Adaptive revelance feedback for large-scale image retrieval

THÈSE N^{O} 5656 (2013)

PRÉSENTÉE LE 19 AVRIL 2013
À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR
LABORATOIRE DE L'IDIAP
PROGRAMME DOCTORAL EN GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Nicolae SUDITU

acceptée sur proposition du jury:

Prof. M. Grossglauser, président du jury
Dr F. Fleuret, directeur de thèse
Dr M. Ferecatu, rapporteur
Dr S. Marchand-Maillet, rapporteur
Prof. J.-Ph. Thiran, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2013

Acknowledgments

I spent four magnificent years in pursuing my doctoral studies, and I am very happy I have chosen this path. I express my deepest gratitude to my thesis director, Dr. François Fleuret, for being always inspiring, and actively brainstorming upon my ideas in a very refreshing way. I address my heartfelt thanks to Dr. Alessandro Vinciarelli, my former adviser, for trusting my determination, and giving me the opportunity to pursue this research. I am especially grateful to Prof. Donald Geman and Dr. Marin Ferecatu for their pioneering work that inspired my research, and their informal feedback in the course of my investigations. I convey my thanks to Prof. Matthias Grossglauser, Prof. Jean-Philippe Thiran, Dr. Marin Ferecatu and Prof. Stéphane Marchand-Maillet for agreeing to serve on my dissertation jury. I am thankful to everybody in the Idiap Research Institute and the Electrical Engineering Doctoral School at EPFL for supporting my work in various ways. I am also thankful to the Hasler Foundation for supporting my research. And lastly but most importantly, I extend my thanks to my family, parents, sister and grand-ma, whose unconditional love and continuous encouragement contributed to my doctoral endeavor like nothing else.

Lausanne, January 2013 N. Suditu

Abstract

Our research addresses the need for an efficient, effective, and interactive access to large-scale image collections. Image retrieval needs are evolving beyond the capabilities of the traditional indexing based on manual annotation, and the most desirable characteristic of any image retrieval system is to be able to deal with automatically-extracted visual indexing features, while providing an intuitive and simple interaction with users.

In this thesis, we investigate an innovative query-free retrieval approach that was proposed by Ferecatu and Geman. Starting from an heuristic sampling of the collection, this approach does not require any explicit query, neither keywords nor image-examples. It relies solely on an iterative relevance feedback mechanism driven by the user's subjective judgments of image similarities. At each iteration, the system displays a small set of images and the user is asked to choose the image that best matches in her opinion what she is searching for. The system updates an internal state based on automatically-extracted indexing features, and it displays a new set of images accordingly. The idea is that the system converges towards what the user is searching for, and iteratively it displays more and more relevant images.

Our contributions are related to four complementary aspects of the iterative relevance feedback mechanism. First, we formalize a large-scale approach based on a hierarchical tree-like organization of the images computed off-line. Second, we propose a versatile modulation of the exploration/exploitation trade-off based on the consistency of the system internal states between successive iterations. Third, we elaborate a long-term optimization of the similarity metric based on the user searching session logs accumulated off-line. Forth, we propose a dynamic short-term adaptation of the similarity metric based on the relevance feedback events accumulated on-the-fly at each iteration. Furthermore, we round up our research by integrating all our contributions together into one comprehensive retrieval system.

Experimental validation was carried out by implementing a web-application which includes all our contributions. This software is distributed to the public under the AGPL Version 3 open-source license. We carried out plenty of user-based evaluation campaigns, and we analyzed systematically all our contributions. We show empirically that each of them improves significantly the retrieval performance of the original framework. Moreover, we show that they are complementary to each other, and their overall integration is consistently beneficial.

We foresee that our contributions, along with our open-source web-application, will motivate further investigations and facilitate further experiments. We hope that our research brings the iterative relevance feedback mechanism one step closer to commercial applications.

Keywords

content-based image retrieval, query-free, large-scale, interactive relevance feedback, adaptive exploration/exploitation trade-off, log-based similarity learning, multi-modal indexing features, user-based evaluation.

Résumé

Ce mémoire traite de l'accès efficace et interactif à des collections d'images de grande taille. Les besoins en recherche d'images dépassent aujourd'hui les capacités des méthodes d'indexation traditionnelles qui reposent sur des annotations manuelles, et la qualité la plus désirable de tout système de recherche d'images est d'utiliser des descripteurs visuels d'indexation extraits de manière automatique, tout en étant simple et d'utilisation intuitive.

Nous avons étudié une approche innovante de recherche sans requête, qui fut proposée initialement par Ferecatu et Geman. Partant d'un échantillonnage heuristique de la collection, cette approche ne requiert aucune requête explicite, aucun mot-clé et aucun exemple. Elle repose uniquement sur l'information que fournit l'utilisateur sur la ressemblance entre les images qui lui sont montrées aux fil des itérations. À chaque étape, le système affiche un petit nombre d'images et demande à l'utilisteur de sélectionner celle qui selon lui, ressemble le plus à ce qu'il recherche. Le système met à jour un état interne basé sur des descripteurs d'indexation extraits de manière automatique, et affiche des nouvelles images en conséquence. Le système converge vers ce que l'utilisateur recherche, et affiche des images de plus en plus pertinentes.

Nos contributions portent sur quatre aspects complémentaires du mécanisme de retour itératif sur la pertinence. Premièrement, nous formalisons une approche grande échelle basée sur une arborescence pré-calculée des images. Deuxièmement, nous gérons le dilemme exploration/exploitation en prenant en compte la cohérence des états internes du système entre deux itérations successives. Troisièmement, nous optimisons hors ligne la mesure de similarité à l'aide de données collectées lors des sessions de recherches antérieures. Quatrièmement, enfin, nous proposons une adaptation dynamique à court terme de ladite mesure de similarité en nous basant sur le retour de pertinence acquis à chaque itération.

Les expériences de validation ont été menées grâce à une application web qui inclut l'ensemble de nos contributions. Ce logiciel est distribué au public sous une licence open-source AGPL Version 3. Nous avons mené de nombreuses campagnes d'évaluation avec des groupes d'utilisateurs, et nous avons analysé systématiquement les performances de toutes nos contributions. Nous avons montré que chacune d'elle améliore de manière significative la performance de la recherche initiale. De plus, nous avons montré qu'elles sont complémentaires, et que leur intégration conjointe est bénéfique.

Nous espérons que nos contributions ainsi que notre application web open-source amèneront à d'autres études et d'autres expériences. Finalement, nous espérons que les résultats de cette thèse rapprocheront le mécanisme itératif de retour sur la pertinence d'une utilisation commerciale.

Mots-clés

recherche d'images basée sur le contenu, grande base d'images, estimation de la pertinence, compromis exploration/exploitation, apprentissage de similarité, descripteurs d'indexation multimodaux, évaluation centrée sur l'utilisateur.

Table of Contents

Acknowledgments **iii**

Abstract (English/Français) **v**

Table of contents **ix**

List of figures **xiii**

List of tables **xvii**

Notation **xix**

1 Introduction **1**
- 1.1 Objective 1
- 1.2 Motivation 2
- 1.3 Contribution 3
- 1.4 Organization 7

2 Content-based image retrieval **9**
- 2.1 Image collections 9
- 2.2 Indexing information 10
  - 2.2.1 Textual information 10
  - 2.2.2 Contextual information 11
  - 2.2.3 Visual information 11
  - 2.2.4 Multi-modal information 12
- 2.3 Retrieval approaches 12
  - 2.3.1 Query-based retrieval 13
  - 2.3.2 Relevance feedback 13
  - 2.3.3 Query-free retrieval 14
- 2.4 Motivation for our work 15

3 Query-free retrieval framework **17**
- 3.1 Retrieval paradigm 17
- 3.2 Posterior probabilities of relevance 18
- 3.3 Selection of the displayed images 20

- 3.4 Similarity metric . . . 22
- 3.5 Experimental results . . . 23
  - 3.5.1 Intuitive analysis . . . 26
  - 3.5.2 User-based evaluation . . . 26
  - 3.5.3 Automatic tests . . . 28
- 3.6 Research issues . . . 28
- 3.7 Summary . . . 29

4 Large-scale HEAT framework **31**

- 4.1 Introduction . . . 31
- 4.2 State-of-the-art . . . 32
- 4.3 Scalable system . . . 33
  - 4.3.1 Tree and trace . . . 33
  - 4.3.2 Approximation of p_t . . . 35
  - 4.3.3 Trace refinement . . . 36
  - 4.3.4 Algorithm integration . . . 37
- 4.4 Experimental results . . . 37
  - 4.4.1 System setup . . . 39
  - 4.4.2 Evaluation scenario . . . 39
  - 4.4.3 Performance impact . . . 40
  - 4.4.4 Computational impact . . . 42
- 4.5 Summary . . . 44

5 Exploration/Exploitation Trade-off **45**

- 5.1 Introduction . . . 45
- 5.2 State-of-the-art . . . 46
- 5.3 Mass-zoom system . . . 48
  - 5.3.1 Exploration/exploitation trade-off . . . 48
  - 5.3.2 Heuristics based on a consistency score . . . 51
  - 5.3.3 Capabilities of the mass-zoom system . . . 52
- 5.4 Experimental results . . . 53
  - 5.4.1 System setup . . . 53
  - 5.4.2 Evaluation scenario . . . 53
  - 5.4.3 Results analysis . . . 55
- 5.5 Summary . . . 59

6 Log-based image similarity **61**

- 6.1 Introduction . . . 61
- 6.2 State-of-the-art . . . 62
- 6.3 Log-based similarity metric . . . 63
  - 6.3.1 Weighted Euclidean distance . . . 64
  - 6.3.2 Log-based weights learning . . . 64
- 6.4 Experimental results . . . 65

6.4.1 Log-based weights analysis . . . 66
6.4.2 Evaluation scenario . . . 69
6.4.3 Results analysis . . . 71
6.5 Summary . . . 71

7 Multi-modal image similarity 73
7.1 Introduction . . . 73
7.2 State-of-the-art . . . 74
7.3 Joint visual and textual-based metric . . . 75
7.3.1 Bi-modal adaptive metric . . . 75
7.3.2 Weight estimation . . . 76
7.4 Experimental results . . . 76
7.4.1 System setup . . . 77
7.4.2 Evaluation scenario . . . 77
7.4.3 Results analysis . . . 78
7.5 Summary . . . 81

8 System integration 83
8.1 Integration overview . . . 83
8.2 Experimental results . . . 84
8.2.1 System setup . . . 85
8.2.2 Evaluation scenario . . . 85
8.2.3 Results analysis . . . 85
8.2.4 Informal discussions . . . 90
8.3 Summary . . . 94

9 Conclusion and foresight 95
9.1 General summary . . . 95
9.2 Further research . . . 96
9.3 Suitable applications . . . 98
9.4 Closing note . . . 98

Appendices 101

A Web-application 101
A.1 Functionality . . . 101
A.2 Implementation . . . 103
A.3 Remarks . . . 107

B Test platform 109
B.1 Functionality . . . 109
B.2 Implementation . . . 110
B.3 Automatic tests . . . 112
B.4 Remarks . . . 113

C Image collections **115**
C.1 Corel stock photo library 115
C.2 ImageNet dataset 117
C.3 Synthetic collection 119

Bibliography **121**

Curriculum vitae **127**

List of Figures

1.1 Organization of the thesis, showing the chapter inter-relationships 6

3.1 Starting a new searching session. The user does not submit any query, neither keywords nor image-examples. After a simple trigger, the first iteration $t = 0$ starts. The system initializes $p_0(k) = 0.5$ for all $k \in \Omega$, and then selects the first display set D_0. 19

3.2 Relevance feedback loop. At iteration t the system displays D_t. The next iteration $t+1$ is triggered by the relevance feedback event $\{D_t, x_t^*\}$. The system will update $p_{t+1}(k)$ for all $k \in \Omega$, and then will select the new display set D_{t+1}. 19

3.3 Calibration functions and their parameters 22

3.4 The posterior probabilities $p_t(k)$ for all $k \in \Omega$ are updated iteratively. Here the evolution of the probabilities is illustrated via the synthetic image collection. One can see how the distribution of probabilities evolves towards matching the user retrieval objective. 24

3.5 The set of displayed images is generated via the Voronoi tessellation algorithm. Here the intermediate steps of the algorithm are illustrated via the synthetic image collection. 25

4.1 Relation between the hierarchical tree and the trace adaptive partitioning . . . 34

4.2 Evolution of the trace for the synthetic collection, when searching for images with points close to the center 38

4.3 Retrieval performance of the HEAT framework 41

4.4 Timing of the system responses (in seconds) as the users experienced them during the evaluations 43

5.1 Evolution of the distribution of probabilities of relevance for the searching session illustrated in Figure 5.2 47

5.2 Evolution of the display set for the original framework with the synthetic collection, when searching for images with points close to the center 49

5.3 Evolution of the display set for the mass-zoom system with the synthetic collection, when searching for images with points close to the center 50

5.4 Consistency scores are estimated based on the cumulative distribution function for the Gaussian distribution. Our heuristic gives a consistency score in the [0.5, 2.0] interval. 51

5.5 The users were asked to search for semantic categories described in words and accompanied by image examples as shown here. In order to ensure a sufficiently reliable diversity, there were 6 semantic categories. 54
5.6 Retrieval performance of the mass-zoom system for L^2 and L^2-Iso16 similarity metrics 56
5.7 Retrieval performance of the mass-zoom system for L^1 and L^1-Iso16 similarity metrics 57
5.8 Zoom-factor average and standard deviation 58

6.1 Histogram of the feature weights in the optimal weighting vector $\boldsymbol{\alpha}$ 67
6.2 Cumulative distribution of the similarity distances in the collection 67
6.3 Here we recall the calibration functions in Figure 3.3 68
6.4 Influence of the calibration parameters on the cost function 68
6.5 Retrieval performance of the log-based similarity metric 70

7.1 Calibration functions for each modality separately 76
7.2 Retrieval performance of the multi-modal similarity metric for the first group of users 79
7.3 Retrieval performance of the multi-modal similarity metric for the second group of users 80

8.1 Retrieval performance of the integration of mass-zoom system and the log-based similarity metric with the 60K image collection 86
8.2 Retrieval performance of the HEAT framework in combination with the log-based similarity metric 87
8.3 Evolution of the comprehensive system for the synthetic collection, when searching for images with points close to the center 88
8.4 Retrieval performance of the comprehensive system that integrates all our contributions 91
8.5 Retrieval performance of the comprehensive system for three different trace sizes. The bigger the trace, the better the performance, but the difference between 1,000 vs. 1,500 is smaller than the difference between 500 vs. 1,000. 92
8.6 Timing of the comprehensive system responses (in seconds) as the users experienced them during the evaluations 93

A.1 Client-server architecture of the retrieval system 104
A.2 Screenshot of the web-interface for the searching application 104
A.3 Screenshot of the web-interface for the testing application 105
A.4 Screenshot of the web-interface for the statistics application 105

B.1 Architecture of the test platform 110
B.2 Retrieval performance of the automatic tests in comparison with the user evaluations for the original system and the mass-zoom extension 111

B.3 Retrieval performance of the automatic tests in comparison with the user evaluations for the HEAT system and the mass-zoom extension 111

C.1 Examples of images and their associated keywords from the Corel stock photo library 116
C.2 Screenshot of the ImageNet exploration tool web-interface 118
C.3 Visual SIFT-based image features provided by ImageNet 118
C.4 Abstract representation with a synthetic collection, where each image has as visual content one single point in the 2D Cartesian space, and the indexing features are the corresponding coordinates of that point 119
C.5 Examples of images in the synthetic collection. Actually, these 8 images are the images in the display set D_0 that is used consistently as the first display set in all the illustrated searching sessions. 120
C.6 Abstract representations of D_0 shown above in Figure C.5 120
C.7 More abstract representations of D_0 shown above in Figure C.5 120

List of Tables

3.1 Notation for the original framework 19
3.2 Procedures to select a meaningful set of images D_t to be displayed next given the current estimate of probabilities $\mathbf{p} = \{p_t(k)\ \forall k \in \Omega\}$ 22

4.1 Notation for the HEAT retrieval framework 34
4.2 Retrieval performance of the HEAT framework. Here are a few discrete values read from Figure 4.3 41

5.1 Notation for the mass-zoom extension 51
5.2 Binomial-test for statistical significance corresponding to Figures 5.6-5.7 55
5.3 Retrieval performance of the mass-zoom system for L^2 and L^2-Iso16 similarity metrics. Here are a few discrete values read from Figure 5.6 56
5.4 Retrieval performance of the mass-zoom system for L^1 and L^1-Iso16 similarity metrics. Here are a few discrete values read from Figure 5.7 57

6.1 Notation for the log-based similarity learning 63
6.2 Retrieval performance of the log-based similarity metric. Here are a few discrete values read from Figures 6.5 70

7.1 Notation for the adaptive multi-modal similarity metric 75
7.2 The 12 semantic categories described only in words 77
7.3 Retrieval performance of the multi-modal similarity metric for the first group of users. Here are a few discrete values read from Figure 7.2 79
7.4 Retrieval performance of the multi-modal similarity metric for the second group of users. Here are a few discrete values read from Figure 7.3 80
7.5 Binomial-test for statistical significance corresponding to Figure 7.2 81
7.6 Binomial-test for statistical significance corresponding to Figure 7.3 81

8.1 Retrieval performance of the HEAT framework in combination with the log-based similarity metric. Here are a few discrete values read from Figure 8.2 . . . 87
8.2 Retrieval performance of the comprehensive system that integrates all our contributions. Here are a few discrete values read from Figure 8.4 91
8.3 Retrieval performance of the comprehensive system for three different trace sizes. Here are a few discrete values read from Figure 8.5 92

A.1 System configuration data 102
A.2 Target task data 102
A.3 Searching session data 102

Notation

Ω	image collection, where the images are identified by their indexes $\{1, 2, \ldots\}$
k, h, x	images or image feature vectors
$S \subset \Omega$	set of images that the user is searching for
$D_t \subset \Omega$	set of images shown to the user at iteration t
$x_t^* \in D_t$	image chosen by the user at iteration t
$\{D_t,\ x_t^*\}$	relevance feedback event at iteration t
$p_t(k)$	probability of relevance of image k at iteration t
$\mathcal{N}$	complete set of nodes of the hierarchical tree
$\mathcal{T}_t$	trace at iteration t
$\Omega(N)$	set of images associated with node N
k_N^*	representative image of node N
m_t	target mass for building the display set in the original system
m_t^{zoom}	target mass in the mass-zoom approach
c_t	consistency score at iteration t
$\boldsymbol{\alpha}$	weighting vector learned off-line from the user logs
w_t	weight of the bi-modal adaptation at iteration t

1 Introduction

The expansion of the World Wide Web, accompanied by inexpensive recording capabilities and mass storage and sharing tools, facilitate the public access to multimedia data of unprecedented size. Some of the largest on-line repositories for such data that include different modalities like images, audio, video and text, are Flickr, YouTube, FaceBook, Twitter, and of course the World Wide Web as a whole. Such an amount of information creates enormous possibilities and challenges at the same time. While it is easy to share and access everything anytime, it is hard to search and find anything specific.

This thesis is related to content-based image retrieval, and investigates a novel retrieval approach proposed initially by Ferecatu and Geman [24, 25] which has the major advantage of being query-free. It does not require any explicit query, neither keywords nor image examples, and relies solely on an iterative relevance feedback mechanism. At each iteration, the system displays a small set of images, and the user chooses one image that best matches what she is searching for. The idea is that the system converges , and iteratively displays more and more relevant images.

In this chapter, we state the scope of the thesis, and give an overview of our contributions. First, we motivate the scope of our thesis in the research field of content-based image retrieval. Then, we provide an overview of our contributions, and cite our related publications.

1.1 Objective

Image retrieval, as well as multimedia retrieval in general, has changed considerably in the last decade due to the expansion of the World Wide Web accompanied by inexpensive recording, storage and sharing capabilities. A decade ago, the largest image collections were stock photography collections such as Getty Images and Corbis, containing hundreds of thousands of images carefully annotated with keywords from a well specified vocabulary by experts with a homogeneous and professional knowledge. Nowadays, the on-line image collections such as Flickr and FaceBook are orders of magnitude larger.

Content-based image retrieval has been under active research for a few decades, and although great progress has been made and many retrieval approaches have been proposed, there are still many questions that remain open in both the perceptual cognitive and the algorithmic technical aspects. Regarding the cognitive aspect, novel similarity measures or rankings are needed to capture better the human perception of image similarities. Regarding the technical aspect, novel algorithms are needed to compute or approximate efficiently such similarity measures at large-scale.

The broad objective of this thesis is to investigate new ways for an efficient, effective and interactive access to large-scale image collections, and thus to contribute to (the technical aspect of) the research field of content-based image retrieval. Our aspiration was to contribute to the retrieval paradigm based on iterative relevance feedback. First, we aimed to contribute to the efficiency of various algorithms, by analyzing and improving their technical characteristics and mathematical properties. Then, we aimed to contribute to the usability of various interfaces, by evaluating and enhancing the user experience. In overall, we aimed to bring the iterative relevance feedback paradigm one step closer to commercial applications.

1.2 Motivation

There is a noticeable need for retrieval systems that are able to provide efficient access to these large-scale image collections containing billions of items. Since it is virtually impossible to annotate manually all images, the image retrieval needs are evolving beyond the capabilities of the straight-forward text-based retrieval systems in both public and private domains. In this regard, the most desirable characteristic of any image retrieval system is to be able to deal with automatically extracted visual-based features, while providing an intuitive and simple interaction with users.

Research has begun to tackle this challenge via automatic tagging based on annotation propagation [56, 39, 65]. However, formulating a query might not be the most efficient way of searching for images since the visual content is often difficult to describe in terms of keywords. Relevance feedback is indeed envisioned by many researchers as the only alternative that could cope properly with the challenges in image retrieval, and multimedia retrieval in general [51, 69, 14].

We were inspired by an innovative retrieval approach proposed initially by Ferecatu and Geman [24, 25] which has the major advantage of being query-free. It does not require any explicit query, neither keywords nor image examples, and relies solely on an iterative relevance feedback mechanism. At each iteration, the system displays a small set of images, and the user chooses one image that best matches what she is searching for. The system updates an internal state, and displays a new set of images accordingly. In this process, the system converges towards what the user is searching for, and after a few iterations the displayed sets start to include more and more images that satisfy the user.

The motivation for a query-free retrieval approach comes from the observation that formulating a query might not be the most optimal way of initializing a searching session. On the one hand, the user retrieval needs are often difficult to describe in terms of keywords. One may need to be more specific than what two-three keywords can capture or, even worst, one cannot express in keywords what she is searching visually. On the other hand, relevant images may be easily filtered out since any query is inherently incomplete and sub-optimal. The user has to be familiar with the keyword vocabulary and to understand the underlying indexing in order to be able to re-formulate manually a more optimal query and to steer/refine the search. By hiding entirely the indexing features, the query-free user interface is minimalist and self-explanatory.

1.3 Contribution

Our research in this thesis focuses on extending and reshaping various aspects of the retrieval framework proposed initially by Ferecatu and Geman [24, 25]. We have investigated the state-of-the-art approach in four complementary aspects, namely large-scale distributed system architecture, exploration/exploitation trade-off, image similarity learning and multi-modal image features. The key achievements are as follows.

Large-scale HEAT framework

The original approach requires at each iteration a computational effort that is tightly related to the size of the image collection Ω. On the one hand, the probabilities of relevance are computed for all the images in the collection, and this implies $\mathcal{O}(\|\Omega\|)$ complexity. On the other hand, the selection of the displayed images involves sorting operations of $\mathcal{O}(\|\Omega\| \cdot \log \|\Omega\|)$ complexity over the entire collection.

We propose an approach that effectively decouples the computational effort from the size of the collection, and yet preserves the retrieval capabilities. Using an adaptive partitioning of the image collection, we provide the means for controlling the trade-off between the retrieval performance and the computational effort. This retrieval approach promises an interactive access to image collections of unprecedented size.

The evaluation was organized on a collection of 1,000,000 images from the ImageNet database [18]. The experiments show that our system provides a sustainable performance where the original system proposed by Ferecatu and Geman [24, 25] would cease to function within any reasonable time-frame.

- Nicolae Suditu and François Fleuret. HEAT: Iterative relevance feedback with one million images. In *Proceedings of the IEEE International Conference on Computer Vision* (ICCV), pages 2118–2125, November 2011.

Exploration/exploitation trade-off

Content-based image retrieval systems have to cope with two different regimes: understanding broadly the categories of interest to the user, and refining the search in this or these categories to converge to specific images among them. As argued by Ferecatu and Geman [24, 25], the original approach is well suited for image category search and that is, in other words, the first retrieval regime of exploring the image collection, but it is quite unsuitable for the second regime.

We propose an approach that encompasses these two regimes, and infers from the user actions a smooth transition between them. We introduce the idea of an adaptive modulation of the exploration/exploitation trade-off that transforms the original approach into a versatile retrieval framework with full searching capabilities.

Our approach is compared to the state-of-the-art approach it extends by conducting user evaluations on a collection of 60,000 sampled uniformly from the ImageNet database [18]. Evaluation gives evidence that our approach brings a significant improvement to the retrieval capabilities beyond finding an image category, and is able to support refining the user interest in an efficient manner.

- Nicolae Suditu and François Fleuret. Iterative Relevance Feedback with Adaptive Exploration/Exploitation Trade-Off. In *Proceedings of the ACM International Conference on Information and Knowledge Management* (CIKM), pp.1323–1331, October 2012.

