model. *Information Sciences - Informatics and Computer Science*, 154(1-2):39–58, 2003.
- [284] Z. Wang, Z. Chi, and D. Feng. Fuzzy integral for leaf image retrieval. In *Proc. IEEE Intl. Conf. Fuzzy Systems*, 2002.
- [285] M. Webe, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proc. ECCV*, 2000.
- [286] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik. Feature selection for SVMs. In *Proc. NIPS*, 2000.
- [287] G. Widmer. Tracking context changes through meta-learning. *Machine Learning*, 27(3):259–286, 1997.
- [288] R.C. Wilson and E.R. Hancock. Structural matching by discrete relaxation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(6):634–648, 1997.
- [289] H.J. Wolfson and I. Rigoutsos. Geometric hashing: an overview. *IEEE Trans. Computational Science and Engineering*, 4(4):10–21, 1997.
- [290] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.
- [291] R.C.F. Wong and C.H.C. Leung. Automatic semantic annotation of real-world web images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(11):1933–1944, 2008.
- [292] G. Wu, E. Y. Chang, and N. Panda. Formulating context-dependent similarity functions. In *Proc. ACM Multimedia*, 2005.
- [293] H. Wu, H. Lu, and S. Ma. WillHunter: Interactive image retrieval with multilevel relevance measurement. In *Proc. IEEE ICPR*, 2004.
- [294] P. Wu and B. S. Manjunath. Adaptive nearest neighbor search for relevance feedback in large image databases. In *Proc. ACM Multimedia*, 2001.

- [295] W. Wu and J. Yang. SmartLabel: An object labeling tool using iterated harmonic energy minimization. In *Proc. ACM Multimedia*, 2006.
- [296] Y. Wu, E. Y. Chang, K. C. C. Chang, and J. R. Smith. Optimal multimodal fusion for multimedia data analysis. In *Proc. ACM Multimedia*, 2004.
- [297] Y. Wu, Q. Tian, and T. S. Huang. Discriminant-EM algorithm with application to image retrieval. In *Proc. IEEE CVPR*, 2000.
- [298] E.P. Xing, A.Y. Ng, M.I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *Proc. NIPS*, 2003.
- [299] C. Yang, M. Dong, and F. Fotouhi. Region based image annotation through multiple-instance learning. In *Proc. ACM Multimedia*, 2005.
- [300] C. Yang, M. Dong, and F. Fotouhi. Semantic feedback for interactive image retrieval. In *Proc. ACM Multimedia*, 2005.
- [301] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proc. ACM CHI*, 2003.
- [302] J. Yu, J. Amores, N. Sebe, and Q. Tian. Toward robust distance metric analysis for similarity estimation. In *Proc. IEEE CVPR*, 2006.
- [303] S. X. Yu and J. Shi. Segmentation given partial grouping constraints. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(2):173–183, 2004.
- [304] Y. Zhai, A. Yilmaz, and M. Shah. Story segmentation in news videos using visual and textual cues. In *Proc. ACM Multimedia*, 2005.
- [305] D.-Q. Zhang and S.-F. Chang. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *Proc. ACM Multimedia*, 2004.
- [306] H. Zhang, R. Rahmani, S. R. Cholleti, and S. A. Goldman. Local image representations using pruned salient points with applications to cbir. In *Proc. ACM Multimedia*, 2006.
- [307] H. J. Zhang, L. Wenying, and C. Hu. iFind - A system for semantics and feature based image retrieval over internet. In *Proc. ACM Multimedia*, 2000.
- [308] L. Zhang, L. Chen, F. Jing, K. Deng, and W. Y. Ma. EnjoyPhoto - A vertical image search engine for enjoying high-quality photos. In *Proc. ACM Multimedia*, 2006.
- [309] L. Zhang, L. Chen, M. Li, and H.-J. Zhang. Automated annotation of human faces in family albums. In *Proc. ACM Multimedia*, 2003.

- [310] Q. Zhang, S. A. Goldman, W. Yu, and J. E. Fritts. Content-based image retrieval using multiple-instance learning. In *Proc. ICML*, 2002.
- [311] R. Zhang and Z. Zhang. Hidden semantic concept discovery in region based image retrieval. In *Proc. IEEE CVPR*, 2004.
- [312] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE Trans. Medical Imaging*, 20(1):45–57, 2001.
- [313] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, 2003.
- [314] B. Zheng, D. C. McClean, and X. Lu. Identifying biological concepts from a protein-related corpus with a probabilistic topic model. *BMC Bioinformatics*, 7(58), 2006.
- [315] X. Zheng, D. Cai, X. He, W.-Y. Ma, and X. Lin. Locality preserving clustering for image database. In *Proc. ACM Multimedia*, 2004.
- [316] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Scholkopf. Ranking on data manifolds. In *Proc. NIPS*, 2003.
- [317] X. S. Zhou and T. S. Huang. Comparing discriminating transformations and SVM for learning during multimedia retrieval. In *Proc. ACM Multimedia*, 2001.
- [318] X. S. Zhou and T. S. Huang. Small sample learning during multimedia retrieval using biasmap. In *Proc. IEEE CVPR*, 2001.
- [319] X. S. Zhou and T. S. Huang. Unifying keywords and visual contents in image retrieval. *IEEE Multimedia*, 9(2):23–33, 2002.
- [320] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8:536–544, 2003.
- [321] L. Zhu, A. Zhang, A. Rao, and R. Srihari. Keyblock: An approach for content-based image retrieval. In *Proc. ACM Multimedia*, 2000.
- [322] S.-C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(9):884–900, 1996.

Vita

Ritendra Datta

Ritendra Datta completed his Bachelor of Engineering degree from the Bengal Engineering and Science University, Shibpur, India, in June 2004, and subsequently entered the Ph.D. program in Computer Science and Engineering at Penn State, in August 2004. For the academic year 2004-05, he was awarded the Glenn Singley Memorial Graduate Fellowship in Engineering by Penn State. While in the program, he worked as summer researcher at the IBM T.J. Watson Research Center, Xerox PARC, and Google, in 2006, 2007, and 2008 respectively. In 2007, IBM Research named him as one of eight Emerging Leaders in Multimedia. According to Google Scholar, as of February 2009, his first-authored papers have received over 240 citations. His research interests lie in statistical modeling and machine learning based analysis, interpretation, inferencing, and organization of data for a range of applications, including image content analysis, content-based image search, aesthetic quality inference, entity resolution, bioinformatics, business analytics, computer systems, and social networks.

Peer-Reviewed Journal Papers

- Ritendra Datta, Jia Li, and James Z. Wang, “Mixture of Preference Models for Content-based Aesthetics Inference,” To be submitted.
- Ritendra Datta, Jia Li, and James Z. Wang, “Adapting Automatic Image Annotation via Meta-learning,” Under review.
- Ritendra Datta, Jia Li, and James Z. Wang, “Exploiting the Human-Machine Gap in Image Recognition for Designing CAPTCHAs,” Under review.
- Ritendra Datta and Marshall Bern, “Spectrum Fusion: Using Multiple Mass Spectra for De Novo Peptide Sequencing,” *Journal of Computational Biology*, Accepted.
- Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang, “Image Retrieval: Ideas, Influences, and Trends of the New Age,” *ACM Computing Surveys*, vol. 40, no. 2, pp. 5:1–60, 2008.
- Ritendra Datta, Weina Ge, Jia Li, and James Z. Wang, “Toward Bridging the Annotation- Retrieval Gap in Image Search,” *IEEE Multimedia*, vol. 14, no. 3, pp. 24-35, 2007.