Log-based similarity metric

Interactive image retrieval based on user relevance feedback strongly depends on the extent to which the “closeness” in the similarity metric (i.e. the distances between the image feature vectors) accounts for the “closeness” in the subjective perception of the users.

We propose to improve the visual-based similarity metric on which the original framework relies by learning from user logs to adapt the low-level image features and model explicitly the user similarity judgments.

Our technique is evaluated on two collections from ImageNet, a small collection of 60,000 images and a large collection of 1,000,000 images, and shown to bring about 10% improvement in the retrieval performance.

- Nicolae Suditu and François Fleuret. Interactive Image Retrieval with Log-based Similarity Learning. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition* (CVPR), July 2013, *submitted.*

Multi-modal similarity metric

The original approach uses a rigid similarity metric based on low-level indexing features extracted from the images visual content (i.e. global descriptors of color, texture and shape). The alignment with the user subjective perception of image similarities is done via a linear calibration that is invariant during the searching sessions.

We propose an extension that integrates indexing features extracted from both the images visual content and their accompanying annotation keywords. This is motivated by the intuition that the retrieval needs are sometimes modeled better by visual features, sometimes by textual features, and sometimes by a combination of both.

Our approach is evaluated on a collection of 35,000 images from the COREL database, and shown to be intuitive, easy to use and efficient. The system succeeds to retrieve images that satisfy the users in less than 5 iterations in 60% of the cases.

- Nicolae Suditu, Alessandro Vinciarelli and François Fleuret. Query-Free Interactive Image Retrieval Based on Visual and Textual Features. *Idiap Research Institute internal report*, 2010.

System software design

We invested a considerable amount of effort in developing the retrieval system as a web-application. Besides the advantage of permanent availability for evaluations, this implementation encourages the adherence to a realistic system architecture.

The implementation has about 12,000 lines of code written in Python based on the Django platform, and it complies with the PEP8 code style standard. The indexing information and the user logs are stored in a relational database based on MySQL. Besides the searching functionality, there is infra-structure for user-based evaluation purposes that handles user accounts, stores evaluation data and computes statistics.

The application software has been published under the AGPL Version 3.0 open-source license. We hope that our code adds transparency to our research work, facilitates the reproducibility of our experiments, and offers a good foundation for further investigations.

- Nicolae Suditu, http://www.idiap.ch/software/imr/, software release copyrighted by Idiap Research Institute, available under the AGPL Version 3 open-source license, 2010–2012.

- Nicolae Suditu, http://imr.idiap.ch/, demo web-application, 2009–2012.

System integration

We rounded up our research by investigating the integration of all contributions together into one comprehensive retrieval system. Our contributions touch different components of the retrieval system, and from the retrieval performance point of view their integration makes a lot of sense.

In order to evaluate the overall retrieval performance, we organized several user-based evaluation campaigns in the same manner as for each individual contribution, and we evaluated systematically different combinations of our contributions. We got evidence that each contribution complement each other, and their combination improves consistently the retrieval performance for both small and large collections.

- Nicolae Suditu and François Fleuret. Adaptive Relevance Feedback for Large-scale Image Retrieval. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (TPAMI), 2013, *work in progress.*

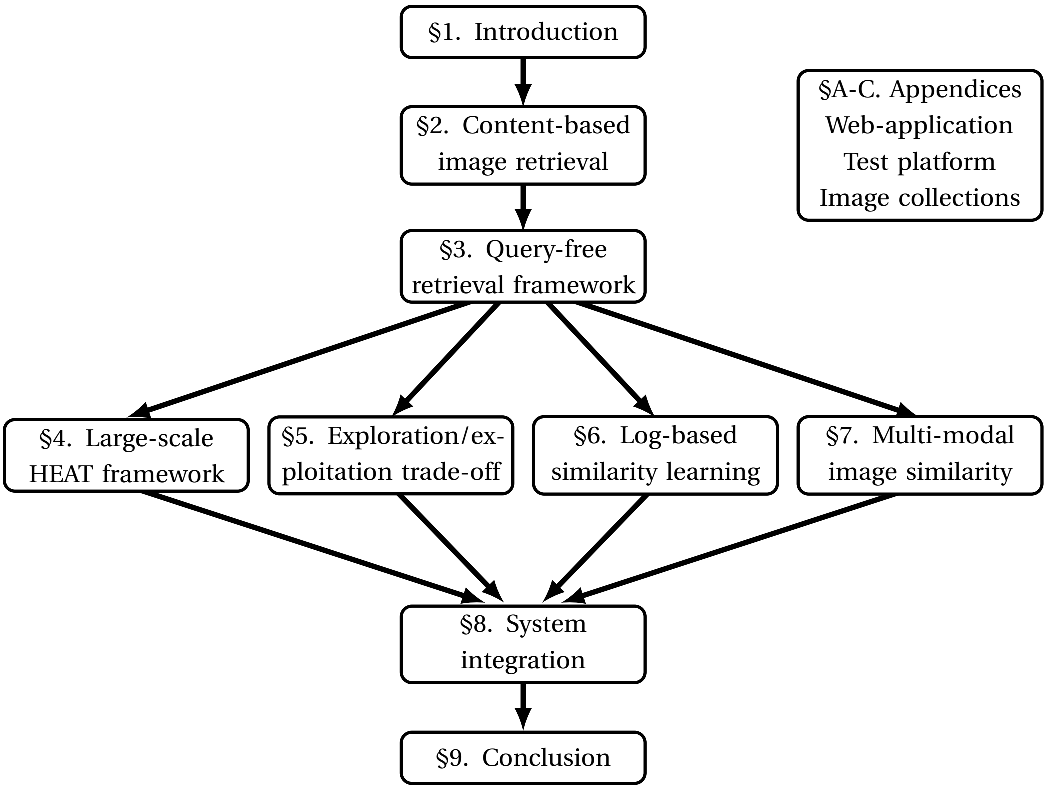

§1. Introduction
§2. Content-based image retrieval
§3. Query-free retrieval framework
§A-C. Appendices
Web-application
Test platform
Image collections
§4. Large-scale HEAT framework
§5. Exploration/exploitation trade-off
§6. Log-based similarity learning
§7. Multi-modal image similarity
§8. System integration
§9. Conclusion

Figure 1.1: Organization of the thesis, showing the chapter inter-relationships.

1.4 Organization

So far in this Chapter §1 we have stated the general scope of the thesis, and we gave a brief overview of our contributions. The structure of the rest of the thesis is straight-forward following our contributions as sketched in Figure 1.1.

Chapter §2 defines and motivates our research topic by an overview of the relevant state-of-the-art. First, we provide a brief overview of the research field of content-based image retrieval. Then, we provide a closer look at the state-of-the-art in relevance feedback, and how our work is positioned in this research landscape.

Chapter §3 describes the retrieval framework that is central to our work. First, we present in detail the query-free retrieval approach, and provide the theoretical justifications for the component algorithms. Then, we provide the first intuitive analyses that motivate the research directions we have chosen to pursue further on.

The next chapters present our contributions. Chapter §4 presents the large-scale HEAT framework. Chapter §5 presents the exploration/exploitation trade-off. Chapter §6 presents the user-based similarity learning, and Chapter §7 presents the adaptive multi-modal similarity metric. Chapter §8 presents the system integration.

Chapter §9 summarizes our work, and opens new directions for further research. First, we conclude the thesis with a final overview of our contributions. Then, we outline and motivate a few potential directions for future work.

In the Appendices, we present some of the secondary contributions of this thesis. Appendix §A describes the web-application and all its components, and mentions our software development choices. Appendix §B describes the test platform that we have used thoroughly throughout our work. Appendix §C describes the three image collections that we have used in our research for automatic simulations and user-based evaluations, namely Corel Stock Photo Library, ImageNet dataset and our synthetic image collection.

2 Content-based image retrieval

Image retrieval, as a field of multimedia information retrieval, resides at the intersection of various disciplines such as computer vision, machine learning, information retrieval, human-computer interaction, database systems and psychology [64]. We are approaching this field from the engineering point of view, and we are trying to pay attention to aspects from other disciplines as much as possible.

In this chapter, we define and motivate our research topic by an overview of the relevant state-of-the-art. First, we provide a brief overview of the research field of content-based image retrieval. Then, we provide a closer look at the state-of-the-art in relevance feedback mechanisms, and position our research work.

2.1 Image collections

The scale of nowadays collections is hard to grasp even for professionals. Photo-sharing on the Web keeps getting more and more popular. Facebook[1] has over 15 billion images, and Flickr[2] over 4 billion images. Surprisingly, these well-known sharing platforms are not the largest in terms of the number of images, and for example Image Shack[3] hosts more than 20 billion images. These numbers are meant only to get the idea, as they become out-dated as soon as they are written down. These collections are not only huge, but also continue to grow at fast rates. For example, Facebook users are adding photos at a rate of 850 million photos a month.[4]

We adhere to the assumption that large-scale collections are not only large, but also inherently un-structured (i.e. lacking any semantic or thematic indexing as in the archived libraries) and continuously out-dated (i.e. images are frequently being added, replaced or removed). Thus, our affinity is towards retrieval approaches that could accommodate such assumptions.

---

[1] http://www.facebook.com/

[2] http://www.flickr.com/

[3] http://imageshack.us/

[4] http://techcrunch.com/

2.2 Indexing information

The value of a collection depends on its accessibility, which in turn depends on a corresponding relevant indexing. A decade ago, the largest image collections were the stock photography collections such as Getty Images[5] and Corbis[6] containing hundreds of thousand images. Images were annotated carefully with keywords from a well specified vocabulary by people with a homogeneous and professional knowledge.

Nowadays, the on-line image collections such as Flickr[7] or FaceBook[8] are orders of magnitude larger. Although many images are annotated, the keywords are less reliable due to subjective perception and less consistent due to uncontrolled vocabulary. Moreover, it is almost impossible to annotate manually all images. In order to build satisfactory retrieval applications, the indexing information should be acquired automatically.

Modern digital technologies creates enormous possibilities and challenges at the same time. Nowadays, the on-line image collections such as Flickr[9] or FaceBook[10] report growings of millions of images per week. Although many images are annotated, it is virtually impossible to annotate manually all of them. Still, the value of a collection depends on its accessibility, which in turn depends on its indexing. In order to build large-scale retrieval applications, the indexing information should be generated automatically from the images themselves.

There are two main paradigms for using the visual-based features. The first paradigm is to use them directly in order to define some similarity metric between images, in the same way as textual-based features are used. The second paradigm is to use them indirectly, to translate automatically the visual-based features into textual information as a pre-processing operation [35] via automatic image interpretation [19] or annotation propagation [44].

Many visual-based features have been proposed to characterize globally color distributions, textures and edge layouts [42]. Probably the most known are the visual features based on SIFT (Scale Invariant Feature Transform) [41], which is highly distinctive and robust to affine transformations and limited changes in illumination and 3D viewpoint.

2.2.1 Textual information

The main source of textual information that is exploited by the existing search engines (e.g. Yahoo!, Google) is the image's captions or the paragraphs found in the proximity of the images as they are arranged in multimedia documents (e.g. web-pages with news, articles, reviews) [36]. Photo-sharing repositories and social networking web-sites support and encourage the users

---

[5] http://www.gettyimages.com

[6] http://www.corbisimages.com

[7] http://www.flickr.com

[8] http://www.facebook.com

[9] http://www.flickr.com

[10] http://www.facebook.com

to provide annotation keywords for their own images and write feedback comments for the images of other users [12].

Textual-based features have been extensively investigated researched for text retrieval systems and, basically, any such features can be imported by the image retrieval systems. After the textual information is cleaned up by parsing, stemming and other techniques, the textual-based feature vectors are constructed to reflect the presence of the indexing terms [52]. My research work done so far is using textual features based on LSA (latent semantic analysis) [17]. LSA takes advantage of the implicit associations between keywords, and it escapes the unreliability, ambiguity and redundancy of individual keywords.

2.2.2 Contextual information

Modern digital equipment associates automatically images with meta-data such as date/time, location (i.e. Global Positioning System coordinates), and acquisition technical details (e.g. device type and configuration, resolution, luminosity, exposure settings) [16].

Photo-sharing repositories and social networking web-sites accumulate meta-data such as the number of accesses/viewings, number of references/links, popularity rankings or comments of users. These kinds of contextual information could be exploited in retrieval applications [15].

Another source of information that is currently emerging, at least in the research community, is the implicit tagging from the nonverbal behavior displayed by users while interacting with multimedia data (e.g. facial expressions, vocal outbursts) [63].

2.2.3 Visual information

However, the textual and contextual information cannot fully characterize the visual content of the images. Making abstraction of its feasibility, the manual annotations are subjective and incomplete by nature. For this reason, there have been proposed the use of image processing techniques to capture automatically the visual content of images [38]. The IBM QBIC project [26] developed in 1995 is regarded as the pioneer of visual-based retrieval systems.

The visual content, or appearance, of the images is described mathematically in vector spaces based on image processing techniques (e.g. global color, texture and shape information, or a combination of these). Contrary to textual information, the visual information is abstract and does not allow for intuitive search. There is no unique explanation for the difficulties encountered in content based retrieval [56].

- The concept of "semantic gap" has been extensively used in the research community to express the discrepancy and un-correlation between the abstract vectorial representations and the actual semantic interpretation of the visual content. That is why these abstract representations were called low-level features in the first place.

- The "numerical gap" refers to the incapacity of the low-level features to characterize sufficiently the visual content of the images in order to discriminate appropriately between "relevant" and "irrelevant" images.

There are two main paradigms for using the visual-based features. The first paradigm is to use them directly in order to define some similarity metric between images, in the same way as textual-based features are used. The second paradigm is to use them indirectly, to translate automatically the visual-based features into textual information (e.g. annotation keywords) as a pre-processing operation [35]. The main idea is to achieve automatic annotation via automatic image interpretation [19] or annotation propagation [44]. Automatic translation is a complex task, involving computer vision problems such as object recognition, and this research direction make slow progress.

Many visual-based features have been proposed to characterize globally the color distribution, texture and edge layout and many are already included in the MPEG-7 standard [42]. My research work is using visual features based on SIFT (Scale Invariant Feature Transform) [41]. SIFT feature vectors are highly distinctive and robust to affine transformations, changes in illumination and limited changes in 3D viewpoint.

2.2.4 Multi-modal information

In the recent years, research confirmed that both visual-based and textual-based features have inherent limitations, and the retrieval systems are better off if they exploit both feature types in a multi-modal fashion, in order to compensate each other for their own limitations (see Smeulders et al. [56]).

The simplest approach is to simply concatenate the visual-based and textual-based features or to combine them in other rigid manner in order to obtain *composite features*. Since the results have been encouraging [37, 23, 54], they motivated the research of more advanced combinations such as *dynamically weighted features* [68].

2.3 Retrieval approaches

There are two extreme image retrieval needs: exploration and exploitation. The expectations of the users will always be somewhere in between, and always different. Ideally, the retrieval systems should support a seamless transition between them.

- The "exploration" need refers to the case when the user wants to browse the collection while committing to a rather vague notion of relevancy that may vary over time.

- The “exploitation” need refers to the case when the user wants to find all the images that share some specific characteristics.

Most of the retrieval systems offer a standard pipe-line of retrieval approaches. A search session is initiated by submitting a query to the retrieval system. The most common type of query is a set of keywords, as in the case of text retrieval. After retrieving the first results, some systems offer relevance feedback tools that support the user in refining the results in an iterative manner. Some systems offer more complex interfaces for tuning algorithm parameters or profile/preference parameters.

2.3.1 Query-based retrieval

The classical image retrieval approach was to annotate each image manually based on a limited vocabulary of keywords (i.e. to create manually the textual information) and, basically, to reduce image retrieval to text retrieval and to make use of the well-known and well-researched *query-by-keywords* approach [10, 59]. As in the case of text retrieval, formulating a query is more suited for the exploitation stage than for the exploration stage of the retrieval process. For exploration, the user must rely on her creativity to reformulate queries and to understand the indexing miscarriages.

The trend in the recent years shows that image retrieval systems must evolve beyond the capabilities of the straight-forward text-based surrogates [50]. Formulating a query might not be anymore the most efficient way of searching for images. If the annotation keywords are not fully consistent, even the most optimal query may easily exclude relevant images and include non-relevant images. Moreover, users not familiar with the keywords vocabulary will likely formulate only sub-optimal queries. All these difficulties add on top of the fact that the retrieval needs are often difficult to describe in terms of keywords.

In consequence, research proposed alternative approaches that use the visual-based features directly in the indexing/retrieving operations. The main idea consists of specifying the query as a set of feature vectors and, then, searching the collection for the best match. The difficulty is now shifted into specifying such abstract queries, which can be done only indirectly. The most generic meanings are *query-by-visual-examples*, in which the user must provide image examples similar to what he is searching for [57], and *query-by-sketching*, in which the user must hand-draw some simple colors, textures or shapes [26]. These unconventional types of queries have their own limitations by assuming suitable image examples at hand or reasonable drawing skills [3].

2.3.2 Relevance feedback

Another way of identifying what the user is searching for is by using relevance feedback mechanisms. In general, relevance feedback is any information about the retrieved results, given by users to a retrieval system. Whereas introduced in text retrieval [29], relevance feedback has attracted more considerable attention in the content-based image retrieval. In fact, relevance feedback is envisioned by many researchers as the only alternative that could cope properly with the challenges in image retrieval [51, 69, 14]. Replacing the burden of formulating explicit complex queries, or having good image examples at hand, by some similarity judgments is very appealing in this new field.

One could think to make use of many sorts of information from subsequent retrieval sessions [14]. In my research, relevance feedback refers only to the information acquired in the current retrieval session by including the user in the retrieval loop [49]. For this, the session is divided into several consecutive iterations; at every round the user provides feedback regarding the retrieval results, labeling relevant images (i.e. positive feedback) and sometimes also non-relevant images (i.e. negative feedback). The system use this new information in order to refine the results.

While early works in MARS [7] and MindReader [33] developed mechanisms for rich feedback information (e.g. ranking many images, tuning many parameters), the current consensus is that mechanisms should deal with scarce feedback (e.g. marking a few “relevant” images and no tuning parameters) [13]. Obviously, the minimalist relevance feedback mechanism would require marking as “relevant” one single image at each iteration.

As reported in surveys [56, 62], there are many content-based image retrieval systems in research form but very few have been commercially developed. Scalability is crucial for an image retrieval system to be practical and realistic [67]. Some of the recently proposed pro-scalable approaches are Vima’s Image Search Engine [66], Virage VIR Image Engine [2] and Cortina [48].

2.3.3 Query-free retrieval

The innovative idea of searching images without any explicit query appeared in the work of Cox et al. [13]. The backbone of their approach was a relevance feedback mechanism based on a Bayesian framework. Fang and Geman [21] and Ferecatu and Geman [24, 25] extended the Bayesian framework and provided theoretical justifications for the main algorithms.

Starting from an heuristic sampling of the collection, this approach does not require any explicit query. It relies solely on an iterative relevance feedback mechanism. At each iteration, it displays a small set of images and the user is asked to show the image that best matches what he is searching for. After a few iterations, the displayed set starts to include images that satisfy the user. By hiding entirely the indexing features, the user interface is effortless and self-explanatory. Moreover, this approach is intuitively suitable to support a seamless transition between the exploration stage and the exploitation stage of the retrieval process.

Sharing the same line of thinking, a perception-based image retrieval system was developed by Chang et al. [8]. In essence, the system models the user retrieval needs as feature grouping of *k*-CNF/DNF Boolean form. Starting without any explicit query, it requires an iterative rich relevance feedback consisting of positive and negative labeled images. The displayed sets of images for conducting an efficient and moderately robust to noise relevance feedback process are selected intelligently by an active learning algorithm presented in Chang and Li [9].

Query-free retrieval falls naturally under the ostensive model which was researched by Campbell and Rijsbergen [6] and more recently by Urban and Jose [61] The ostensive model works mainly with the assumption that the user information need is dynamic and developing, and thus the recent relevance feedback is more indicative to the current information need than the relevance feedback given previously. What they propose is an adaptive query learning scheme based on textual and visual-based features, which is used in order to avoid the query formulation process, and thus bridge the semantic gap more naturally.

2.4 Motivation for our work

In this chapter, we have introduced very briefly the research field of content-based image retrieval. Then, we took a closer look at the state-of-the-art in visual-based indexing features and the retrieval approaches able to cope with the current technological shifts in recording capabilities and mass storage and sharing tools.

We adhere to the assumption that large-scale collections are not only large, but also inherently un-structured (i.e. lacking any semantic or thematic indexing as in the archived libraries) and continuously out-dated (i.e. images are frequently being added, replaced or removed). Thus, our affinity is towards retrieval approaches that could accommodate such assumptions. On the one hand, the retrieval solutions should be computationally scalable in both off-line and on-the-fly operations. On the other hand, the indexing information should support incremental updates, without requiring updates from scratch each time something changes.

We were inspired in special by query-free retrieval approaches as the one proposed by Ferecatu and Geman [24, 25]. The user does not have to be familiar with the indexing vocabulary and to understand the underlying indexing technique in order to be able to re-think a more optimal query and to refine manually the search. By hiding entirely the indexing features, the query-free user interface has the potential of being minimalist and self-explanatory.

Only a few of the existing retrieval systems are powered specifically by relevance feedback tools. Although research agrees on their potential benefits, the public image search engines provide very limited functionality of this kind. As reported in surveys [56, 62, 15], more research is needed for achieving maturity in terms of efficiency, usability and scalability, which are essential characteristics for a successful system.

3 Query-free retrieval framework

The innovative retrieval framework that is central to our research was proposed initially by Ferecatu and Geman [24, 25]. Starting from an heuristic sampling of the collection, this approach does not require any explicit query, neither keywords nor image-examples. Based on an interactive relevance feedback mechanism, the system converges iteratively towards what the user is searching for, and in this process it displays iteratively more and more images relevant to the user.

In this chapter, we present this interactive query-free retrieval framework, and provide the theoretical justifications for the component algorithms as in [24, 25]. Basically, the retrieval framework embodies an iterative relevance feedback mechanism that has two components. First, there is a Bayesian model that estimates the probability of relevance of any image in the collection as a conditional probability given the relevance feedback events. Second, there is a strategy for selecting what images to show next given the estimates of the probabilities of relevance of all the images in the collection.

Along the summary of the original approach, we motivate the research directions we have chosen to pursue further on. We provide the first intuitive analyses of the system components, and open the discussion about the limitations of the original approach. We also introduce the experimental setup of our user-based evaluation campaigns.

3.1 Retrieval paradigm

Given a collection of images $\Omega = \{1, 2, \ldots\}$, the objective of the retrieval process is to identify the subset $S \subset \Omega$ containing all the images that the user is searching for. The retrieval process identifies the subset S in a probabilistic manner, by estimating the probability of relevance of every image in the collection as a conditional probability given the relevance feedback events accumulated from the user.

The retrieval process is iterative. The user starts a searching session having a target image in mind, as for example “birds on water”. The system is simply triggered and the first iteration

$t = 0$ starts as in Figure 3.1. The system initializes $p_0(k) = 0.5$ for all $k \in \Omega$, and then selects the first display set D_0. The user must choose the closest image in her opinion to what she is searching for, in our case "birds on water". The choice is subjective, and the user has to choose one and only one image as the positive image example. This is the event $B_0 = \{D_0,\ x_0^*\}$ for the system, the first relevance feedback event. With this information, the conditional probabilities are re-estimated for all images in the collection. Then, the system generates another display set D_1, and then it waits for the next iteration relevance feedback. The user chooses the closest image in D_1, and this new relevance feedback event is accumulated in B_1. Probabilities are re-estimated now based on B_1. And this process continues on and on as in Figure 3.2. The idea is that the system converges iteratively towards what the user is searching for, and iteratively displays more and more images relevant to the user. The notation convention is that the iteration $t+1$ starts immediately after the user feedback $\{D_t,\ x_t^*\}$.

Internally, the retrieval framework embodies an iterative relevance feedback mechanism that has two main components. First, there is a Bayesian model that estimates the probabilities of relevance of the images in the collection as conditional probabilities depending on the relevance feedback events. Second, there is a strategy to select what images to show next given the estimates of the probabilities of relevance of all the images in the collection.

3.2 Posterior probabilities of relevance

The probabilistic process assumes that the user knows S, and thus she can decide without doubt if an image belongs to S or not. Thus, for any image $k \in \Omega$ there are two distinct possibilities, $k \in S$ or $k \notin S$, and this can be interpreted as a binary event. Naturally, the system does not know S, and treats it as a random variable.

Relevance feedback events are accumulated iteratively as shown in Figure 3.2. After the system displays a set of images $D_t \subset \Omega$, $\|D_t\| = 8$, the user chooses one single image $x_t^* \in D_t$ that she considers to be the closest to S (i.e. the set of images that she is searching for), and this event is denoted as $\{D_t,\ x_t^*\}$. The cumulative event up to iteration t can be expressed as:

$$B_t = \cap_{i=0}^{t}\{D_i,\ x_i^*\}\ \ \forall t \geq 0. \tag{3.1}$$

The conditional probabilities $p_{t+1}(k) = P(k \in S \mid B_t)$ are estimated after each relevance feedback event. Initially, when there is no relevance feedback yet, the probabilities $p_0(k)$ are initialized with 0.5 for all $k \in \Omega$. Subsequently, the conditional probabilities are estimated via an image similarity model defined over the metric space of the indexing features. Before we return to this issue in §3.4, we further elaborate the Bayesian modeling.

Assuming that the events $\{D_t,\ x_t^*\}$ are conditionally independent from each other given the

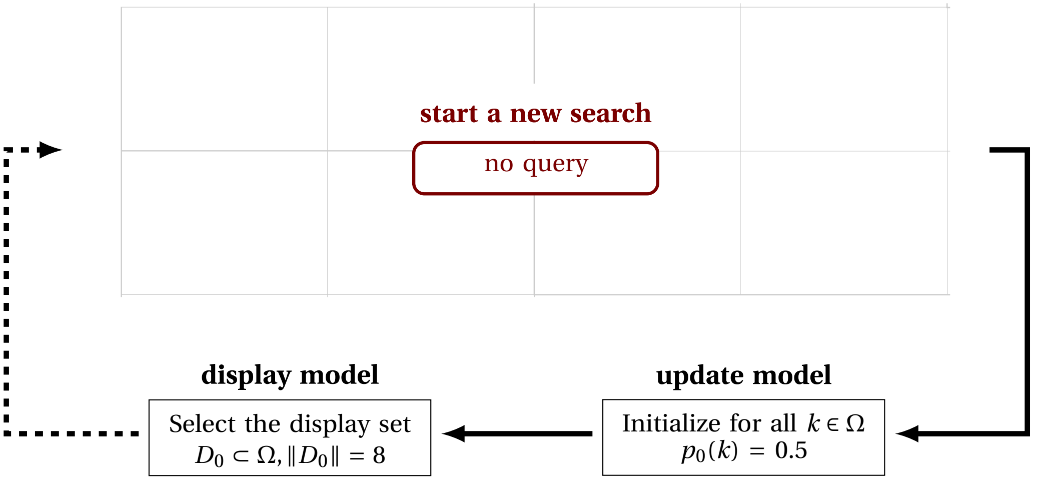


Figure 3.1: Starting a new searching session. The user does not submit any query, neither keywords nor image-examples. After a simple trigger, the first iteration $t = 0$ starts. The system initializes $p_0(k) = 0.5$ for all $k \in \Omega$, and then selects the first display set D_0.

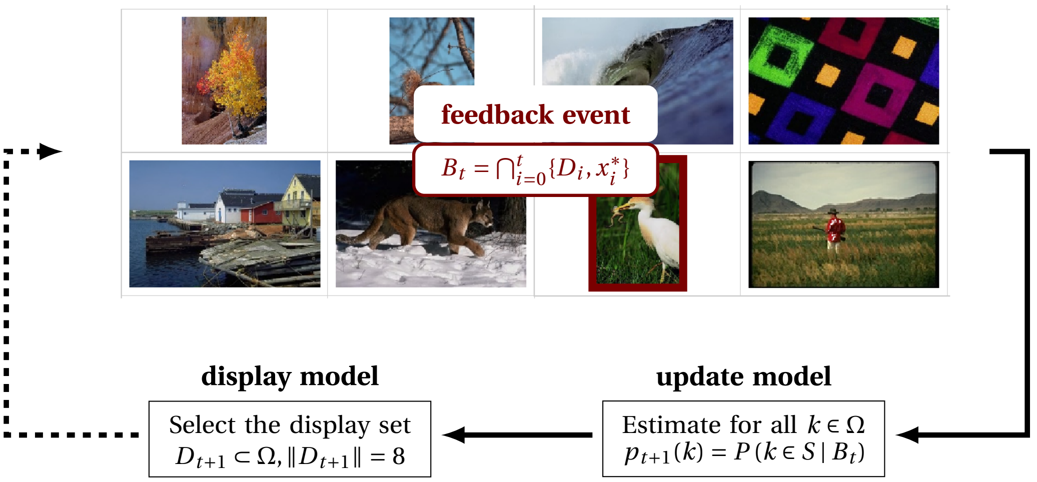

feedback event
$B_t = \bigcap_{i=0}^{t} \{D_i, x_i^*\}$
display model
Select the display set
$D_{t+1} \subset \Omega, \|D_{t+1}\| = 8$
update model
Estimate for all $k \in \Omega$
$p_{t+1}(k) = P(k \in S \mid B_t)$

Figure 3.2: Relevance feedback loop. At iteration t the system displays D_t. The next iteration $t+1$ is triggered by the relevance feedback event $\{D_t,\ x_t^*\}$. The system will update $p_{t+1}(k)$ for all $k \in \Omega$, and then will select the new display set D_{t+1}.

Table 3.1: Notation

Ω	image collection, where the images are identified by their indexes $\{1, 2, \dots\ k, \dots\}$
$S \subset \Omega$	set of images that the user is searching for
$D_t \subset \Omega$	set of images shown to the user at iteration t
$x_t^* \in D_t$	image chosen by the user at iteration t
$\{D_t,\ x_t^*\}$	relevance feedback event at iteration t
$p_t(k)$	probability of relevance of image k at iteration t

retrieval objectives, and using Bayes theorem, $p_t(k)$ can be expressed recursively:

$$p_{t+1}(k) = \frac{p_t(k) \cdot P_t^+(k)}{p_t(k) \cdot P_t^+(k) + (1 - p_t(k)) \cdot P_t^-(k)}, \tag{3.2}$$

where

$$P_t^+(k) = P\big(\{D_t,\, x_t^*\} \mid k \in S\big), \tag{3.3}$$

$$P_t^-(k) = P\big(\{D_t,\, x_t^*\} \mid k \notin S\big). \tag{3.4}$$

One may observe that the probabilities in Equations (3.3-3.4) should model as much as possible the user similarity judgments, and the better the model, the more reliable the relevance feedback. We shall return to this issue in §3.4.

3.3 Selection of the displayed images

A sensible technique to select what images to display next in D_{t+1} is to use an estimate of the marginal conditional probabilities of relevance $p_{t+1}(k) = P(k \in S \mid B_t)$. Instead of simply selecting the images with the highest probabilities of relevance, the selection technique should sample the image collection with the purpose of maximizing the efficiency of the relevance feedback events. The displayed images should at the same time concentrate on the relevant images and maintain some exploratory sampling among the non relevant images.

Ideally, each next display set D_{t+1} should maximize the flow of information from the user to the system, and therefore should minimize the uncertainty about S given the relevance feedback history B_t and the new evidence x_{t+1}^* that would be provided on D_{t+1} itself. This optimization problem is intractable because it implies looping over all subsets of size 8 in Ω:

$$D_{t+1} = \operatorname*{argmin}_{D \in \Omega} H\big(S \mid B_t,\, \{D, x^*\}\big). \tag{3.5}$$

Using the properties of conditional entropy, the entropy in Equation (3.5) can be re-written as:

$$\begin{aligned} H(S \mid B_t,\, \{D, x^*\}) &= H(\{D, x^*\}, S \mid B_t) - H(\{D, x^*\} \mid B_t) \\ &= H(\{D, x^*\} \mid S,\, B_t) + H(S \mid B_t) - H(\{D, x^*\} \mid B_t). \end{aligned} \tag{3.6}$$

Now in Equation (3.6), if one considers that the user knows S and knows to answer accordingly, there is no uncertainty in the first term. Also, one can observe that the second term does not depend on D:

$$H\big(\{D, x^*\} \mid S,\, B_t\big) = 0. \tag{3.7}$$

As a result, it follows that the optimal display set D_{t+1} is the one for which $H(\{D,x^*\} \mid B_t)$ is maximized. Since the entropy is maximized at the uniform distribution, the optimal display set D_{t+1} should include approximatively equally-likely images. This is equivalent to say that the Voronoi partitioning based on the images in D_{t+1} and on the metric d should have cells of equal mass under the appropriate distribution over Ω. This distribution is inaccessible since it involves the posterior over all subsets of Ω, but can be replaced by the distribution of p_t:

$$P\left(\{D,x^*\} \mid B_t\right) \approx \frac{1}{\|D\|}. \tag{3.8}$$

In our retrieval system, the displayed images, namely D_t with $\|D_t\| = 8$, are generated via a Voronoi tessellation algorithm proposed by Fang and Geman [21] that approximates the ideal but intractable Voronoi partitioning. The algorithm selects the images $x \in D_t$ by growing subsequent Voronoi cells based on the image similarity distances and their current probabilities of relevance. The optimum probability mass of each Voronoi cell would be an exact fraction m_t of the total probability mass:

$$m_t = \frac{1}{\|D_t\|} \cdot \sum_{k\in\Omega} p_t(k). \tag{3.9}$$

The first selected image is the image with the highest probability in the entire collection Ω:

$$x^{(0)} = \underset{k\in\Omega}{\operatorname{argmax}}\ p_t(k), \tag{3.10}$$

and the Voronoi cell $\mathscr{C}^{(0)}$ is grown by including images one by one, as ordered by their similarity distances to x_0 in increasing order, until its probability mass reaches the optimum. The second image is selected among the images from outside the first Voronoi cell:

$$x^{(1)} = \underset{k\in\Omega\setminus\mathscr{C}^{(0)}}{\operatorname{argmax}}\ p_t(k), \tag{3.11}$$

and the Voronoi cell $\mathscr{C}^{(1)}$ is grown by including images in a similar manner. The algorithm loop continues until the set of images D_t is complete.

The first display set D_0 is generated by running the algorithm with the initial probabilities of relevance, $p_0(k) = 0.5$ for all $k \in \Omega$. The algorithm still grows the Voronoi cells but chooses the images randomly between the equally probable candidates.

Table 3.2 formalizes the procedures to select the set D_t to be displayed next. Given a target mass m, the procedure **ComputeDisplaySet** picks each image successively, each time selecting the one with the highest p_t which does not belong to the neighborhoods of mass m centered on the images already selected. In the function **ComputeCells**, the neighborhoods are grown in parallel by including images one by one, as ordered by their similarity distances, until the probability mass of each neighborhood reaches the target mass m.

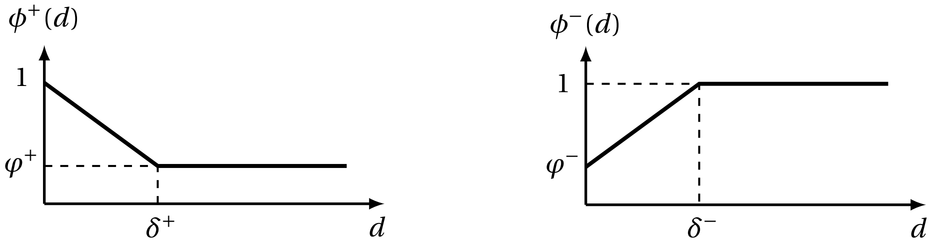


Figure 3.3: Calibration functions. δ^+, δ^- are the thresholds that normalize the distances, and φ^+ and φ^- are the attenuations that compensate for the partial mismatch between the distances and the user perception of image similarities (i.e. semantic gap) as explained in [24].

3.4 Similarity metric

The probabilities $P_t^+(k)$ and $P_t^-(k)$ in Equations (3.3-3.4) are modeled based on a similarity metric defined over the image feature space as in [25], which puts higher probability on the images similar to the chosen ones and accounts for an effect of "saturation" that ignores the increase in the image dissimilarities beyond a certain threshold:

$$P_t^+(k) = \frac{\phi^+(d(k, x_t^*))}{\sum_{x \in D_t} \phi^+(d(k, x))}, \tag{3.12}$$

$$P_t^-(k) = \frac{\phi^-(d(k, x_t^*))}{\sum_{x \in D_t} \phi^-(d(k, x))}. \tag{3.13}$$

Table 3.2: Procedures to compute a meaningful display set D_t. Given the current estimate of probabilities $\mathbf{p} = \{p_t(k)\ \forall k \in \Omega\}$, the cardinality $\|D_t\| = Q$, and a target mass m, the function **ComputeDisplaySet** returns a list of images $x^{(1)}, \ldots, x^{(Q)}$ such that each of them has a high individual p_t, and they have disjoint neighborhoods $\mathscr{C}^{(1)}, \ldots, \mathscr{C}^{(Q)}$ of mass m. Given the probabilities $\mathbf{p}$, a list of images and a mass m, the function **ComputeCells** returns the corresponding disjoint neighborhoods, all of the same mass m.

```
Function ComputeDisplaySet($\mathbf{p}, Q, m$)
for $q = 1, \ldots, Q$ do
    $\mathscr{C}^{(1)}, \ldots, \mathscr{C}^{(q-1)} \leftarrow$ ComputeCells($\mathbf{p}, x^{(1)}, \ldots, x^{(q-1)}, m$)
    $x^{(q)} \leftarrow \operatorname*{argmax}_{k \in \Omega \setminus \cup_{i=1}^{q-1} \mathscr{C}^{(i)}} p(k)$
end for
return $x^{(1)}, \ldots, x^{(Q)}$

Function ComputeCells($\mathbf{p}, x^{(1)}, \ldots, x^{(i)}, m$)
return $\mathscr{C}^{(1)}, \ldots, \mathscr{C}^{(i)}$
    s.t. $\forall q \ \sum_{k \in \mathscr{C}^{(q)}} p(k) = m$
    and $\forall q,\ r \neq q,\ \forall k \in \mathscr{C}^{(q)} \ \|k - x^{(q)}\| \leq \|k - x^{(r)}\|$
```

The distance d between the images is the L^2 norm between the image feature vectors (i.e. indexing information):

$$d(k,h) = \sqrt{\sum_{f=1}^{F} (k_f - h_f)^2}, \tag{3.14}$$

where F is the dimensionality of the image feature space. As we explain in §3.5, our experiments use bags-of-words based on SIFT, but any other indexing feature vectors will do.

ϕ^+ and ϕ^- are calibration functions designed to capture the user perception of image similarities and error-prone decision-making behavior. We consider calibration functions of parametric forms as shown in Figure 3.3 and the general idea is that δ^+, δ^- are thresholds beyond which the L^2 norm fails to resemble the user perception, and φ^+ and φ^- are attenuations that compensate for the partial mismatch between the distances and the user perception of image similarities (i.e. the semantic gap).

In [24], the calibration functions aim to maximize the likelihood of the user answers under the probabilistic model of the framework. In [25], the calibration problem is re-formulated starting from the psychological interpretation of the parameters. The idea is to learn the δ and φ parameters in Figure 3.3 via a statistical technique that requires an image labeling task. During the labeling session, the user input is collected in the same manner as a searching session, with the key difference that the target S is communicated visually. As during a searching session, the user is supposed to choose among the displayed images the image that is the closest to the target in her opinion. In this way, the labeling session collects data triplets of the form $(S_i,\, D_i,\, x_i^*)$ that can be used to formulate a maximum likelihood technique:

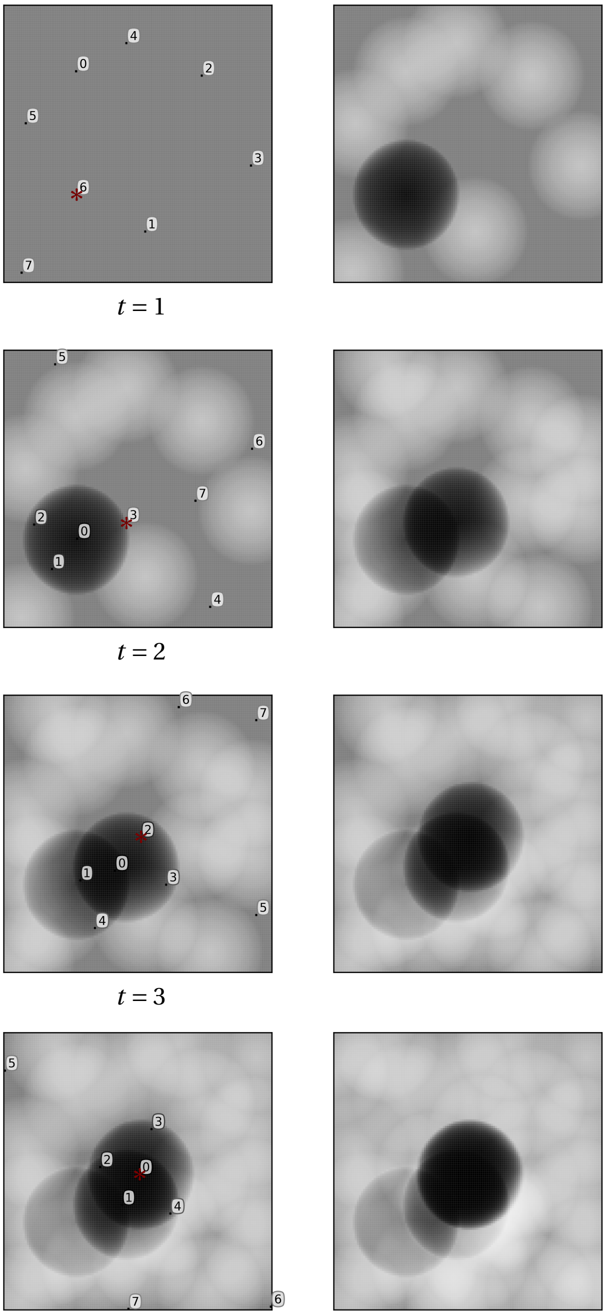
$$L^+(\delta^+,\, \varphi^+) = \prod_i P_i^+(S_i,\, D_i,\, x_i^*) = \prod_i \frac{\phi^+(d(x_i^*,\, S_i))}{\sum_{x\in D_i} \phi^+(d(x,\, S_i))}, \tag{3.15}$$

$$L^-(\delta^-,\, \varphi^-) = \prod_i P_i(S_i,\, D_i,\, x_i^*) = \prod_i \frac{\phi^-(d(x_i^*,\, \Omega\backslash S_i))}{\sum_{x\in D_i} \phi^-(d(x,\, \Omega\backslash S_i))}. \tag{3.16}$$

3.5 Experimental results

We developed the retrieval system as a web-application which has the advantage of permanent availability for demos and evaluations. We distributed the application software under the AGPL Version 3 open-source license in order to give transparency to our work, and to facilitate the reproducibility of our experiments. The web-application is further described in §A.

In our experiments and evaluations, we set up our retrieval system for three image collections, namely the Corel stock photo library, the ImageNet dataset and the synthetic collection generated by ourselves. These collections and their characteristics are further described in §C.



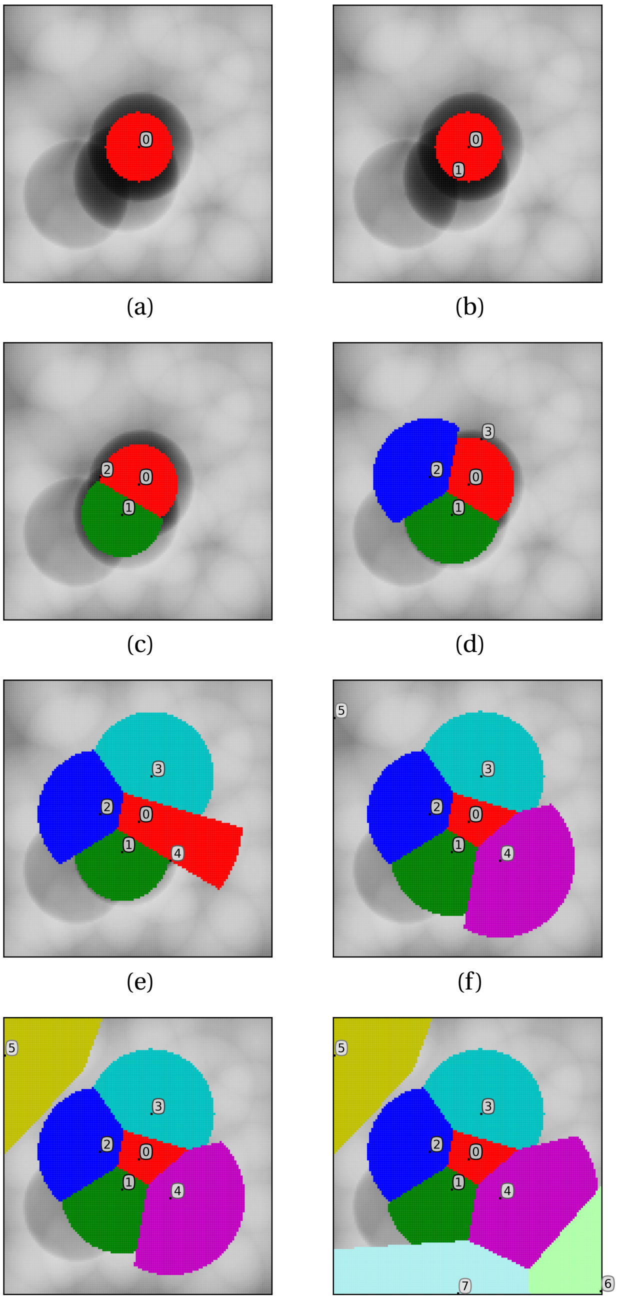
$t = 1$

$t = 2$

$t = 3$

$t = 4$

Figure 3.4: The posterior probabilities $p_t(k)$ for all $k \in \Omega$ are updated iteratively. Here, the relevance feedback events are given by a user who is searching for images with points close to the center. One can see how the distribution of probabilities evolves towards matching the user retrieval objective.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 3.5: The set of displayed images is generated via the Voronoi tessellation algorithm. To illustrate its intermediate steps, the images already selected are marked in black and their current Voronoi cells are indicated by colors. (a): The first image $x^{(0)}$ is selected, and the first Voronoi cell $\mathscr{C}^{(0)}$ is grown. (b): The second image $x^{(1)}$ is selected. (c): The Voronoi cells $\mathscr{C}^{(0)}$ and $\mathscr{C}^{(1)}$ are grown in parallel. $\mathscr{C}^{(0)}$ is shrunken by detaching the images closer to $x^{(1)}$, and then re-grown by including other images that are still closer to $x^{(0)}$. (d-h): The algorithm proceeds in the same manner until the set of displayed images is complete.

During our research, we did various experiments, and carried out plenty of user-based evaluation campaigns. Here, we introduce our experimental setups and our evaluations that we have been conducted in order to analyze and understand the system behavior and to validate systematically our contributions.

3.5.1 Intuitive analysis

For an intuitive illustration of the system behavior, we created a synthetic image collection where each image has two indexing features in the $[0,1]$ interval. These features are interpreted as coordinates in the 2D Cartesian space, and are used in a dual manner in order to define the image visual content and then to position the image in the abstract representation of the entire collection. On the one hand, the features define the image visual content, which is a single point positioned accordingly, and thus there is no gap between the indexing features and the semantic meaning. On the one hand, they define also the position of the image itself in the landscape of the entire collection, and thus it results a nice abstract representation. In Appendix §C.3, we explain in detail this synthetic collection, and its abstract representation.

Figure 3.4 shows how the probabilities of relevance are gradually updated on successive iterations. We can see how the system is calibrated in such a way that images closer to the chosen image get higher probabilities and images closer to the other displayed images get lower probabilities. The images far from any of the displayed images keep their probabilities unchanged.

Figure 3.5 shows the intermediate steps of the Voronoi tessellation algorithm. One can see how the Voronoi cells are grown, and how the images to be displayed are selected. Intuitively, the cells including regions with higher probabilities are smaller than the cells including regions with lower probabilities. In this way, the system concentrates on regions with high probabilities while still insists on sampling the entire collection.

The abstract 2D representation provides important clues about the system behavior, although it does not resemble entirely the real case of multi-dimensional image features. On the one hand, the Voronoi clusters grow more naturally in multi-directions and they are not that hindered by one another as in the 2D case. On the other hand, there is the “spherical” effect of the Euclidean high-dimensional space (i.e. the image feature vectors are all apart from each other) and the image similarity distances may not be that discriminative as in the 2D case.

3.5.2 User-based evaluation

The only realistic way to evaluate the performance of a retrieval system is to ask users who bring in the challenge of subjective perception of image similarities. Therefore, we organized several user-based evaluation campaigns in order to analyze our contributions. Here, we present the details of our principal evaluation scenario that we used in all our subsequent campaigns with very small variations.

Each of our evaluations has been conducted with groups of 20 users or more who were not familiar with the system and thus not tempted to favor the evaluation. The evaluations do not rely on any *a priori* defined ground truth. Instead, they rely on comparing different system configurations. Each user has to perform several searching sessions in which they are asked to search for some given semantic categories, and they are asked to end the searching sessions as soon as they are satisfied by one of the displayed images. The retrieval performance is the cumulative percentage of successful sessions per number of iterations.

The evaluations rely on comparing different system configurations, and usually each evaluation includes at least three configurations:

- our proposed system, or systems if we test multiple variants
- the original baseline system
- a random system that displays images randomly without replacement. The random system discards totally the relevance feedback and the similarity metric, and thus provides the lowest acceptable performance.

In order to ensure a sufficiently reliable diversity, there were 6 semantic categories described only in words:

- domestic dogs in close-up portrait
- electronic devices as TV, radio, mobile
- big boats as ferryboats, cargoes
- exotic fruits in close up portrait
- furniture items as cupboards, tables, chairs
- public buildings as shops, malls

In order to ensure comparable difficulty, these categories were chosen to be relevant for about 1–2% of the image collections based on the available ground-truth information (e.g. the cardinality and the associated keywords of the ImageNet categories). Here, we should mention that the ground-truth information (e.g. synsets or other keywords) was considered only as benchmark meta-data for assessing the retrieval difficulty, and our retrieval systems do not make use of it.

The interpretation of the semantic categories in the sense of visual content was left to the user. The users were only told to end the searching sessions when they were satisfied first by one of the displayed images. In order to avoid any bias, the searching sessions were presented in a random fashion. The system configurations and the semantic categories were randomized all together in one single user test. The users were not aware of which configuration was active in a certain session. In fact, they were not introduced to anything beyond the evaluation interface in Figure A.3.

In order to ensure a minimum variance in the overall scenario, all our evaluations have been set up in such a way that each user was assigned to perform one searching session for each combination of all the system configurations and the semantic targets included in the evaluation. Thus, each user was searching for the same semantic category with all system configurations, in an anonymized fashion, and we used this symmetry in our binomial tests for statistical significance.

Having 6 semantic categories and at least 20 users, each evaluation provided us with 120 searching sessions or more for each system configuration, which we considered to be sufficient for our preliminary research investigations and our limited resources.

$$6 \text{ semantic categories} \times 20 \text{ users} = 120 \text{ searching sessions / system configuration}$$

Again, here we presented our principal evaluation scenario. Later on in the next chapters, we will address the scenario variations of each evaluation if there were any.

3.5.3 Automatic tests

We have developed a test platform for running the web-application programmatically, without human interaction. This platform implements an automatic user that interacts with the web-application in exactly the same way a human user does. This platform is invaluable in getting confidence before organizing the time-costly user-based evaluations. In §B, we describe further how this test platform can be used for abstract performance evaluations that resemble the user-based evaluations at a certain extent.

Our first use of the automatic tests was to investigate how the retrieval performance depends on the image similarity model. We observed that the retrieval performance of the automatic user is maximized when the parameters δ^+ and δ^- are adjusted to saturate only after including on average 10% of the images, which is approximatively close to 1/8 of the collection. The parameters φ^+ and φ^- do not play an important role for the automatic user that gives always ideal relevance feedback, and they are more meaningful for human users. Therefore, we were taken over the same optimum values as derived in [24], namely 0.06 and 0.29.

3.6 Research issues

The theoretical study, the intuitive analysis and some preliminary user-based experiments, all of these helped us to identify the limitations of the retrieval framework and to prioritize a few promising research issues. Here, we anticipate the research issues that will be addressed in the next chapters of this thesis.

We observe that the retrieval framework requires a computational effort that is tightly related to the size of the image collection Ω. On the one hand, the probabilities of relevance are computed for all the images in the collection, and this implies $\mathcal{O}(\|\Omega\|)$ complexity. On the other hand, the Voronoi tessellation algorithm involves sorting operations of $\mathcal{O}(\|\Omega\| \cdot \log \|\Omega\|)$ complexity over the entire collection.

We can identify two different retrieval regimes: understanding broadly the categories of interest to the user, and refining the search in this or these categories to converge to specific images among them. As argued in [24, 25], the retrieval framework is well suited for understanding broadly the image categories, but other retrieval techniques should be employed to retrieve specific images among these identified categories.

A key factor of the retrieval framework, as in case of any other relevance feedback framework, is the similarity measure between the images that governs the retrieval process. The calibration technique elaborated in [25] aims to compensate for the semantic gap between the similarity metric and the user perception of image similarity, but it requires an image labeling task conducted separately from the searching sessions.

More and more multimedia collections tend to include inter-related modalities, as for example photos and annotations in photo-sharing repositories, pictures and captions in news web-sites or x-ray scans and reports in medical databases. Intuitively, these inter-relationships could be exploited in order to compensate the weaknesses of the individual modalities and to provide better indexing features [15]. In line with this trend, we observe that the current approach uses a rigid similarity metric based on low-level features extracted from the visual content of images (i.e. global descriptors of color, texture and shape).

3.7 Summary

In this chapter, we have presented the interactive query-free retrieval framework that is central to our work. The retrieval process is based solely on iterative relevance feedback. At each iteration, the user chooses one single image as positive relevance feedback. The system updates the probabilities of relevance for all the images in the collection, and then selects the images to be displayed next.

The probabilities are updated based on a Bayesian model that requires an image similarity metric that is calibrated in order to minimize the semantic gap. The images to be displayed next are selected via a Voronoi tessellation algorithm that aims to maximize the flow of information from the user to the system.

Along the summary of the original approach, we motivated the research directions we have chosen to pursue further on. Using a synthetic image collection, we provided the first intuitive analyses of the system components, and we opened the discussion about the limitations of the original approach.

4 Large-scale HEAT framework

Scalability is a critical issue in designing retrieval systems, and is one of the first research directions that we investigated. It has been shown repeatedly that iterative relevance feedback is a very efficient solution for content-based image retrieval. However, no existing system scales gracefully to hundreds of thousands or millions of images.

In this chapter, we formalize a new framework dubbed Hierarchical and Expandable Adaptive Trace (HEAT) that scales up to millions of images. Our approach modulates on-the-fly the resolution of the interactive search in different parts of the image collection, by relying on a hierarchical organization of the images computed off-line. Internally, the strategy is to maintain an accurate approximation of the probabilities of relevance of the individual images while fixing an upper bound on the required computation.

Our proposed framework effectively decouples the computational effort from the size of the collection, and still preserves the retrieval capabilities. Our system is compared on the ImageNet dataset to the state-of-the-art approach it extends, by conducting user evaluations on a sub-collection of 33,000 images. Its scalability is then demonstrated by conducting equivalent evaluations on 1,000,000 images.

4.1 Introduction

Modern digital technologies produce large amounts of photos, and some of the largest collections containing billions of images are Flickr, FaceBook, and of course the World Wide Web as a whole. Arguably, the most critical issue in designing retrieval systems is their scalability and ability to accommodate the growing amount of data.

We started to investigate the scalability potential of the retrieval framework by looking at the possibilities to reduce the storage capacity as well as the computational effort. The main goal was to find a novel indexing and updating strategy that can handle collections of 1,000,000 images or more, one order of magnitude larger than the original strategy.

As we saw in Chapter §3, at the core of our retrieval framework there are two components. First, there is the model to compute the probability for an image to be relevant to the user given what images have been shown to her until now and what she has chosen. Second, there is the strategy to select what images to show her given the estimates of the probabilities of relevance of all the images in the collection. In the original approach, these two components require a computational effort that grows quasi-linearly with the size of the collection. Since these two components are involved in the on-line interaction with the user, the original approach can not be practically recommended for collections much larger than about 60,000 images.

The novel approach we propose computes a hierarchical organization of the images off-line. At each iteration of the on-line retrieval process, it selects a "trace" in this hierarchy that corresponds to a partition with a fine resolution in the parts that are rich in relevant images and a coarse resolution in the parts that are clearly discarded by the model. In the iterative process, the trace is dynamically refined by expanding (i.e. some nodes are replaced by their children) and collapsing (i.e. some nodes are replaced by their parent) operations.

Our proposed framework effectively decouples the computational effort from the size of the collection, and still preserves the retrieval capabilities. Experiments show that the required size of the trace for maintaining the same retrieval performance is very modest when compared to the total number of images in the collection. Moreover, one can control explicitly the trade-off between the computational effort and the retrieval performance by bounding the cardinality of the trace.

4.2 State-of-the-art

Relevance feedback is indeed envisioned by many researchers as the only alternative that could cope properly with the challenges in image retrieval, and multimedia retrieval in general [51, 69, 14]. Whereas relevance feedback is a very efficient solution for content-based image retrieval, no existing system scales gracefully to hundreds of thousands or millions of images [51, 69, 14, 15]. Moreover, the relevance feedback is traditionally seen as a post-retrieval mechanism for refining the retrieved results of an initial query formulated explicitly.

The original approach presented in Chapter §3 requires a computational effort that is tightly related to the size of the image collection Ω. On the one hand, the probabilities of relevance are computed for all the images in the collection. Although the computational cost of the probability model is very light in itself, it requires access to the similarity distances from all the images in the collection to each of the displayed images, and this implies either storage capacity of $\mathcal{O}(\|\Omega\|^2)$ complexity off-line or computational effort of $\mathcal{O}(\|\Omega\|)$ on-the-fly. On the other hand, the Voronoi tessellation algorithm involves sorting operations of $\mathcal{O}(\|\Omega\| \cdot \log \|\Omega\|)$ complexity over the entire collection.

Our idea is to support an approximation of the relevance feedback mechanism that uses the Bayesian framework on top of a hierarchical tree-like organization of the image collection.

Although hierarchical trees have been extensively used for zoomable user interfaces as PhotoMesa [4] and many other browsing solutions [30], to the best of our knowledge there is no system that uses such a concept in order to scale up relevance feedback mechanisms.

The closest research we found related to our idea of a dynamically adaptive and traceable cut within a hierarchical tree-like organization is in the field of information visualization and visual data mining, where it is referred to as a tree map [55] and other equivalent terms like fish-eye [1], or tree view [5].

Apparently, it is well accepted by the research community that the advantages of such hierarchically structured organizations break down in the face of the high-dimensional image feature spaces that are typically seen in content-based retrieval. However, in comparison with other relevance feedback mechanisms, the work of Ferecatu and Geman [24, 25] has the specificity of dealing explicitly with the miss-alignment between the image feature space and the user subjective perception of image similarities. This fact encouraged us to look again into this research direction.

4.3 Scalable system

While maintaining all the core operations basically the same, our approach manages to compute the probabilities of relevance of only a small set of representative images. The probabilities of relevance of all the other images in the collection are approximated from these ones. This is achieved by first organizing the image collection as a pre-computed hierarchical tree based on the image similarity distances, and then updating during the retrieval process a partitioning of the image collection according to the estimated probabilities.

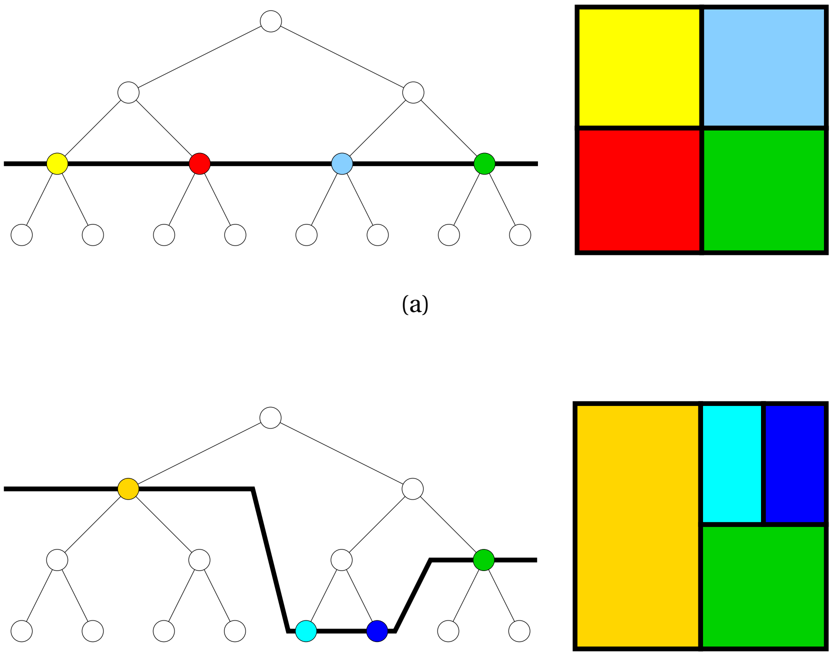
4.3.1 Tree and trace

The image collection Ω is organized in a hierarchical tree $\mathcal{N}$ as sketched in the left side of Figure 4.1. Formally, each node $N \in \mathcal{N}$ has a set of children denoted as $C(N) \subset \mathcal{N}$. Furthermore, each node N is associated with a set of images denoted as $\Omega(N) \subset \Omega$. Each leaf node is associated with one single image, thus if N is a leaf node, then $C(N) = \emptyset$ and $\|\Omega(N)\| = 1$. These sets of images are hierarchically disjunctive and they naturally respect the properties:

$$\forall M,\ M' \in C(N),\ M \neq M' \ \Rightarrow\ \Omega(M) \cap \Omega(M') = \emptyset, \tag{4.1}$$

$$\bigcup_{M \in C(N)} \Omega(M) = \Omega(N). \tag{4.2}$$

Additionally, each node $N \in \mathcal{N}$ has a representative image k^*_N that is the closest image to the center of $\Omega(N)$ in the image feature space.



(a)

(b)

Figure 4.1: Relation between the hierarchical tree and the trace adaptive partitioning. The graph depicted on the left stands for the tree $\mathcal{N}$, and the square on the right stands for the full image collection Ω. Intuitively, each node $N \in \mathcal{N}$ is associated with a subset of images $\Omega(N)$. The thick black lines running through the trees show two different traces $\mathcal{T}$. The colored rectangles show the resulting partitions of the collection, as each rectangle stands for the $\Omega(N)$ associated to the node N of same color. The trace in (a) stays at the same depth, resulting in a homogeneous partitioning. The trace in (b) goes shallower in one part of the collection and deeper in the other part, resulting in a partitioning with varying resolution.

Table 4.1: Notation

$\mathcal{N}$	complete set of nodes of the hierarchical tree
$\mathcal{T}_t$	trace at iteration t
$C(N)$	children nodes of node N
$\Omega(N)$	set of images associated with node N
k^*_N	representative image of node N
$p_t(k^*_N)$	probability of the representative image of node N that approximates $p_t(k)$ for all $k \in \Omega(N)$
$q(N)$	probability mass of node N

A trace $\mathcal{T} \subset \mathcal{N}$ is any set of nodes that stands for a complete and disjunctive partitioning of the image collection that respect the properties:

$$\forall A, B \in \mathcal{T},\ A \neq B \ \Rightarrow\ \Omega(A) \cap \Omega(B) = \emptyset, \tag{4.3}$$

$$\bigcup_{A \in \mathcal{T}} \Omega(A) = \Omega. \tag{4.4}$$

These properties guarantee that any image in the collection is associated to one and only one node in any trace. Therefore, if $N \in \mathcal{T}$ is a node included in the trace, it can be used without ambiguity to represent all its associated images $\Omega(N)$ as explained by the sketch in Figure 4.1.

4.3.2 Approximation of p_t

The computational effort is controlled in our approach by estimating the probabilities of relevance only for the representative images of the nodes that are part of the current trace. From this bounded set of probabilities, we both infer a sound approximation of the Voronoi tessellation algorithm described in § 3.3, and optimize the resolution of the trace as presented next in § 4.3.3.

For any node $N \in \mathcal{T}$, the probabilities of relevance of all the individual images in $\Omega(N)$ are approximated by the probability of relevance of its representative image k_N^*.

At each iteration t, the conditional probabilities $p_t(k_N^*)$ are computed from scratch based on the full history of relevance feedback events B_{t-1} as shown in § 3.2. They are not approximated in any way, and thus they are as if the node N would have been part of the trace since the beginning of the retrieval process.

Furthermore, the prerequisites of the Voronoi tessellation algorithm described in § 3.3 are reconsidered as follows. The probability mass of a node N is approximated as:

$$q(N) = \sum_{k \in \Omega(N)} p_t(k) \approx p_t(k_N^*) \cdot \|\Omega(N)\|. \tag{4.5}$$

The probability mass of the entire collection is approximated as:

$$q^{all} = \sum_{k \in \Omega} p_t(k) \approx \sum_{N \in \mathcal{T}} q(N). \tag{4.6}$$

The optimum probability mass of the Voronoi cells is approximated as:

$$q^{opt} = \frac{1}{\|D_t\|} \cdot q^{all} \approx \frac{1}{\|D_t\|} \cdot \sum_{N \in \mathcal{T}} q(N). \tag{4.7}$$

When a node N is expanded, its probability mass $q(N)$ is substituted by the probability masses of its children, and this results in a finer approximation:

$$q(N) = \sum_{M \in C(N)} q(M) \approx \sum_{M \in C(N)} p_t(k_M^*) \cdot \|\Omega(M)\|. \tag{4.8}$$

When the nodes in $C(N)$ are collapsed, the sum of their probability masses is substituted by the probability mass of their parent, and this results in a coarser approximation:

$$\sum_{M \in C(N)} q(M) = q(N) \approx p_t(k_N^*) \cdot \|\Omega(N)\|. \tag{4.9}$$

Based on these approximations, the Voronoi tessellation algorithm is now performed at the granularity level of the trace instead of the individual images. Therefore, the centers of the Voronoi tessellation are selected among the nodes in the current trace, and the displayed images are their corresponding representative images.

4.3.3 Trace refinement

The aim of the trace refinement is to optimize the approximation of the probabilities of relevance of the individual images under the constraint of preserving a bounded size of the trace. Intuitively, this is achieved when the variances of the probabilities within each node in the trace are small, or in other words when the probability of each image in the collection is approximated as well as possible by the probability of its corresponding representative image. The trace refinement consists of a collapsing operation followed immediately by an expansion operation.

Starting from the current trace, the collapsing operation book-keeps the sets of children that are completely included in the trace, and thus may be replaced by their parents. Recursively, one at a time, the set of children that minimizes the mean-variance cost function:

$$\underset{\forall N,\ C(N) \subset \mathcal{T}}{\operatorname{argmin}} \quad \mu(N) \cdot (\sigma^2(N) + \epsilon \cdot \|\Omega(N)\|) \tag{4.10}$$

is collapsed into its corresponding parent. The probability of relevance of the representative image $p_t(k_N^*)$ is computed from scratch as mentioned in § 4.3.2, and then is used for computing the subsequent mean-variance values. The recursive routine for collapsing nodes exits when the size of the trace reaches the minimum bound.

The probability mean and variance of each node are estimated based on its children:

$$\mu(N) = \frac{\sum_{M \in C(N)} p_t(k_M^*) \cdot \|\Omega(M)\|}{\sum_{M \in C(N)} \|\Omega(M)\|}, \tag{4.11}$$

$$\sigma^2(N) = \frac{\sum_{M\in C(N)} p_t^2(k_M^*)\cdot \|\Omega(M)\|}{\sum_{M\in C(N)} \|\Omega(M)\|} - \mu^2(N). \tag{4.12}$$

In Equation (4.10), ϵ introduces an infinitesimal preference toward collapsing the nodes with smaller cardinality when nodes with different cardinality have comparable mean-variance values. Thus, ϵ is not a sensitive parameter and was set to 10^{-6}, a value related to the size of the collection.

As soon as the collapsing operation exits, the expansion operation replaces all the nodes in the trace with their children and computes the probabilities of relevance of their representative images. This expansion operation could be seen as a sampling of the parent nodes that will be used in the subsequent trace refinement, at the next iteration, in order to identify the new nodes that should be further expanded or can be safely collapsed.

4.3.4 Algorithm integration

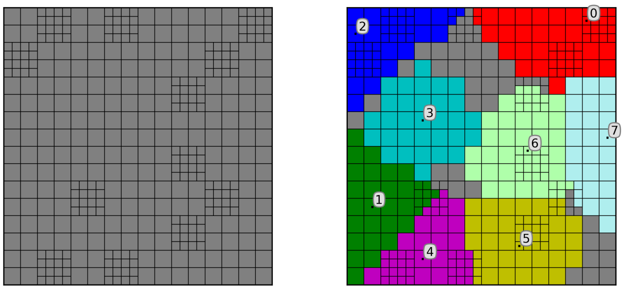
The skeleton of our proposed approach is as follows:

1. Update the probabilities of relevance $p_{t+1}(k_N^*)$ for $\forall N \in \mathcal{T}_t$ based on the previously computed $p_t(k_N^*)$ and according to the newly received relevance feedback event $\{D_t,\ x_t^*\}$.
2. Perform the trace refinement. The trace $\mathcal{T}_t$ is altered via the collapsing and expanding operations resulting in the new trace $\mathcal{T}_{t+1}$.
3. Update the probabilities of relevance $p_{t+1}(k_N^*)$ for $\forall N \in \mathcal{T}_{t+1}$ according to the full history of relevance feedback events $B_t = \cap_{i=0}^{t}\{D_i,\ x_i^*\}$.
4. Select the set of images D_{t+1} by performing the Voronoi tessellation algorithm on the current trace $\mathcal{T}_{t+1}$.
5. Display D_{t+1}. Wait for the relevance feedback event $\{D_{t+1},\ x_{t+1}^*\}$ to occur, and then proceed with the next iteration.

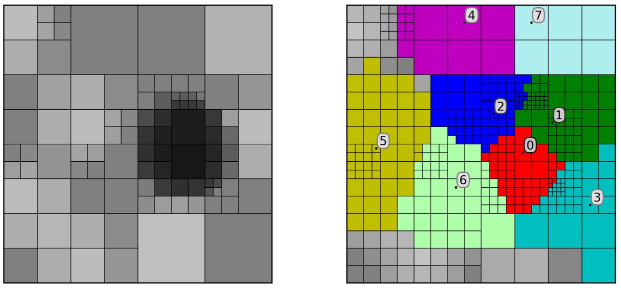
For an intuitive illustration of the system behavior, we set up the HEAT system for the synthetic collection described in §3, and we performed one session of searching for images with points close to the center, which are images close to the center due to the duality between images and points. Figure 4.2 shows how the trace evolves at each iteration, and how the image collection is sampled at different resolutions in different regions.

4.4 Experimental results

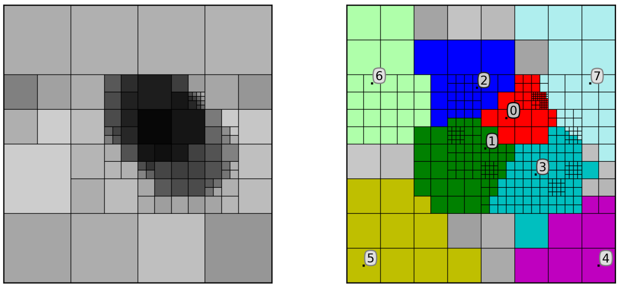
The aim of the following experiments was to evaluate our HEAT system in comparison with the original system in terms of both the retrieval performance and the computational effort.



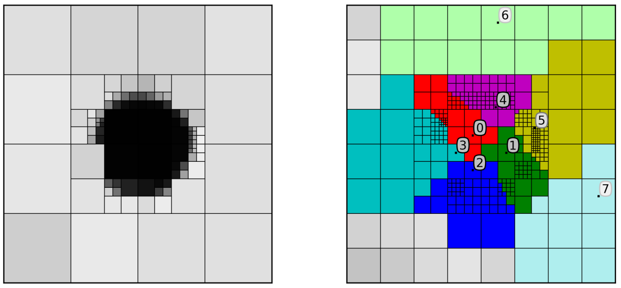
$t = 0$ (initial) $t = 0$ (expand)



$t = 1$ (collapse) $t = 1$ (expand)



$t = 2$ (collapse) $t = 2$ (expand)



$t = 5$ (collapse) $t = 5$ (expand)

Figure 4.2: Evolution of the trace for the synthetic collection, when searching for images with points close to the center. At iteration 0, the trace is initialized randomly. At each iteration, the current trace is collapsed and expanded, the probabilities of relevance are updated, and then the new images to be shown are selected. After 5 iterations, the trace concentrates mostly on the intended region.

Regarding the retrieval performance, we looked for evidence that our extension is capable of providing a retrieval performance comparable to the original one. Regarding the computational effort, we looked for evidence that our extension is capable of scaling up beyond two orders of magnitude.

The experiments were organized with two collections obtained from the ImageNet dataset [18] that has the convenience of being structured in 1,000 semantic categories, where each category has 500–2500 images. Further details about ImageNet are in §C.2. Considering the subset of 1,200,000 images provided with pre-computed SIFT features (Scale Invariant Feature Transform) [41], we obtained a large collection including about 1,054,000 images, namely all the images with valid url at that date. Then, we sampled a small collection of 33,000 images (i.e. 3% of the large collection) with the guarantee of being similarly and proportionally populated as the large collection.

4.4.1 System setup

The image similarity distances are defined simply as the Euclidean distance between the histogram-like feature vectors (i.e. the bags of visual words) of dimension 1,000, as they are provided by the ImageNet.

The relevance feedback framework is calibrated as described in [24], and the parameters of the probability positive and negative models are adjusted to saturate only after including on average 10% of the images in the collection.

The hierarchical tree is generated by applying a divisive top-down k-means algorithm. The tree is initialized with the root node as being the single node and representing all the images in the collection. Recursively, the images of each node are split in 8 k-means clusters. These resulting clusters are used to define the new nodes, one level deeper in the tree. Naturally, the former node becomes a parent node with the newly defined nodes as its children.

Considering the size of the collection, we employ an approximation of k-means that is studied in terms of clustering feasibility and computational complexity in [27]. The clustering of sets of more than 50,000 images is done in two phases. In the first phase, k-means is initialized randomly and then performed – until convergence – on a random sample of 50,000 images in order to obtain an estimate of the centroids. In the second phase, k-means is initialized with the estimated centroids and then performed – only 2 iterations – on the full set of images.

4.4.2 Evaluation scenario

The evaluation was conducted with 20 users not familiar with the system, and consisted of running user tests with three systems:

- our proposed HEAT system

- the original system
- a random system that displays images randomly without replacement

The evaluations follow the scenario in §3.5.2. In order to ensure a sufficiently reliable diversity and comparable difficulty, there were 6 semantic categories described only in words:

- domestic dogs in close-up portrait
- electronic devices as TV, radio, mobile
- big boats as ferryboats, cargoes
- exotic fruits in close up portrait
- furniture items as cupboards, tables, chairs
- public buildings as shops, malls

The interpretation of the semantic categories in the sense of visual content was left to the user. The users were only told to end the searching sessions when they were satisfied by one of the displayed images.

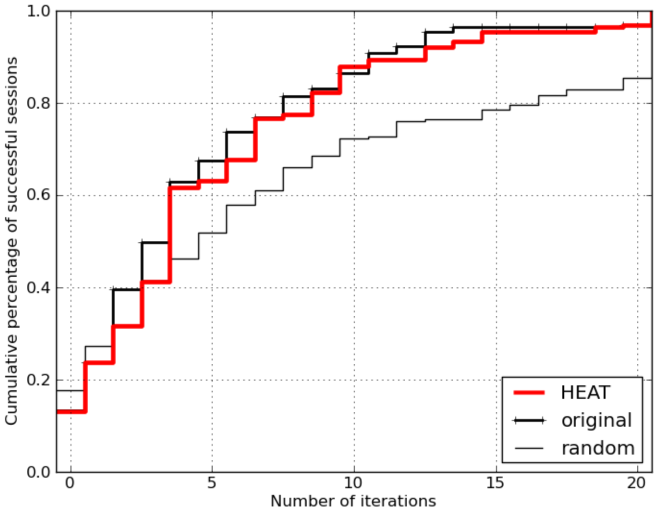
4.4.3 Performance impact

The experiments with the small collection show that our system preserves with fidelity the retrieval capabilities of the original system. Moreover, both systems outperform by far the random display of images. In 80% of the cases, both systems succeed to display a relevant image after 8 iterations, while the random one requires more than 16 iterations. The average performances are shown in Figure 4.3.

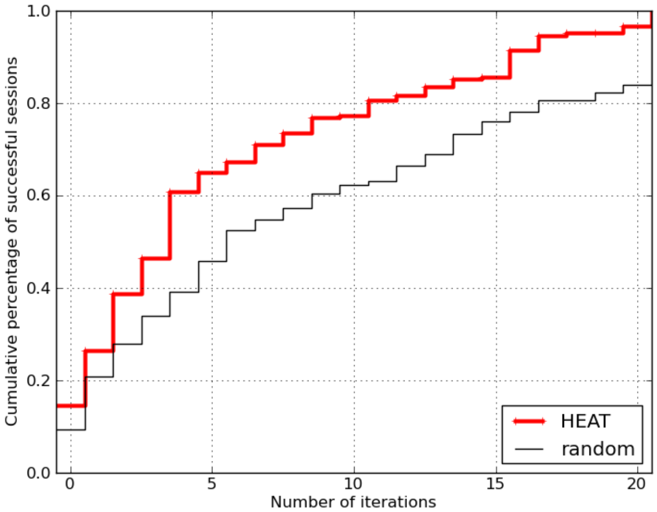
The experiments with the large collection show that our system provides a sustainable performance where the original system proposed by Ferecatu and Geman [24, 25] would cease to function within any reasonable timeframe.

The random system shows similar performance for both collections. Since both collections have a similar semantic diversity based on the ground truth given by the ImageNet, this is exactly what one would expect. Considering the randomized organization of the evaluations, the agreement of the two random baselines gives evidence that the users were consistent among the searching sessions and the performance curves are reliable.

Our evaluation scenario was meant to compare the capability of the systems to converge to semantic categories of a relatively small size. The users were told precisely to end the searching sessions the first time they were satisfied by one of the displayed images. Further evaluations should be conducted in more demanding scenarios.



(a): 33K image collection



(b): 1M image collection

Figure 4.3: Cumulative percentage of successful sessions per number of iterations. The average performances for the small collection are shown on the left: Our system performs as well as the original system proposed by Ferecatu and Geman [24, 25]. Both systems outperform by far the random display of images. In 80% of the cases, both systems succeed to display a relevant image after 8 iterations, while the random one requires more than 16 iterations to achieve the same performance. The average performances for the large collection are shown on the right: Our system shows a sustainable performance against the random system.

Precision	$t < 5$	$t < 10$	$t < 15$
(a): HEAT, 33K image collection	0.62	0.86	0.95
(b): HEAT, 1M image collection	0.65	0.78	0.86

Table 4.2: Retrieval performance. Here are a few discrete values read from Figure 4.3.

For the experiments with the small collection, the trace was limited to collapse at minimum 500 nodes, and this means that each expansion included about 3,000–4,000 nodes. This variation in the number of nodes comes from the fact that the hierarchical tree is unbalanced. For the experiments with the large collection, the trace was limited to collapse at minimum 1,000 nodes, and this means that each expansion included about 6,000–8,000 nodes. We observed that in order to maintain a similar retrieval performance the size of the trace should be slightly increased. It may be due to the larger tree that more nodes are inefficiently used just for maintaining the continuity of the trace. We will address again this issue in Chapter §8.2.3.

4.4.4 Computational impact

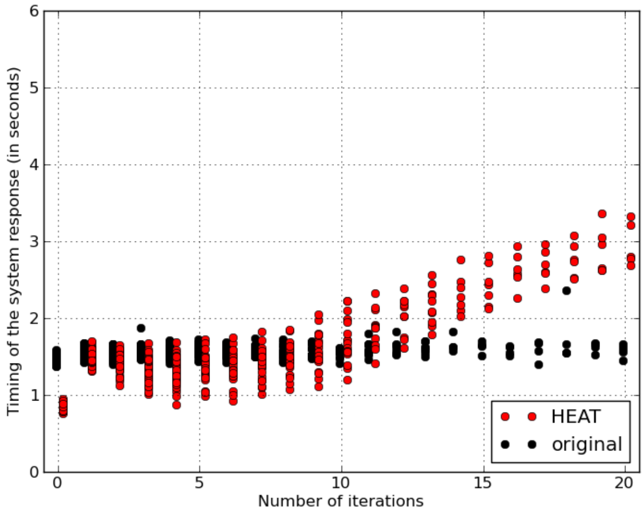
The system responses were timed during the user experiments. Although our implementation can be further optimized, these timings give a tangible evaluation of the computational effort of the systems as shown in Figure 4.4.

The computational effort of the original system is rather constant over the iterations. At each iteration, it has to update the conditional probabilities and to perform the Voronoi tessellation based on a constant number of images, namely the size of the collection. For the small collection, the system responds in about 1.5 seconds. For the large collection, the system would totally fail to respond in any reasonable time even without mentioning the required storage capacity of $\mathscr{O}(\|\Omega\|^2)$ complexity.

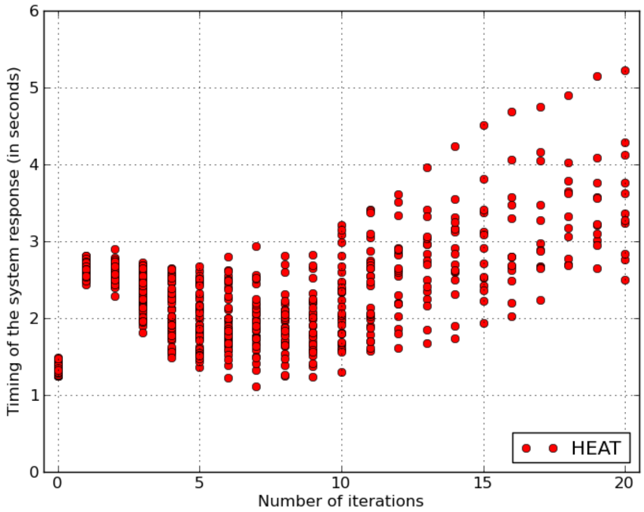
The computational effort of our HEAT system is slightly variable over the iterations. Although it has to update the conditional probabilities and to perform the Voronoi tessellation only based on the representative images and the cardinality of the nodes in the current trace, the system has to access the image feature vectors and to compute the similarity distances on-the-fly. Moreover, the computation from scratch of the conditional probabilities is linearly increasing with the number of iterations. While the original system updates recursively the probabilities only based on the last relevance feedback event, our system updates most of the probabilities from scratch based on the full history of relevance feedback events. One can observe that the nodes in the trace are constantly replaced by the refinement operation.

For a complete view of the computational complexity, the pre-processing required for organizing and indexing the image collections should be taken into account as well. As mentioned already in § 4.4.1, our experiments are using the image feature vectors provided by ImageNet, and thus our next analysis does not take into account the computation implied by the feature-extraction operations.

The pre-processing in the original system consists of computing the similarity distances between every two images in the collection, and thus it has $\mathscr{O}(\|\Omega\|^2)$ complexity. For the small collection, the required capacity for storing the similarity distances in binary files, one file per image, is nearly 4GB. For the large collection, the required storage capacity would be unacceptably large.



(a): 33K image collection



(b): 1M image collection

Figure 4.4: Timing of the system responses (in seconds) as the users experienced them during the evaluations. The timings for the small collection are shown in (a): The computational effort of the original system is constant over the iterations. The computational effort of our system stays in the same range, although it increases slowly with the number of iterations due to the computation from scratch of the probabilities of relevance. The timings for the large collection are shown in (b): The timings remain comparable with the ones for the small collection. The computational effort of our system is decoupled from the collection size, and depends mainly on the trace size.

The pre-processing in our system consists of building the hierarchical tree based on k-means clustering. The computational complexity of the divisive top-down k-means clustering does not have a closed form but it is studied in [27]. The storage of the image feature vectors has $\mathcal{O}(\|\Omega\|)$ complexity, and the storage of the hierarchical tree is truly negligible. The required capacity is only 100MB for the small collection and about 3GB for the large collection.

4.5 Summary

We have presented a retrieval approach that promises an interactive access to image collections of unprecedented size. The experiments show that this iterative relevance feedback mechanism can handle a collection of 1,000,000 images, which is already one order of magnitude larger than most of the state-of-the-art iterative approaches.

Using an adaptive partitioning of the image collection, our HEAT system provides the means for controlling the trade-off between the retrieval performance and the computational effort. This may be a crucial characteristic for real-world applications.

We foresee no barrier in scaling up the approach up to 10 million images or more. The key observation is that the trace refinement is suitable for parallel and distributed computing architectures. The trace could be divided into parts, and each part could be processed separately because the update of the probabilities of relevance is in fact an atomic operation for each individual image.

5 Exploration/Exploitation Trade-off

Content-based image retrieval systems have to cope with two different regimes: understanding broadly the categories of interest to the user, and refining the search in this or these categories to converge to specific images among them. In our retrieval framework, in contrast with other types of retrieval systems, these two regimes are of great importance since the search initialization is hardly optimal (i.e. the page-zero problem) and the relevance feedback must tolerate the semantic gap of the image visual features.

In this chapter, we analyze and improve the relevance feedback mechanism from the point of view of the exploration/exploitation retrieval trade-off. We present a new approach that encompasses these two regimes, and infers from the user actions a smooth transition between them. Starting from the original framework meant to solve the page-zero problem, we propose an adaptive exploration/exploitation trade-off that transforms the framework into a versatile retrieval system with full searching capabilities.

Our proposed approach is compared to the state-of-the-art approach it extends by conducting user evaluations on a collection of 60,000 images sampled uniformly from the ImageNet dataset. Evaluation gives evidence that this new approach brings a significant improvement of the retrieval capabilities beyond finding an image category, and is able to support refining the user interest in an efficient manner.

5.1 Introduction

The interactive retrieval process involves two different regimes. The first one can be seen as an exploration phase, during which the user communicates to the system her categories of interest in a broad way. This first regime transitions into the second one that can be seen as an exploitation phase, where the user specifies more detailed requirements on the visual properties of images, making the system intelligently explore the restricted subset specified during exploration.

Initially we started to investigate new ways for modeling the user relevance feedback being motivated by the observation that the efficiency of the relevance feedback mechanism depends on the distribution of the probabilities of relevance. The main goal was to find a way to adapt the model dynamically, on-the-fly at each iteration, in order to accelerate the relevance feedback convergence.

As we observed in our previous experiments, the original approach is well suited for image category search and that is, in other words, the first retrieval regime of exploring the image collection. Still, finding a specific image that has already a relatively high probability of relevance is problematic since the strategy to select the next displayed images is rigid and insists on sampling the entire collection.

Our core contribution is an adaptive modulation of the exploration/exploitation trade-off, which leads to a versatile retrieval system with full searching capabilities. Internally, our approach employs an estimator of the consistency between the system internal state and the user retrieval objective, and controls dynamically, at each iteration, the selection of the displayed images accordingly.

We set up our web-system for a collection of 60,000 images sampled uniformly from the ImageNet database [18], for which we took over the provided pre-computed SIFT features (Scale Invariant Feature Transform) [41]. We set up four configurations with different similarity metrics and we run user-based evaluations with 20 users. Evaluation gives evidence that our approach brings a significant improvement on the retrieval capabilities of the original system that remains sustainable when employing different similarity metrics.

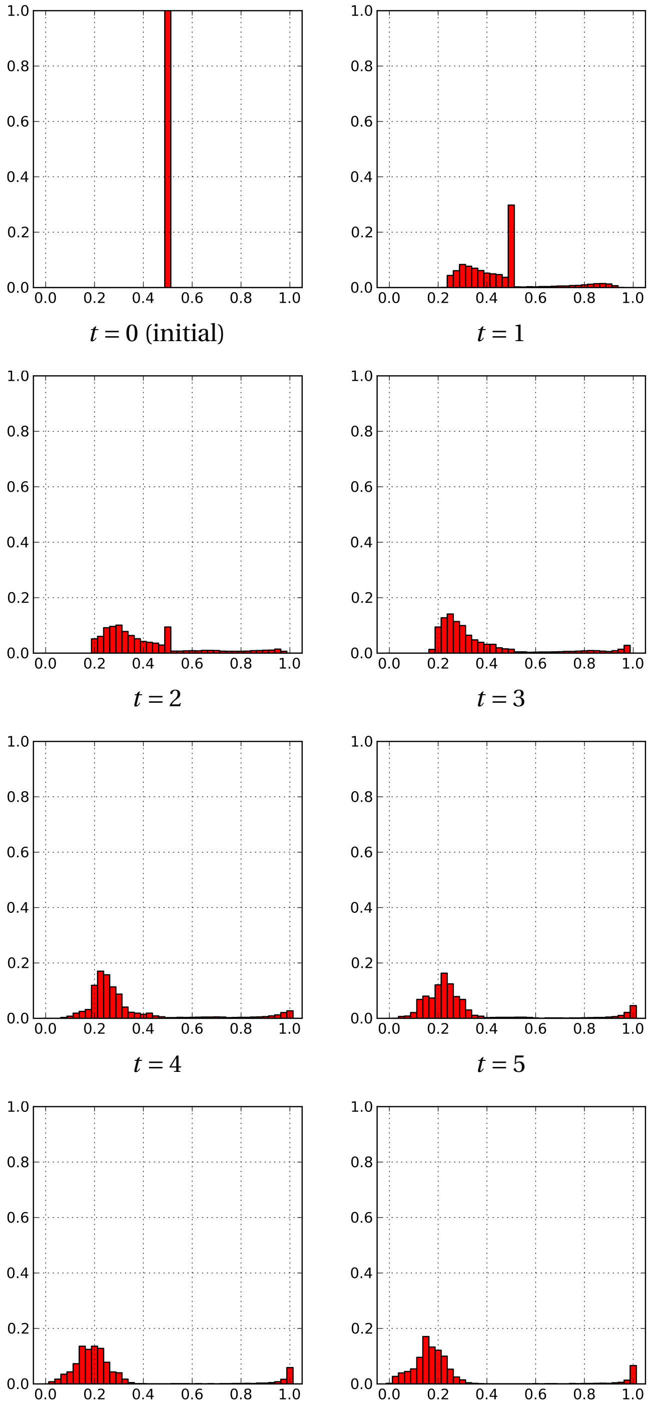
5.2 State-of-the-art

Research proposed many alternative approaches to tackle the two retrieval regimes of exploration and exploitation. Traditionally, they are seen as separate operations and they are treated by separate algorithms. We share the idea that any searching session can be considered as having a mix of exploration and exploitation, and it would be advantageous if they could be treated in a unified way.

As argued by Ferecatu and Geman [24, 25], the original retrieval framework is well suited for image category search and that is, in other words, the first retrieval regime of exploring the image collection. They explicitly suggest that other retrieval techniques should be employed to retrieve specific images among these identified categories and that is, in other words, the second retrieval regime of exploiting the image collection.

A useful insight is given by analyzing the evolution of the retrieval system for the synthetic collection, when searching for images with points close to the center. Figure 5.2 shows the evolution of the displayed images, and Figure 5.1 shows the distribution of the probabilities of relevance.

As shown in Figure 5.1, the distribution of the probabilities evolves quite rapidly in the first iterations. These early iterations correspond to the first retrieval regime when the system is



$t = 0$ (initial) $t = 1$

$t = 2$ $t = 3$

$t = 4$ $t = 5$

$t = 6$ $t = 7$

Figure 5.1: Evolution of the distribution of probabilities of relevance for the searching session illustrated in Figure 5.2. The plots have the probability bins on axis X, and the percentage of images in the collection on axis Y. Initially, all images have the same probability, $p_0(k) = 0.5 \ \forall k \in \Omega$. The distribution evolves rapidly in the first iterations, and evolves slowly after the very first iterations.

in the process of understanding broadly the categories of interest to the user. Later, after the system has achieved a good understanding of the user interest, the distribution of the probabilities evolves quite slowly from one iteration to another. These later iterations correspond to the second retrieval regime when the system is meant to refine the search and to converge to specific images.

As shown in Figure 5.2, the sets of displayed images include an image that is closer and closer, with each iteration, to the user interest. After 3 iterations, the system succeeds to display an image that is clearly in the intended region. Still after 5 iterations, the displayed images concentrate only slightly on the intended region.

The system succeeds efficiently to display an image in the intended region, but it has a hard time to display more and more images in the intended region. The "sampling" algorithm insists on covering the entire collection even after the distribution of probabilities becomes rather stable. One can say that the original system has a big inertia to maintain an exploration regime, and goes very slowly into an exploitation regime.

5.3 Mass-zoom system

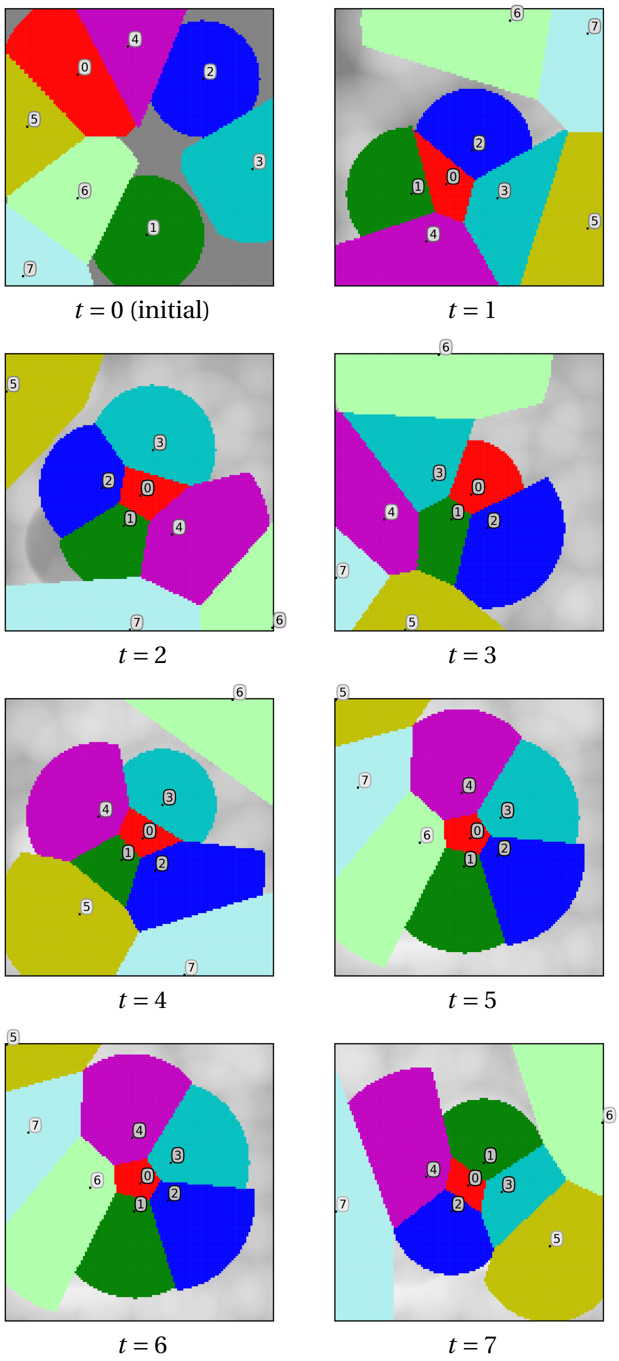
This section presents our solution to eliminate the limitations of the retrieval framework described in §5.2. Intuitively, the system should be aware of the degree of alignment of the distribution of probabilities with the user intent. When the distribution of probabilities is in line with the user intent, the system should concentrate the "sampling" on the regions with high probability.

First, we present the idea of the adaptive strategy to handle the trade-off between exploration and exploitation, by modulating the concentration of the display set on promising images. Second, we present a heuristic that infers dynamically, at each iteration, from the user actions a consistency score that achieve a smooth trade-off that suits the user intent.

5.3.1 Exploration/exploitation trade-off

Our mass-zoom algorithm handles the trade-off between exploration and exploitation by modulating how much the display set should be concentrated on the images assessed as the most relevant. This is achieved by estimating at every iteration the target mass m_t for the displayed image neighborhoods. While this value was a constant fraction of the total mass in the baseline in Equation (3.9) from § 3.3, we propose to link it to an estimate of the confidence of our current estimate of the image relevance. Making the value of this target mass smaller makes the neighborhoods around the images of the display set smaller, which leads to a more compact display set, concentrated on the area of high probability.

Our approach increases the concentration of the display set if the choice of the user is consistent with our current estimate, and decreases it otherwise. We propose the following update



$t = 0$ (initial)

$t = 1$

$t = 2$

$t = 3$

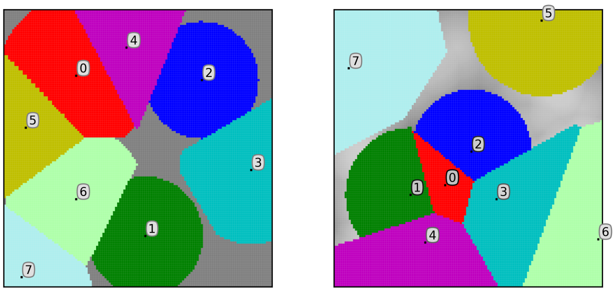
$t = 4$

$t = 5$

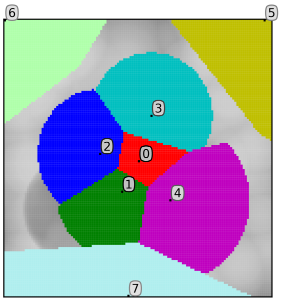
$t = 6$

$t = 7$

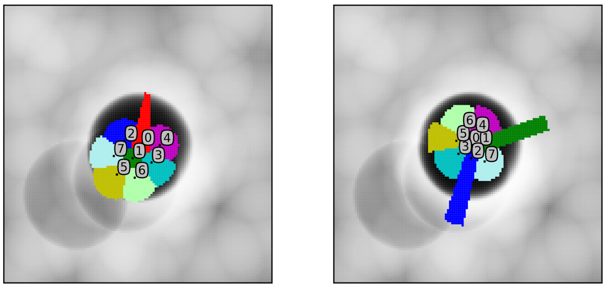
Figure 5.2: Evolution of the display set for the original framework with the synthetic collection, when searching for images with points close to the center. After 5 iterations, the displayed images concentrate slightly on the intended region. Again, the selected images are marked in black and their corresponding Voronoi cells are indicated by colors.



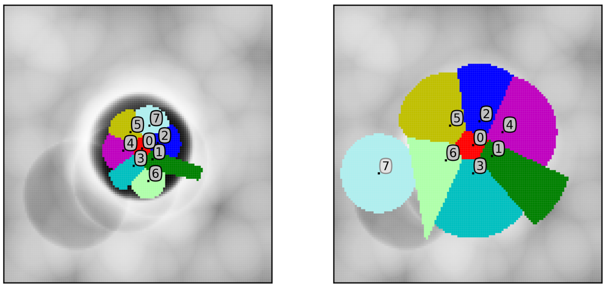
$t = 0$ (initial) $t = 1$



$t = 2$ (exploration) $t = 3$



$t = 4$ (exploitation) $t = 5$ (exploitation)



$t = 6$ $t = 7$ (exploration)

Figure 5.3: Evolution of the display set for the mass-zoom system with the synthetic collection, when searching for images with points close to the center. After 5 iterations, the displayed images concentrate mostly on the intended region. The displayed images provide the freedom to escape the exploitation if necessary. The system continuously estimates the exploration/exploitation trade-off that suits the user.

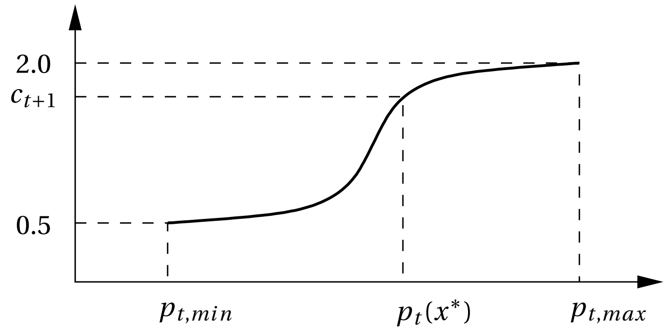


Figure 5.4: Consistency scores are estimated based on the cumulative distribution function for the Gaussian distribution. Our heuristic gives a consistency score in the [0.5, 2.0] interval.

scheme:

$$m_t^{zoom} = z_t \cdot m_t, \tag{5.1}$$

where $z_t \in \left(\frac{1}{m_t},\ 1\right]$ accounts for the consistency between our estimates of the p_t and the user choice.

5.3.2 Heuristics based on a consistency score

Immediately after the relevance feedback event $\{D_t,\ x_t^*\}$, right at the beginning of the next iteration $t+1$, the consistency score aims to estimate the alignment of the system and the user intent, which is defined in Equation 5.2 as the probability, under our model and given the internal state of the system, of choosing the image $x_t^* \in D_t$ that was actually chosen:

$$c_{t+1} \simeq \widehat{P}_{x \sim \mathscr{U}(D_t)} \left[p_t(x_t^*) \geq p_t(x) \right]. \tag{5.2}$$

In the first iteration, the user intent is totally unknown and the consistency score c_0 is initialized to 1.0. Subsequently, the consistency score is estimated based on the probability of relevance of the chosen image $p_t(x_t^*)$ versus the probabilities of relevance of the other displayed images, namely $p_t(x_t)$, for all $x \in D_t$.

Table 5.1: Notation

m_t	target mass for building the display set in the original system
m_t^{zoom}	target mass in the mass-zoom approach
c_t	consistency score at iteration t
z_t	change of the target mass at iteration t

The consistency score is estimated based on the cumulative distribution function for the Gaussian distribution. The proposed heuristic gives a consistency score in the [0.5, 2.0] interval:

$$c_{t+1} = 0.5 + 1.5 \cdot \left(\frac{1}{2} + \operatorname{erf}\left(\frac{p_t(x^*) - \mu}{\sigma \cdot \sqrt{2}} \right) \right), \tag{5.3}$$

where μ is the average of the probabilities in $\|D_t\|$:

$$\mu = \frac{1}{\|D_t\|} \cdot \sum_{x \in D_t} p(x), \tag{5.4}$$

and σ is the standard deviation of the probabilities in $\|D_t\|$:

$$\sigma^2 = \frac{1}{\|D_t\|} \cdot \sum_{x \in D_t} (p(x) - \mu)^2. \tag{5.5}$$

This is motivated by the intuition that if the $p_t(x_t^*)$ is already among the highest probabilities it means that the system has a distribution of the probabilities that is in line with the user intent, and thus the system is consistent with the user intent. If $p_t(x_t^*)$ is relatively low, the system is less consistent with the user intent.

The zoom value that impacts the exploration/exploitation trade-off of the selection of the displayed images is derived from the consistency scores as follows:

$$z_t = \prod_{i=0}^{t} \frac{1}{c_i}. \tag{5.6}$$

5.3.3 Capabilities of the mass-zoom system

For an intuitive illustration, we set up the mass-zoom system for the synthetic collection described in §3, and once again take the case of searching for images with points close to the center. In Figure 5.3, we show the evolution of the displayed images for intermediate iterations during one such searching session.

After efficiently identifying the intended region, the mass-zoom system is able to display more and more images in the intended region. The "sampling" algorithm concentrates on the intended region after the distribution of probabilities becomes rather stable. Although the "sampling" algorithm does not cover the entire collection anymore, the system continuously estimates the exploration/exploitation trade-off that suits the user.

Note that while the synthetic collection is very handy for intuitive illustrations, it should not be mistaken for a real image collection, which typically encompasses high-dimensional image indexing feature spaces. Besides the miss-alignment between the image feature space and the user subjective perception of image similarities, the distribution of the image similarity distances impacts the Voronoi tessellation algorithm as well as the distribution of the probabilities of relevance. We argue that the exploration/exploitation trade-off has even higher impact than in the case of the synthetic collection.

5.4 Experimental results

Evaluation was conducted with 20 users not familiar with the system, and consisted of running user tests with three systems:

- our proposed mass-zoom system
- the original baseline system
- a random system that displays images randomly without replacement

5.4.1 System setup

The system was set up for 60,000 images sampled uniformly from the ImageNet database [18], that has the convenience of being structured in 1000 semantic categories, each composed of 500–2500 images. We considered the semantic information as benchmark meta-data for setting up the evaluation scenario, and we used as indexing features the pre-computed bags of SIFT features of dimension 1000 (Scale Invariant Feature Transform) [41], as they are provided along with the images. For evaluation purposes, we considered four different image similarity metrics defined over these histogram-like indexing feature vectors:

- Euclidean distance (L^2)
- Isomap distance [60] derived from the 16 L^2 nearest neighbors (L^2-Iso16)
- Manhattan distance (L^1)
- Isomap distance derived from the 16 L^1 nearest neighbors (L^1-Iso16)

The relevance feedback framework was calibrated as described in [24], and the parameters of the image similarity model are adjusted to saturate only after including on average 10% of the images in the collection. Therefore, each similarity metric would employ a different image similarity model, adapted to its statistical properties.

5.4.2 Evaluation scenario

The aim of our experiments was to evaluate our mass-zoom system in terms of the retrieval capabilities, and to get evidence that our system is capable of providing capabilities beyond

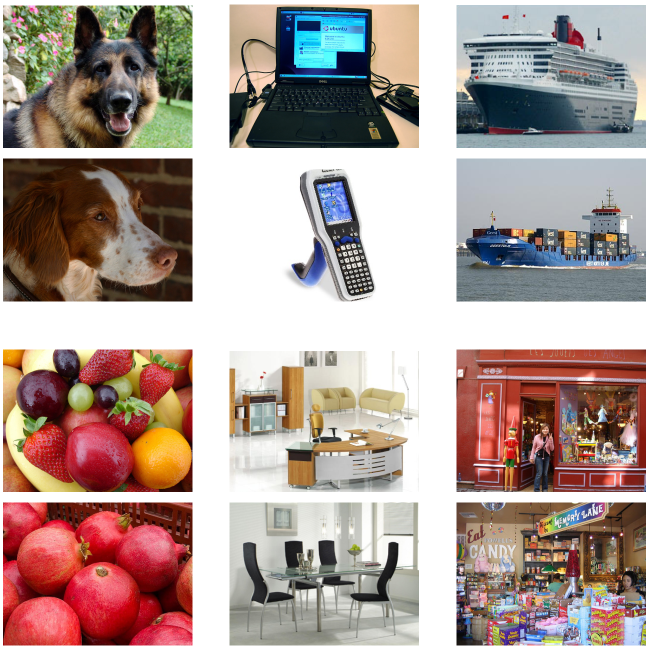


Figure 5.5: The users were asked to search for semantic categories described in words and accompanied by image examples as shown here. In order to ensure a sufficiently reliable diversity, there were 6 semantic categories.

finding an image category, and is able to support refining the user interest in an efficient manner.

In order to isolate our contribution as much as possible, we employed four different similarity metrics on top of the image indexing features, as mentioned in §5.4.1. We did not aim to evaluate which similarity metric suits better the user subjective perception of image similarity, but rather to gather evidence that our contribution remains sustainable when employing different similarity metrics.

The evaluations follow the scenario in §3.5.2. In order to ensure a reliable diversity and comparable difficulty, there were 6 semantic categories described in words and accompanied by the corresponding images in Figure 5.5:

- portraits/close-ups of dogs, wolves
- electronic devices as laptop, mobile phone
- big boats as ferryboats, cargoes
- baskets/plates with fruits, vegetables
- furniture items as tables, chairs
- entrances/windows of shops, shopping centers

In order to avoid any bias, the searching sessions were presented in a random fashion. The semantic categories, the systems and the similarity metrics were randomized all together in one single user test. The users were not aware of which configuration was active in a certain session.

The users were told to end the searching sessions when they were satisfied by four of the displayed images, instead of just only one. We designed the evaluation scenario in this way with the intent of pushing the evaluation beyond a simple image category search. We looked for evidence that the system is able to properly identify the user interest and then refine it more and more in an efficient way.

5.4.3 Results analysis

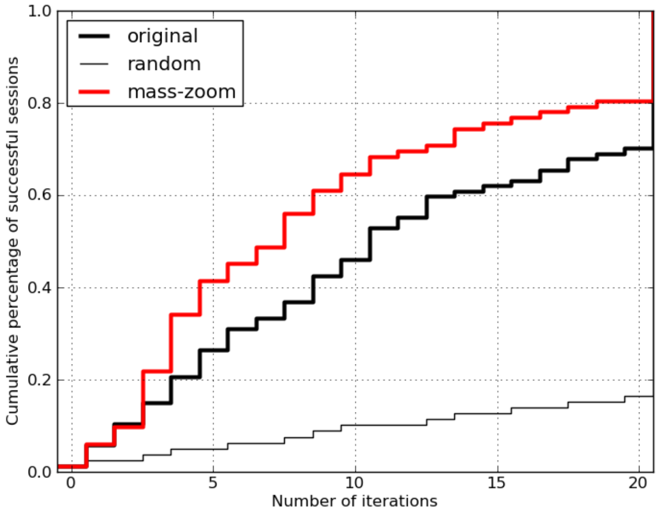
Evaluation shows that the mass-zoom approach is viable. The mass-zoom system is consistently better than the baseline for all configurations. Figures 5.6-5.7 show the cumulative percentage of successful sessions per number of iterations. For example, for L^1 similarity metric, the mass-zoom system finishes successfully in less than 10 iterations in 68% of the cases, and the baseline in only 44% of the cases. The random system is far from achieving the same performance even after 20 iterations. Table 5.3 contains a few discrete values read from Figure 5.6.

Table 5.2 tells about the statistical significance of the evaluation. For each couple of configurations, we counted how many times one performed better than the other for the same user and the same semantic category.Then, we computed the binomial probabilities. In principle, a difference is statistically significant if the corresponding probability is smaller than 0.05.

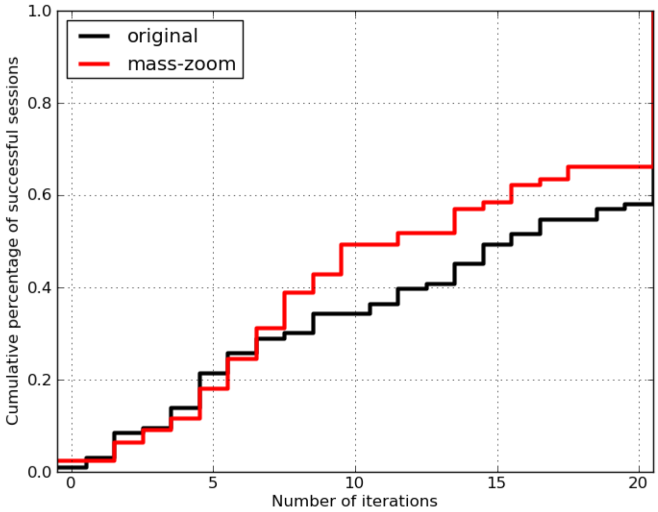
Figure 5.8 shows the evolution of the zoom values z_t from one iteration to the next. By decreasing in average, it shows that the system is consistent with the user interest. One should be aware that the rate by which the system transitions from exploration phase into exploitation phase in Equation (5.3) may affect the results. An optimal rate could be derived by a more extensive user evaluation.

Metrics	**Mass-zoom/Baseline**	**Mass-zoom/Random**
L^2	(35/60) 0.078	(53/60) 0.000
L^2-Iso16	(40/60) 0.004	(50/60) 0.000
L^1	(37/60) 0.026	(56/60) 0.000
L^1-Iso16	(42/60) 0.001	(45/60) 0.000

Table 5.2: Binomial-test for statistical significance for all four similarity metrics, corresponding to the experiments in Figures 5.6-5.7. For example, for L^1 similarity metric, the mass-zoom system performed better than the baseline in 37 times out of 60, and the probability of this to occur by chance is 0.026.



(a): L^2

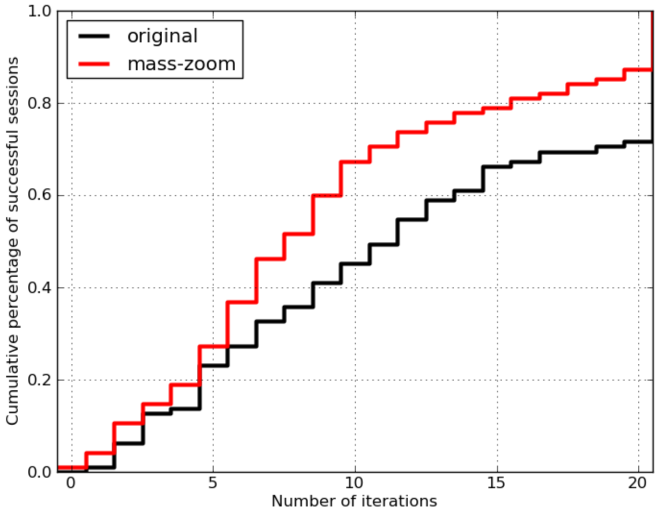


(b): L^2-Iso16

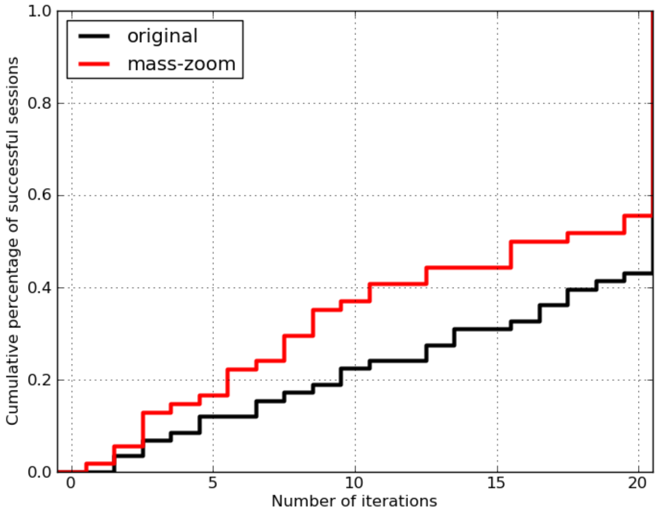
Figure 5.6: Cumulative percentage of successful sessions per number of iterations. Our mass-zoom system shows a sustainable performance against the baseline proposed by Ferecatu and Geman [24, 25] for all four similarity metrics. For example, for L^2 similarity metric, the mass-zoom system finishes successfully in less than 10 iterations in 65% of the cases, and the baseline in only 45% of the cases.

Metrics	**Precision**		
	$t < 5$	$t < 10$	$t < 15$
(a): L^2	0.40/0.30	0.65/0.45	0.75/0.60
(b): L^2-Iso16	0.20/0.20	0.50/0.35	0.60/0.50

Table 5.3: Retrieval performance. Here are a few discrete values read from Figure 5.6.



(a): L^1

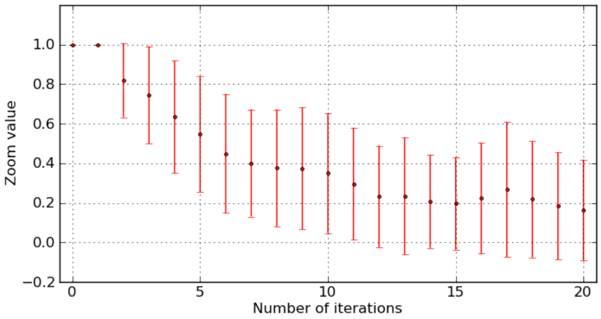


(b): L^1-Iso16

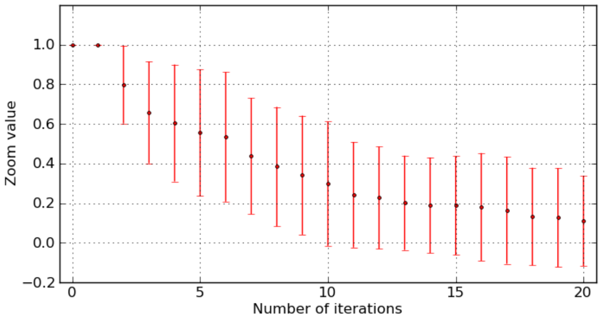
Figure 5.7: Cumulative percentage of successful sessions per number of iterations. Our mass-zoom system shows a sustainable performance against the baseline proposed by Ferecatu and Geman [24, 25] for all four similarity metrics. For example, for L^1 similarity metric, the mass-zoom system finishes successfully in less than 10 iterations in 68% of the cases, and the baseline in only 44% of the cases.

Metrics	**Precision**		
	$t < 5$	$t < 10$	$t < 15$
(a): L^1	0.30/0.25	0.70/0.45	0.80/0.65
(b): L^1-Iso16	0.20/0.15	0.40/0.20	0.50/0.30

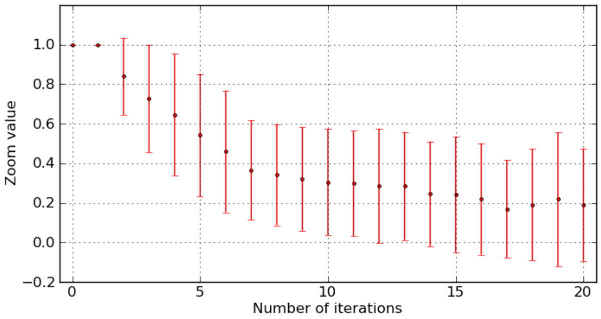
Table 5.4: Retrieval performance. Here are a few discrete values read from Figure 5.7.



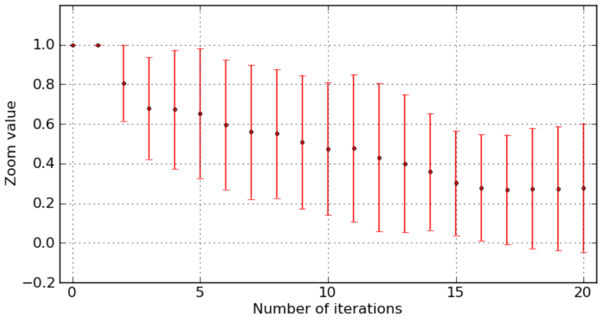
(a): L^2



(b): L^2-Iso16



(c): L^1



(d): L^1-Iso16

Figure 5.8: Zoom-factor average and standard deviation. z_0 and z_1 are always equal to 1 as c_0 is initialized with 1 since there is no relevance feedback history at iteration $t = 0$, and c_1 is always equal to 1 since the probabilities of relevance $p_0(k)$ are all equal to 0.5.

5.5 Summary

We have presented the mass-zoom system that offers full searching capabilities. This adaptive system encompasses both retrieval regimes of exploration and exploitation, and supports a smooth transition between them that increases the consistency between the system and the user.

Internally, our approach employs an estimator of the consistency between the system internal state and the user retrieval objective, and controls dynamically, at each iteration, the selection of the displayed images accordingly.

We demonstrated its feasibility by conducting user evaluations on a collection of 60,000 images sampled uniformly from the ImageNet database. Evaluation shows that our proposed mass-zoom system extends considerably the retrieval capabilities of the original algorithm. Our results give motivation for further investigations on other heuristics or finding more principled ways of trade-off.

6 Log-based image similarity

Interactive image retrieval based on user relevance feedback strongly depends on the extent to which the “closeness” in the similarity metric (i.e. the distances between the image feature vectors) accounts for the “closeness” in the subjective perception of the users.

In this chapter, we propose to improve the similarity metric on which our framework relies by learning from user logs to adapt the low-level image features, and model explicitly the user similarity judgments. Internally, we define a weighted Euclidean distance over the image feature space, and then we optimize it in order to maximize the probabilities of relevance of the images chosen by the users under the probabilistic model of their interactions with the system.

Our technique is evaluated by 20 users on two collections from ImageNet, a small collection of 60,000 images and a large collection of 1,000,000 images, and shown to bring about 10% improvement in the retrieval performance in comparison with the original system.

6.1 Introduction

The image similarity metric obviously has a direct impact on the overall efficiency of the system. The retrieval performance strongly depends on the extent to which the “closeness” in the similarity metric (i.e. the distances between the image feature vectors) accounts for the “closeness” in the subjective perception of the user. Intuitively, the retrieval performance would significantly improve if the similarity metric was better aligned to the user perception.

We started to investigate the possibilities to improve the retrieval capabilities of the original framework by modeling the user similarity judgments beyond the relevance feedback information given during a single searching session. We aimed to identify a solution that is able to support personalized similarity metrics eventually.

We propose to derive a more optimal similarity measure between the images by exploiting the user feedback histories that are acquired naturally during the searching sessions (i.e. user logs information). During the on-the-fly sessions, the user feedback histories are stored as user logs in a database. Then, as the user logs are accumulated in the database, they can be used off-line to improve the similarity metric. This approach has the major advantage that it does not require any extra image labeling that would imply additional effort and resources. As the retrieval system is used, the user logs will gradually cover the entire collection, and the log-based similarity metric will systematically improve.

We observe that the user feedback can be seen as a weak partially reliable image labeling, and we formulate a technique that tunes off-line the image similarity metric in order to model explicitly the user similarity judgments. Internally, we define a weighted Euclidean metric over the image feature space, and then we optimize it in order to maximize the probabilities of relevance of the images chosen by the users. Then, we use this optimized log-based metric instead of the original L^2 norm on-the-fly to run the retrieval process.

We evaluated this technique on our image collection of 60,000 images from ImageNet, for which we could make use of the user logs from some previous experiments. Evaluation shows that the log-based similarity metric improves the retrieval performance, which means that the optimization scheme succeeds to adapt the low-level indexing information in order to align it better with the users' similarity judgments.

6.2 State-of-the-art

A key factor of our retrieval framework, as in case of any other relevance feedback framework, is the similarity measure between the images that governs the retrieval process. Ferecatu and Geman in [25] elaborate a calibration technique that aims to compensate for the semantic gap between the similarity metric and the user perception of image similarity. While their calibration requires an image labeling task conducted separately from the searching sessions, we propose a new technique that has the advantage of exploiting the user input acquired naturally during the searching sessions.

The general idea is that the similarity metric must be aligned reasonably well with the users' similarity judgments, and the better the alignment, the more reliable the relevance feedback. Although there are plenty of sophisticated image similarity metrics [20, 15], it has been recognized that it is too ambitious to expect a single automatically-derived metric to model reasonably well the user perception of image similarity. The similarity metrics should go beyond the low-level automatically-derived indexing information and should model explicitly the user perception.

Extensive research has been done in order to derive similarity metrics fully based on user input such as for example relative similarity of pairs of images [11]. Unfortunately, collecting such user input is as prohibitive as the traditional manual annotation of the images, and is not suitable for large-scale collections.

An interesting alternative is to attempt to tune an existing, automatically generated, similarity metric by learning from the user feedback. The use of relevance feedback for learning the correlation between low-level indexing features and high-level semantics has been attempted by Han et al. [28] and Hoi et al. [31, 32]. Recently, a manifold learning technique to capture the user preferences over a semantic manifold has been proposed by Lin et al. [40].

Our retrieval framework is particularly feasible for such machine-learning approaches that require user logs information (i.e. image labeling). Since the relevance feedback is the core mechanism of searching, the user input is acquired naturally during the searching sessions without any extra-effort. Still, the task of learning might be challenging since there is only one positive image example per iteration, and no explicit negative examples.

Our retrieval system stores all the searching sessions that are performed by the users during the evaluation campaigns in so-called user logs. Each user log corresponds to a searching session, and contains all the data necessary to recall the context of the evaluation campaign and to reproduce that corresponding session. Besides the information about the user, the system configuration and the target task, each user log contains the history of relevance feedback events $\{D_t,\ x_t^*\}$, $t = 1, 2, \ldots T$ where $T \leq 20$, and the user label (i.e. successfully terminated or failed).

For our investigations, we have already a sufficient amount of user logs from our previous evaluation campaigns. For the collection of 60,000 images, there is log information for about 50% of the collection (i.e. 30,000 images). About 50% of the images have been displayed at least 1 time, and about 36% at least 2 times, and so on. For the collection of 1,000,000 images, there is log information for 2% of the collection (i.e. 20,000 images).

6.3 Log-based similarity metric

The retrieval framework could employ any arbitrary similarity measure between the images, but ofter this is just the L^2 or L^1 norm between some automatically-generated image feature vectors. Obviously, such a simple visual-based similarity metric is sub-optimal and the semantic gap is quite significant.

We propose to tune such a trivial similarity metric by weighting the individual features based on the user logs information. We share the idea that the tuning technique should exploit the user logs in an incremental fashion, by gradually taking into account the user logs as they

Table 6.1: Notation

$\boldsymbol{\alpha}$	weighting vector learned off-line from the user logs
$d_{\boldsymbol{\alpha}}(k,h)$	weighted Euclidean distance between k and h
C	cost function defined on the probabilities of relevance of the images chosen by the users

become naturally accumulated in the database. As the retrieval system is used, the user logs will gradually cover the entire collection, and the log-based similarity metric will systematically improve.

6.3.1 Weighted Euclidean distance

The original framework employs a similarity metric that is the Euclidean metric over visual-based image feature vectors based on SIFT (Scale Invariant Feature Transform) [41]. Our approach re-defines the similarity metric as a weighted Euclidean distance over the image feature space as in Equation (6.1). Thus, we introduce a weighting vector $\boldsymbol{\alpha}$ for which we will elaborate an optimization scheme based on the user logs.

$$d_{\boldsymbol{\alpha}}(k,h) = \sqrt{\sum_{f=1}^{F} \alpha_f \cdot (k_f - h_f)^2}, \tag{6.1}$$

where F is the dimensionality of the image feature space.

6.3.2 Log-based weights learning

Our retrieval system stores all the data necessary to reproduce the searching sessions in so-called user logs. These user logs contain the history of relevance feedback events $\{D_t,\ x_t^*\}$, $t = 1,2,\dots T$ where $T \leq 20$, as well as all the system parameter settings, for each searching session. Furthermore, each searching session is user labeled as successfully terminated or failed.

Our challenge is that the searching session are labeled only globally as successfully terminated or failed. This tells us if the last display set contains target images or not, but unfortunately does not tell explicitly which images.

If we knew explicitly the image or the images that satisfied the user, we could adapt the weighting vector in order to maximize the probabilities of relevance of those images. This way the weighted Euclidean distance would provide distributions of probabilities that are more consistent with the user intent.

Our alternative is to adapt the weighting vector in the sense of making the probabilistic model able to predict better the images chosen by the user. We consider that, in the searching sessions that were successfully terminated, all the history of relevance feedback events was for good and helped the user to get to the final display set that satisfied her. With this assumption, all the history of relevance feedback events are regarded as equally important, and it makes sense to adapt the weighting vector in order to maximize the probabilities of all the images chosen in all relevance feedback events.

With these considerations, we define the cost function as the total sum-log of the probabilities of relevance of the chosen images at the time of their displaying:

$$C = \sum_{u=1}^{U} \sum_{t=0}^{T} \log p_t(x_t^*), \tag{6.2}$$

where U stands for all the user logs (i.e. searching session histories), and T stands for the number of iterations of each searching session.

Next, we should choose an optimization algorithm in order to learn the optimal weighting parameter $\boldsymbol{\alpha}$ that maximizes the cost function in Equation (6.2):

$$\boldsymbol{\alpha}_{optim} = \arg\max_{\boldsymbol{\alpha}} C. \tag{6.3}$$

We propose to optimize the weighting vector based on the full-batch gradient descent method combined with a simple line search. If the amount of user logs becomes large, the cost function could be optimized using approximations, as for example the stochastic gradient descent method.

Initially, at iteration $n = 0$ the weighting vector $\boldsymbol{\alpha}^0$ is set to $\mathbf{1}$. In the subsequent iterations, the gradient descent algorithm is performed according to Equation (6.4), which can be expanded for each weighting coefficient α_i as in Equation (6.5):

$$\boldsymbol{\alpha}^{n+1} = \boldsymbol{\alpha}^n + \gamma_n \cdot \nabla C(\boldsymbol{\alpha}^n),\ n > 0, \tag{6.4}$$

$$\begin{aligned} \alpha_i^{n+1} &= \alpha_i^n + \gamma_n \cdot \nabla C(\alpha_i^n) \\ &= \alpha_i^n + \gamma_n \cdot \left.\tfrac{\partial C}{\partial \alpha_i}\right|_{\alpha_i = \alpha_i^n}. \end{aligned} \tag{6.5}$$

The partial derivatives can be elaborated starting from the top derivative:

$$\frac{\partial C}{\partial \alpha_i} = \sum_{u=1}^{U} \sum_{t=0}^{T} \frac{1}{p_t(x_t^*)} \cdot \frac{\partial p_t(x_t^*)}{\partial \alpha_i}. \tag{6.6}$$

6.4 Experimental results

The aim of the following experiments is to evaluate if the optimization scheme we propose succeeds to adapt the low-level indexing information in order to align it better with the users' similarity judgments, and thus to improve the retrieval performance of the original framework.

We are using for this evaluation the same collection of 60,000 images from ImageNet dataset as in §5.4, and with the same indexing features, namely the bags of SIFT features provided by ImageNet. Thus, we are able to make use of the user logs from our previous evaluation campaign.

Our experiments are presented in two parts. First, we analyze the robustness of the optimization scheme in §6.4.1, and we obtain the optimal weighed Euclidean distance in order to set up the optimized system. Second, we evaluate the efficiency of the proposed technique by conducting a user-test campaign, for which we give the set up details in §6.4.2, and interpret the outcome in §6.4.3.

6.4.1 Log-based weights analysis

As already mentioned in §6.2, we make use of the user logs from some of our previous experiments in §5.4. In our optimization scheme, we considered all successfully terminated searching sessions that were performed for the L^2 type of distances. There are in total 142 searching sessions, that results in a cumulative set of 1050 relevance feedback events. With these data, we performed the full-batch gradient descent algorithm with a simple line search. The algorithm converged after 8,000 iterations, but we let it run more, and stopped it after 10,000 iterations.

Figure 6.1 shows the histogram of the weights in the final optimal weighting vector $\boldsymbol{\alpha}$, which were obtained after performing the gradient descent algorithm. Here, we recall that the original distances are equivalent to the uniform weighting vector $\mathbf{1}$. We can see that the optimal weighting vector remains reasonably bounded although no upper constrains have been enforced. About 15% of the image features are zeroed, and the maximum weight is no larger than 5.

Figure 6.2 shows the cumulative distributions of the image similarity distances in the collection for both the original and the optimized distances. We can see that the distributions remain rather alike, which means that the weighting vector is normalized properly by the optimization scheme. About 10% of the distances in the collection are smaller than 75, and about 10% of them are larger than 125. The majority of 80% of the distances are in the range 75-125, and this is the “spherical” effect of the Euclidean distance on the high-dimensional feature vectors.

Figure 6.4 shows the influence of the calibration parameters on the cost function, for both the original and the optimized distances. Each plot corresponds to a calibration parameter and shows how the cost function depends on that parameter while keeping the others un-changed. We can see that cost functions corresponding to the optimized distances give very much the same peaks as the ones corresponding to the original distances, which means that the weighting vector is normalized properly by the optimization scheme, without imposing any ad-hoc hard constraints. The only parameter that may differ from our initial settings is φ^-, as the cost function is maximized when φ^- collapses to 0. Here, we observe that in fact $\varphi^- = 0$ is not a critical setting in our setup since there are not many small distances in between the images in the collection, as we explain in Figure 6.2.

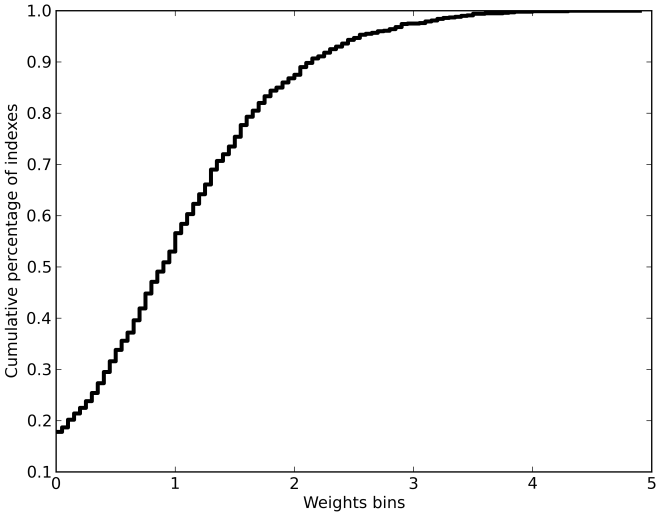


Figure 6.1: Histogram of the feature weights in the optimal weighting vector $\boldsymbol{\alpha}$, which was obtained after running the gradient descent algorithm. Here, we recall that the original distances are equivalent to the uniform weighting vector $\mathbf{1}$. We can see that the optimal weighting vector remains reasonably bounded although no upper constrains have been enforced. About 15% of the image features are zeroed, and the maximum weight is no larger than 5.

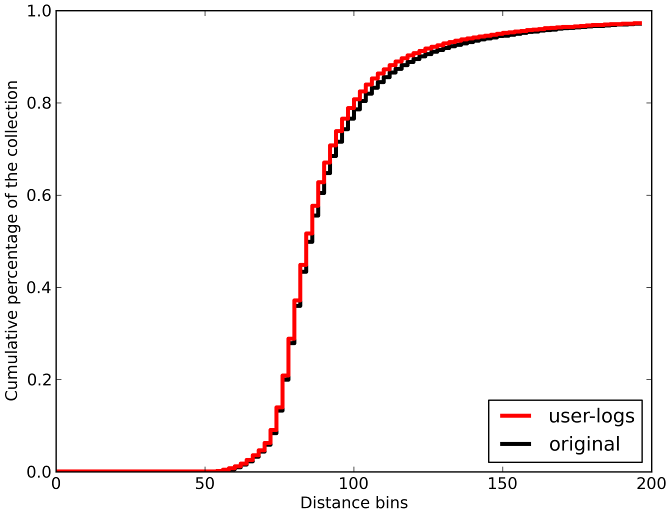


Figure 6.2: Cumulative distribution of the similarity distances in the collection, for both the original distances and the optimized distances. We can see that the distributions remain rather alike, which means that the weighting vector is normalized properly by the optimization scheme. About 10% of the distances in the collection are smaller than 75, and about 10% of them are larger than 125. The majority of 80% of the distances are in the range 75-125, and this is the “spherical” effect of the Euclidean distance on the high-dimensional SIFT-based feature vectors.

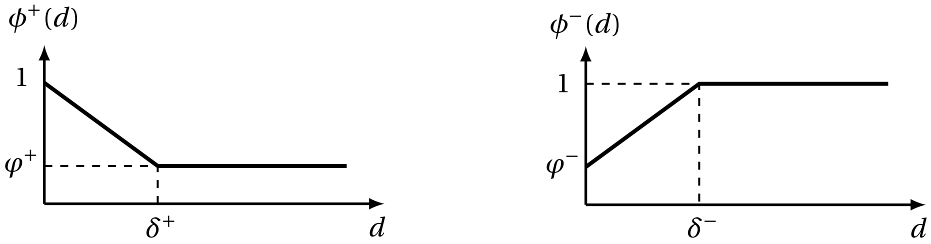
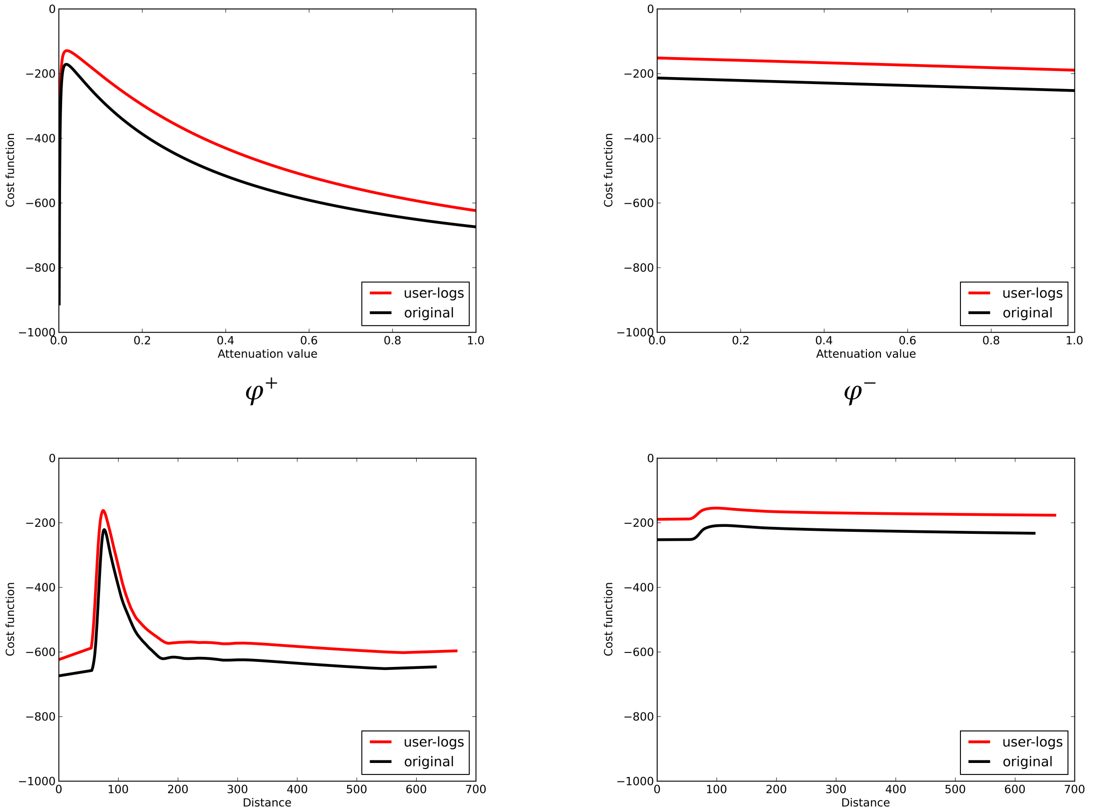


Figure 6.3: Here we recall the calibration functions in Figure 3.3, for identifying the calibration parameters that we are referring in the next Figure 6.4. δ^+, δ^- are the thresholds that normalize the distances, and φ^+ and φ^- are the attenuations that compensate for the partial mismatch between the distances and the user perception of image similarities.



φ^+ φ^-

δ^+ δ^-

Figure 6.4: Influence of the calibration parameters on the cost function, for both the original and the optimized distances. To identify the parameters, one should recall the calibration functions in Figure 6.3. Here, each plot corresponds to a parameter and shows how the cost function depends on that parameter while keeping the others un-changed. We can see that the cost functions corresponding to the optimized distances give similar peaks as the ones corresponding to the original distances, which means that the weighting vector is normalized properly by the optimization scheme.

6.4.2 Evaluation scenario

Evaluation was conducted with 20 users not familiar with the system, and consisted of running user tests with:

- our proposed system with *user-logs* optimized distances
- the baseline system with *original* distances
- a random system that displays images randomly without replacement

The systems were set up for the same collection of 60,000 images from ImageNet dataset as in §5.4, and with the same indexing features, namely the bags of SIFT features provided by ImageNet. Thus, the original system is un-changed from the previous experiments.

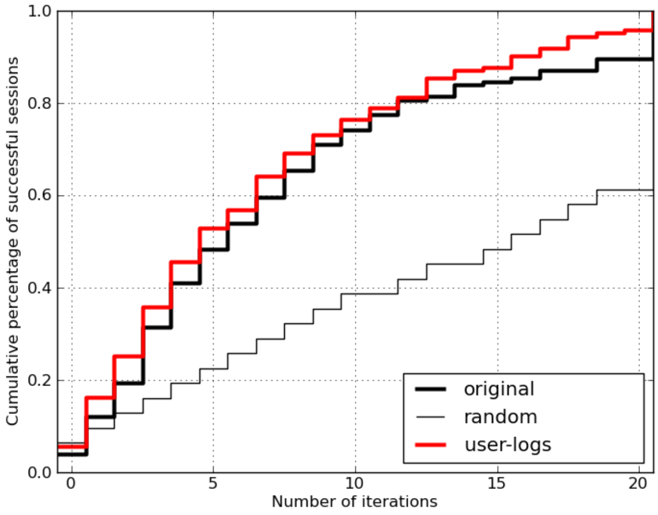
The parameters of the calibration functions in Figure 3.3 were also set in the same fashion, namely to saturate only after including on average 10% of the images in the collection. As we saw above in §6.4.1, the original and the optimized distances have very much alike statistical properties. Thus, the parameters of the calibration functions were set to the same values for both the original system and our proposed optimized system.

The evaluation scenario is preserved as well as much as possible. The users were asked to search for the same semantic targets as:

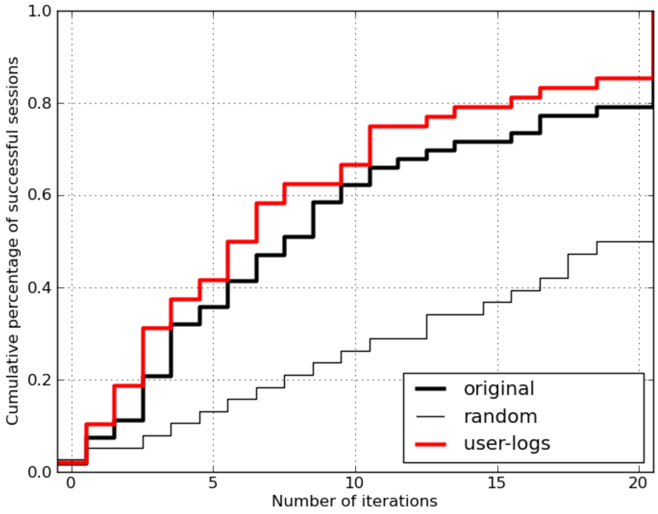
- portraits/close-ups of dogs, wolves
- electronic devices as laptop, mobile phone
- big boats as ferryboats, cargoes
- baskets/plates with fruits, vegetables
- furniture items as tables, chairs
- entrances/windows of shops, shopping centers

The only difference is that the users were told to end the searching sessions when they were satisfied by one single displayed image, instead of four images. We considered that this choice is more suitable for the generic evaluation of the adapted similarity metric.

Each user performed searching sessions corresponding to all combinations of systems and semantic targets, and thus our evaluation resulted in 120 sessions per system.



(a): Average performance of all the users



(b): Average performance of the demanding half of the users

Figure 6.5: Retrieval performance of the original framework in combination with the log-based similarity metric. Plot (a) shows the average performance of all the users for each of the assigned configurations. The optimized system performs consistently better than the original one, and saturates about 7% higher after 20 iterations. Plot (b) shows the average performance of the demanding half of the users. The relative improvement is even more accentuated.

	Precision ($t < 20$)		
	random	original	user logs
(a): all the users	0.61	0.90	0.97
(b): the demanding half of the users	0.50	0.79	0.87

Table 6.2: Retrieval performance. Here are a few discrete values read from Figures 6.5.

6.4.3 Results analysis

The user-based evaluation shows that the log-based similarity metric improves the retrieval performance. Next, we discuss the performance as the cumulative percentage of successful sessions per number of iterations.

Figure 6.5 shows that the optimized system performs consistently better than the original one. The optimized system saturates at 97% after 20 iterations, while the original one only reaches about 90%. This means that the log-based similarity metric is better aligned with the users' similarity judgments.

The average performance flattens out the differences between the users. As much as we tried to "calibrate" the users while instructing them, we observed a large variation in their tests. In order to get a better insight, we had the idea to divide the users in two equal sub-groups based on their level of demand: un-demanding and demanding users.

What we did is we considered the average number of iterations of each user test as an indication of the level of demand of the user. We computed the average number of iterations of each user test by averaging all the searching sessions done by that user, for all the configurations and for all the semantic targets. Then, we ordered the users accordingly to their average number of iterations, and we split them in two groups of equal sizes: un-demanding half and demanding half.

Figure 6.5.(b) shows the retrieval performance for the demanding half of the users. The random system has a much lower performance than in the case of all the users, and all three systems have a much lower success rate for the first display set at iteration 0. This quantify somehow the higher level of demand of the demanding users. The demanding users benefit even more from the optimized metric, and this is line with the intuitive reasoning.

6.5 Summary

This chapter focused on improving the retrieval capabilities of the original framework by modeling the user similarity judgments in long-term beyond the relevance feedback information given during a single searching session. We re-defined the image similarity metric as a weighted Euclidean distance over the image feature space, and we elaborated a technique to optimize it off-line based on the user logs.

Our approach has two major advantages. On the one hand, it exploits the user feedback that is acquired naturally during the searching sessions, and does not require any log acquisition campaigns. Even if the similarity models would be changed, the user logs can still be used in the same manner. On the other hand, it is generic and can leverage very large amounts of user logs. As the retrieval system is used, the user logs will gradually cover the entire collection, and the log-based similarity metric will systematically improve.

Our experiments give evidence that the optimization scheme we propose is beneficial for the retrieval performance of the original framework. On the one hand, it normalizes properly the weighted Euclidean distance to the calibration parameters. On the other hand, it succeeds to adapt the low-level indexing information in order to align it better with the users' similarity judgments.

The evaluation results give motivation for further investigations when a larger amount of user logs would become available. Eventually, this technique could be able to support personalized similarity metrics for each user separately.

7 Multi-modal image similarity

Multimedia collections often tend to include inter-related modalities, as for example photos and annotations in photo-sharing repositories, pictures and captions in news web-sites or x-ray scans and reports in medical databases. Intuitively, these inter-relationships could be exploited in order to provide a better indexing by compensating the weaknesses of the individual modalities.

In this chapter, we present a multi-modal extension of our retrieval framework for exploiting indexing features extracted from different modalities, as for example features extracted from both images visual content and their associated annotation keywords. We propose an adaptive similarity metric that weights dynamically, on-the-fly at each iteration, between the features of different modalities, depending on what the user is searching for.

The effectiveness of our multi-modal extension was assessed by 2 independent user-based evaluations with 30 users each on a subset of 35,000 images from the Corel stock photo library, for which we extracted SIFT-based visual features and LSA-based textual features. The system succeeds to retrieve images that satisfy the users in less than 5 iterations in 60% of the cases.

7.1 Introduction

A key characteristic of many multimedia collections is that data of different modalities are interrelated, as for example images and annotations in photo collections, songs and lyrics in music collections, or movies and moviescripts in video collections. The particular aim of the multi-modal research is to exploit the complementarity of the multi-modal information in order to minimize the semantic gap.

Our primary focus was to integrate visual-based and textual-based features. In the recent years, research confirmed that both visual-based and textual-based features have inherent limitations, and the retrieval systems are better off if they exploit both feature types in a multimodal fashion, in order to compensate each other for their own limitations [56].

The original approach uses a similarity metric based on visual features. The simplest approach towards multi-modality would be to simply concatenate the visual-based and textual-based features in order to obtain some *composite features*. A more desirable approach would be to extend the retrieval framework for more advanced approaches such as *dynamically weighted features* [68].

Our multi-modal extension relies on an adaptive similarity metric that weights dynamically, at each iteration, the individual metrics of different modalities. In the iterative process, the weighting is estimated depending on what the user is searching for via a Maximum Likelihood approach. This weighting motivated by the intuition that the retrieval needs are sometimes modeled better by visual features, sometimes by textual features, and sometimes by a combination of both.

The effectiveness of our multi-modal extension was assessed by 2 independent user-based evaluations with 30 users each on a subset of 35,000 images from the Corel stock photo library, for which we extracted SIFT-based visual content features and LSA-based textual features. We primarily investigate the integration of indexing features extracted from both the image's visual content and their accompanying annotation keywords. However, our approach can be applied straight-forward for any other types of features and for any larger number of features.

7.2 State-of-the-art

The recent evolution of multimedia collections towards including inter-related modalities motivates the research of retrieval systems that are able to exploit multiple types of information into a unified framework (i.e. multimodal retrieval) [15]. In line with this trend, we started to investigate how to extend the retrieval framework in order to integrate multi-modal indexing features.

The original approach uses a rigid similarity metric based on low-level features extracted from the visual content of images (i.e. global descriptors of color, texture and shape). As already stated in the previous chapter §6.2, there is no such thing as an omnipotent similarity metric, and research proposes various adaptations and compensations in order to minimize the semantic gap. Learning from the user is good, but it still needs to have good features to start with. In this regard, the adaptive multi-modal metrics are an alternative that show promising potential [56].

In our system, the visual features are based on SIFT (Scale Invariant Feature Transform) [41]. SIFT feature vectors are highly distinctive and robust to affine transformations, changes in illumination and limited changes in 3D viewpoint. The textual features are based on LSA (latent semantic analysis) [17]. LSA takes advantage of the implicit associations between keywords, and it escapes the unreliability, ambiguity and redundancy of individual keywords.

7.3 Joint visual and textual-based metric

We propose an extension that exploits indexing features extracted from both the visual content and the annotation keywords of images. Moreover, we propose an adaptive similarity metric that weights dynamically, on-the-fly at each iteration, between the visual-based and textual-based features.

Our approach uses features extracted from both visual content and annotation keywords of images. The visual-based features are derived using SIFT [41] and the textual-based features are derived through LSA [17]. Our particular choices are specified in §7.4.1. Then, for every two images $k, l \in \Omega$, the visual-based distances $d_{\mathsf{V}}(k,l)$, and the textual-based distances $d_{\mathsf{T}}(k,l)$ are obtained as Euclidean distances between the corresponding feature vectors.

7.3.1 Bi-modal adaptive metric

For each modality separately, the corresponding distances are calibrated with monotonous functions, one for ϕ^+ and one for ϕ^- as shown in Figure 7.1. They are meant to normalize the corresponding distances, and also to compensate for the partial match between the distances and the user subjective perception of image similarities, in a similar manner as in the original framework and under the same justifications as in [24].

ϕ^+ and ϕ^- in Equations (3.12-3.13) are defined as a weighted sum of both visual-based and textual-based distances:

$$\phi^+(k,x) = w \cdot \phi_{\mathsf{V}}^+(d_{\mathsf{V}}(k,x)) + (1-w) \cdot \phi_{\mathsf{T}}^+(d_{\mathsf{T}}(k,x)), \tag{7.1}$$

$$\phi^-(k,x) = w \cdot \phi_{\mathsf{V}}^-(d_{\mathsf{V}}(k,x)) + (1-w) \cdot \phi_{\mathsf{T}}^-(d_{\mathsf{T}}(k,x)). \tag{7.2}$$

This is motivated by the intuition that the retrieval objectives as well as the subjective perception of image similarities are sometimes modeled better by visual features, sometimes by textual features, and sometimes by a combination of both.

Table 7.1: Notation

w_t	weighting parameter learned on-the-fly at each iteration t
$d_{\mathsf{V/T}}$	similarity metrics for each modality separately
$\phi_{\mathsf{V/T}}^{+/-}$	calibration functions for each modality separately

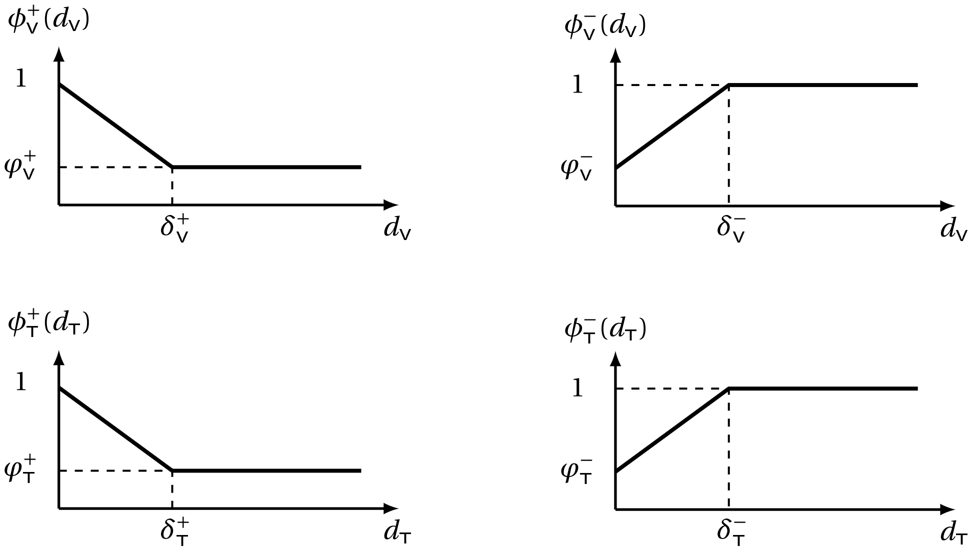


Figure 7.1: Calibration functions. For each modality separately, the corresponding calibration functions normalize the corresponding distances, and also aim to compensate for the partial match between the distances and the user subjective perception of image similarities, under the same justifications as in [24].

7.3.2 Weight estimation

In the first iteration $t = 0$, both distance types are equally weighted by setting w_0 in Equations (7.1-7.2) to 0.5. In the subsequent iterations, the weighting parameter w_t is estimated based on a Maximum Likelihood approach:

$$w_{t+1}^{*} = \arg\max_{w\in[0,1]} \frac{p_t(x_t^{*})}{\sum_{x\in D_t} p_t(x)}. \tag{7.3}$$

Immediately after the relevance feedback event $\{D_t,\ x_t^{*}\}$, before updating the probabilities $p_{t+1}(k)$ for all $k \in \Omega$, the probabilities $p_t(x)$ for the displayed images $x \in D_t$ are re-estimated for a few discrete weight values. The optimal value w_{t+1}^{*} is the weight that distinguishes the most x_t^{*} from all $x \in D_t$, and in consequence will make the most out of the relevance feedback event $\{D_t,\ x_t^{*}\}$. In our experiments, we considered 11 discrete values of $w \in \{0, 0.1, \ldots, 1\}$.

7.4 Experimental results

Our evaluation has been conducted with 2 groups of 30 users not familiar with the system. The evaluation does not rely on any apriori defined ground truth. Instead, it relies on comparing four configurations:

- *bimodal-adaptive* is weighting dynamically, at each iteration, between the visual-based and textual-based features as described in §3.4.

- *unimodal-visual* is a particular case obtained by setting w to 1.
- *unimodal-textual* is a particular case obtained by setting w to 0.
- *pure-sampling* is a special configuration in which w is set to 0.5 and the probabilities $p_t(k)$ are fixed to 0.5 and never updated. Basically, *pure-sampling* uses the similarity metric in order to run the Voronoi tessellation algorithm, but discards the relevance feedback. Thus, it provides a fair base-line for showing the real contribution of the relevance feedback itself.

7.4.1 System setup

The system was set up for a subset of 35,000 photos from the Corel stock photo library. Each image is associated with 5-7 keywords from a vocabulary of about 5,000 keywords [45, 53]. For further insight, one can consult §C.1.

For the visual-based metric, SIFT features [41] were extracted for each image by detecting points of interest at 4 scales (i.e. this resulted in 50-300 features per image). A subset of 300,000 features was chosen randomly and clustered in 500 classes with the K-means algorithm. A reference SIFT vocabulary was formed by the resulted centroids. Then, a histogram-like feature vector was derived for each image by computing the membership of its own SIFT features to the centroids in the SIFT vocabulary.

For the textual-based metric, the Boolean image–keyword matrix was created by considering a vocabulary of about 5,000 keywords, all the keywords that were used to annotate at least 3 images. Then, LSA was applied as explained in [17] to obtain vector representations of dimension 500.

7.4.2 Evaluation scenario

Each user group was assigned with three configurations: the first group with *bimodal-adaptive*, *unimodal-visual* and *pure-sampling*; the second group with *bimodal-adaptive*, *unimodal-*

Table 7.2: The 12 semantic categories described only in words.

bird/birds on water	airplane in the sky
sailing boat/boats	forest landscape
people doing sport	waterfall
city panorama	garden with flowers
animals in the wild	historical site
people on the streets	sandy beach

visual and *unimodal-textual*.

The evaluations follow the scenario in §3.5.2. In order to ensure sufficient diversity and comparable difficulty, there were 12 semantic categories described only in words, as they are mentioned in Table 7.2.

Each user was asked to perform one searching session for each semantic category, thus 12 sessions in total. The interpretation of the semantic category in the sense of visual content was left to the user. The users were only told to end the session when they were satisfied by at least one image. The evaluation interface is shown in Figure A.3. One third of the users was available to perform 36 searching sessions in total, one for each configuration and each semantic category.

7.4.3 Results analysis

Our evaluation shows that the approach is viable. We can see in Figure 7.2.(a) that all configurations using relevance feedback perform consistently better than *pure-sampling*. This means that the system is intuitive and able to deal with the user subjectivity in making similarity judgments.

Tables 7.5-7.6 tell about the statistical significance of the evaluation. For each couple of configurations, we counted how many times one configuration performed better than the other for the same user and the same semantic category, whenever there was available data. Then, we computed the binomial probabilities. In principle, a difference is statistical significant if the corresponding probability is smaller than 0.05.

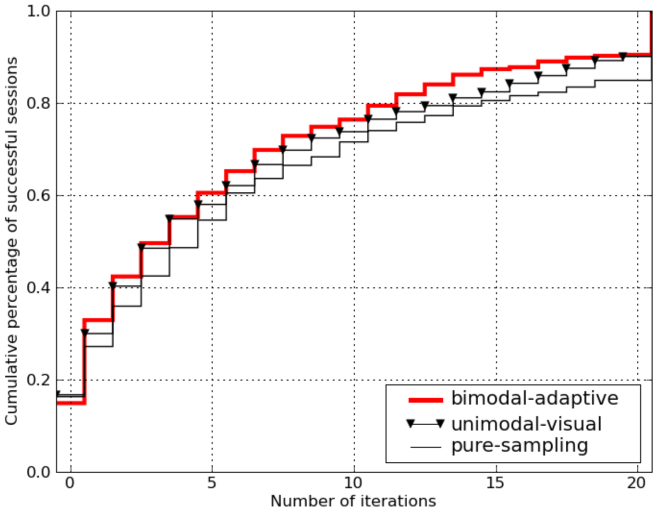
By adding textual features, *bimodal-adaptive* and *unimodal-textual* perform significantly better than *unimodal-visual* in Figure 7.3.(a). *Bimodal-adaptive* and *unimodal-textual* are not significantly different as we can see in Table 7.6. Since the semantic categories were specified textually, it is likely that the textual features were favored. In other contexts, visual features may become prevalent. Further evaluations should definitely address this issue.

These total averages flatten out the differences between users. Following the same reasoning as in the previous chapter §6.4.3, one can divide the users in two equal sub-groups based on their performance over all configurations: un-demanding and demanding users. Figures 7.3.(a-b) show how the demanding users benefit from *bimodal-adaptive*.

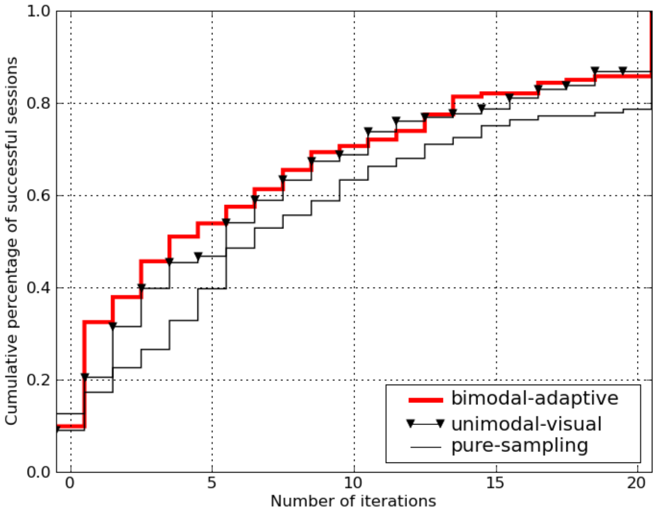
About 60% of the sessions are successfully terminated in less than 5 iterations, and 80% in less than 10 iterations. The retrieval performance remains very reasonable when thinking of the two most extreme cases. If the collection would be arranged as a tree with 8 branches at each node, the *perfectly-structured* search will need about 3 iterations in average and $\log_8 \|\Omega\| \approx 5$ iterations at maximum.[1] If the collection would be totally unstructured, the *uniformly-random* search will need $\|\Omega\| / (\|D\| \cdot (L+1)) \approx 12$ iterations in average and a lot more at maximum.

---

[1] $\|\Omega\| \approx 35{,}000$, $\|D\| = 8$, $L \approx 350$ are the sizes of the image collection, the display set, and the semantic category.



(a): Average performance of the first group

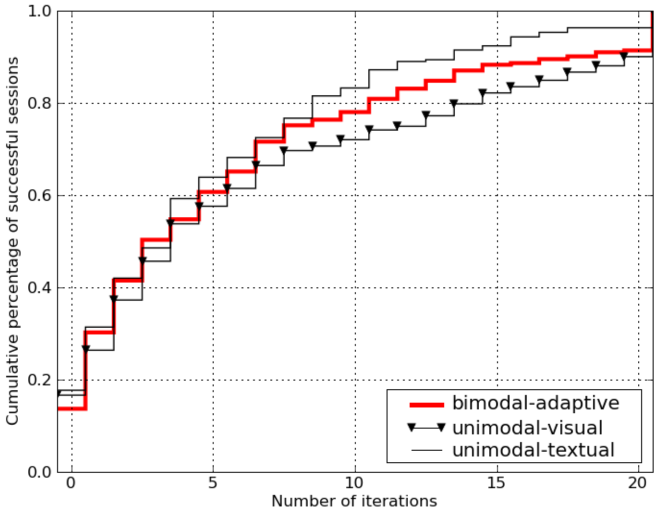


(b): Average performance of the demanding half of the first group

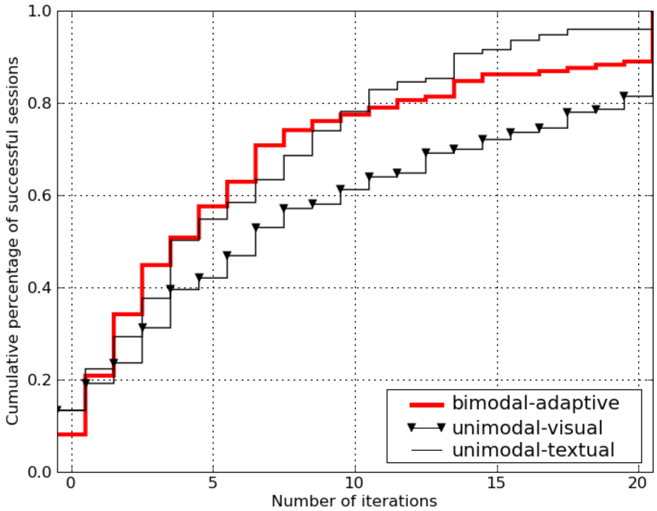
Figure 7.2: Retrieval performance as the cumulative percentage of successful sessions per number of iterations. Plot (a) shows the average performance of the first group for each of the assigned configurations. Plot (b) shows the average performance of the demanding half of the group.

	Precision ($t < 5$)	**Precision** ($t < 10$)	**Precision** ($t < 15$)
(a): all the users of the first group	0.54/0.58/0.60	0.73/0.75/0.77	0.80/0.82/0.88
(b): the demanding users of the first group	0.40/0.48/0.53	0.63/0.69/0.70	0.75/0.79/0.82

Table 7.3: Retrieval performance. Here are a few discrete values read from Figure 7.2.



(a): Average performance of the second group



(b): Average performance of the demanding half of the second group

Figure 7.3: Retrieval performance as the cumulative percentage of successful sessions per number of iterations. Plot (a) shows the average performance of the second group for each of the assigned configurations. Plot (b) shows the average performance of the demanding half of the group.

	Precision ($t < 5$)	**Precision** ($t < 10$)	**Precision** ($t < 15$)
(a): all the users of the second group	0.55/0.58/0.60	0.72/0.74/0.76	0.80/0.82/0.88
(b): the demanding users of the second group	0.40/0.48/0.54	0.63/0.69/0.70	0.75/0.79/0.82

Table 7.4: Retrieval performance. Here are a few discrete values read from Figure 7.3.

adaptive/visual
(117/207) 0.025
adaptive/pure-sampling
(130/217) 0.002
visual/pure-sampling
(127/210) 0.001

(a): all the users of
the first group

adaptive/visual
(61/99) 0.007
adaptive/pure-sampling
(66/108) 0.008
visual/pure-sampling
(62/101) 0.007

(b): the demanding users
of the first group

Table 7.5: Binomial-test for statistical significance corresponding to the experiments in Figure 7.2. For example, for all the users the first group, *bimodal-adaptive* performed better than *unimodal-visual* in 117 times out of 207, and the probability of this to occur by chance is 0.025.

adaptive/visual
(75/132) 0.048
adaptive/textual
(72/133) 0.149
textual/visual
(76/133) 0.041

(a): all the users of
the second group

adaptive/visual
(52/84) 0.010
adaptive/textual
(43/77) 0.128
textual/visual
(43/73) 0.050

(b): the demanding users
of the second group

Table 7.6: Binomial-test for statistical significance corresponding to the experiments in Figure 7.3. For example, for all the users in the second group, *bimodal-adaptive* performed better than *unimodal-visual* in 75 times out of 132, and the probability of this to occur by chance is 0.048.

7.5 Summary

We have presented a multi-modal extension of the retrieval framework for exploiting indexing features extracted from different modalities. This extension relies on an adaptive similarity metric that weights dynamically, at each iteration, the individual metrics of different modalities, depending on what the user is searching for.

We have primarily exploited indexing features extracted from both images visual content and their associated annotation keywords. This bi-modality choice was motivated by the intuition that the visual-based and textual-based features are complementary to each other, and the retrieval needs are sometimes modeled better by visual features, sometimes by textual features, and sometimes by a combination of both.

The evaluation results give motivation for further investigations on how the system could benefit from other indexing features and similarity metrics. Although evaluated for the bi-modal case with one visual-based and one textual-based feature types, our extension is ready to be applied for the multi-modal case with only minor changes.

8 System integration

We have presented so far, in the previous Chapters §4-7, four contributions that are complementary to each other, and touch different components of the retrieval system, namely the large-scale HEAT framework, the exploration/exploitation trade-off, the log-based similarity learning, and the multi-modal similarity metric.

In this chapter, we investigate the integration of all contributions together into one comprehensive retrieval system. From the retrieval performance point of view their integration makes a lot of sense, and it is fair to expect their integration to be beneficial.

We organized user-based evaluation campaigns in the same manner as for each individual contribution, and we evaluated systematically different combinations of our contributions. We got evidence that each contribution complements each other consistently for both small and large collections. Then, we got evidence that the overall retrieval performance of the comprehensive retrieval system is also consistently beneficial.

8.1 Integration overview

We rounded up our research by investigating the integration of all our contributions together into one comprehensive retrieval system. They are complementary to each other, and touch different components of the retrieval system. Thus, from the retrieval performance point of view their integration makes a lot of sense.

Still, we have to support our intuition with some evidence that each contribution complements each other, and that their combination is consistently beneficial. Therefore, we identified a few intermediate milestones towards the integration of all contributions together into one comprehensive retrieval system.

Mass-zoom system with log-based similarity metric

The integration of the log-based similarity metric in the mass-zoom system is straight-forward. Intuitively, it is fair to expect that the two contributions complements each other. On the one hand, the log-based similarity metric models better the human perception of image similarity, and thus it can only make the relevance feedback more reliable. On the other hand, the mass-zoom system takes advantage of the consistency in the relevance feedback.

HEAT framework with mass-zoom extension

The integration of the mass-zoom extension in the HEAT framework is also straight-forward from the technical point of view. We integrated them as such, but we are aware that the heuristics employed by the algorithmic components should be re-investigated in the new context of integration. The criteria of trace refinement may be impeded by a very small mass-zoom factor, but this remains an open-question for future work.

HEAT framework with log-based similarity metric

The integration of the log-based similarity metric and the HEAT system impacts not only the on-the-fly models but also the quality of the pre-computed hierarchical organization of the collection. Since the log-based similarity metric models better the human perception of image similarity, the hierarchical organization will be also reflecting better the human perception. Intuitively, it is fair to expect that the HEAT system benefits from log-based similarity metric even more than the original system.

HEAT framework with multi-modal similarity metric

The integration of our adaptive multi-modal contribution and the HEAT framework remains an open-question for future work, as there is still one issue to tackle from the technical point of view. Currently, the large-scale HEAT framework is based on a pre-computed hierarchical organization of the collection that assumes an invariant similarity metric. The HEAT system works with a fixed pre-computed organization that is suitable for any fixed weighting of the multi-modal indexing features. Unfortunately, the HEAT system cannot integrate simply the dynamical weighting that adapts at each relevance feedback iteration.

8.2 Experimental results

We aim to evaluate the overall performance of the comprehensive system that integrates all our contributions. For this, we will first evaluate systematically different combinations of our contributions, in order to get evidence that each contribution complements each other, and that their combination performs consistently for both small and large collections. Then, we will do the experiments and the analysis for the final evaluation.

8.2.1 System setup

We set up the experiments for the two collections from ImageNet dataset that we used to evaluate each individual contribution, namely the small collection of 60,000 images and the large collection of 1,000,000 images. We also take over all the calibration settings and the log-based similarity metric as they were derived for each individual contribution in the previous chapters.

Following the remark in §8.1, we set up multiple instances of the HEAT system with the corresponding hierarchical organizations for the different similarity metrics, one for the original Euclidean metric and one for the log-based similarity metric.

Thus we ended up with:

- the baseline system with *original* distances
- the baseline system with *user-logs* optimized distances
- the mass-zoom system with *user-logs* optimized distances

and also with the large-scale variants:

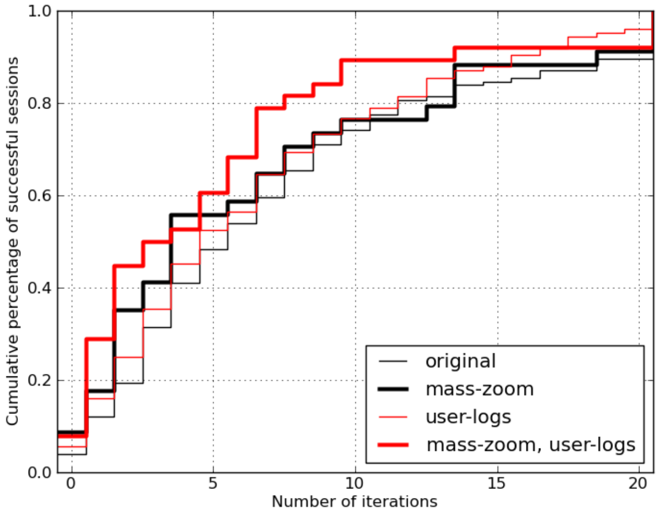
- the HEAT system with *original* distances
- the HEAT system with *user-logs* optimized distances
- the HEAT system with mass-zoom and *user-logs* optimized distances

8.2.2 Evaluation scenario

We organized several user-based evaluation campaigns based on the same evaluation scenario as in our previous campaigns. The scenario details are in §3.5.2. The evaluations were conducted with 20 users, and each user performed searching sessions corresponding to all combinations of systems and semantic targets, Thus, our evaluation resulted in 120 sessions per system.

8.2.3 Results analysis

Next we discuss the outcome of our evaluation campaigns. We first organized two evaluation campaigns to evaluate two partial combinations of our contributions, and then we organized one final evaluation campaign to evaluate the total integration of our contributions.



60K image collection

Figure 8.1: Retrieval performance of the integration of mass-zoom system and the log-based similarity metric with the 60K image collection. Each of our contributions taken individually, namely the mass-zoom system and the log-based similarity metric, improves the retrieval performance of the original system. Furthermore, the integration of the mass-zoom system and the log-based similarity metric is beneficial. The two contributions complement each other, and their combination significantly improves the overall performance.

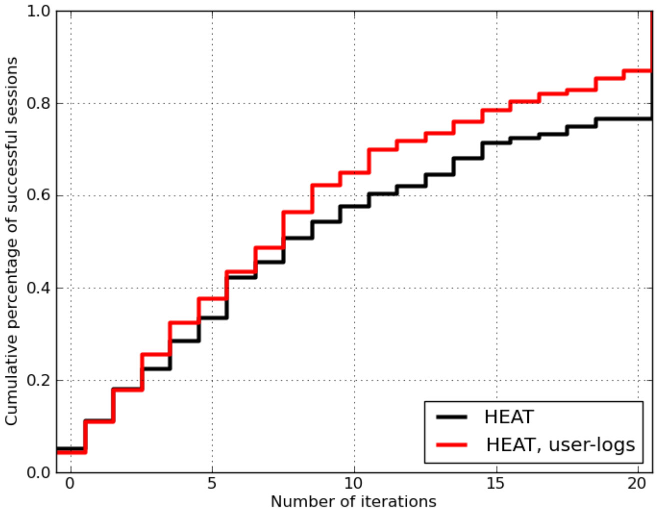
Mass-zoom system with log-based similarity metric

Figure 8.1 shows the retrieval performance of the mass-zoom system in combination with the log-based similarity metric. Experiments were conducted only with the small collection of 60,000 images, since it cannot cope with the large-scale collection. The integration of the two contributions is compared with each contribution taken individually, and then all are compared with the original system.

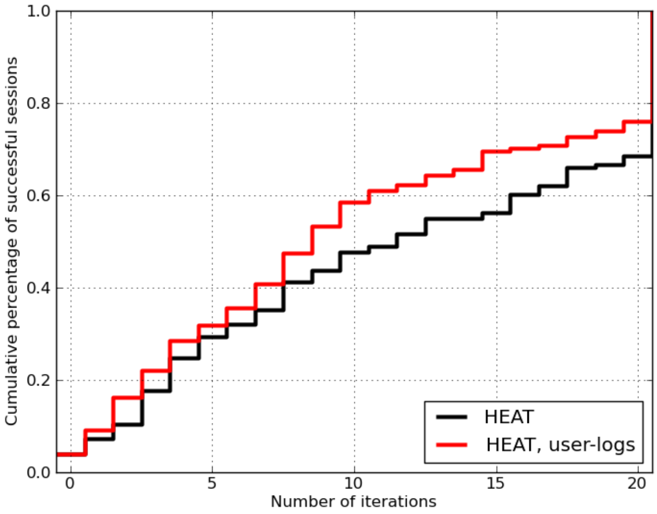
The evaluation shows that each contribution taken individually improves the retrieval performance of the original system, and this is re-enforce the results of our previous evaluations. Furthermore, the integration of the mass-zoom system and the log-based similarity metric is beneficial. The two contributions complement each other, and their combination significantly improves the overall performance. The optimized system provides 80% rate of success in less than 8 iterations, while the original system reaches the same rate only after 13 iterations, which is 5 iterations more.

HEAT framework with log-based similarity metric

Figures 8.2.(a-b) show the retrieval performance of the HEAT systems in combination with the log-based similarity metric. For both small and large collections, we can see that the integration of the HEAT system with the optimized metric is beneficial. In fact, the HEAT



(a): 60K image collection

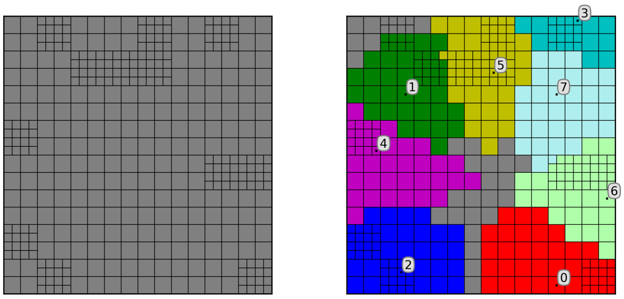


(b): 1M image collection

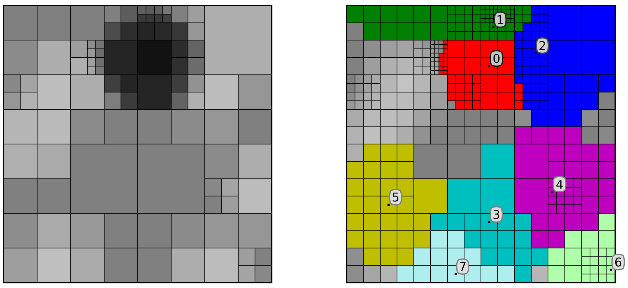
Figure 8.2: Retrieval performance of the HEAT framework in combination with the log-based similarity metric. Plot (a) shows the performance of HEAT for the small collection of 60,000 images and plot (b) for the large collection of 1,000,000 images. The integration of the HEAT system with the log-based similarity metric is beneficial. The HEAT system benefits even more than the original system since the log-based similarity metric improves not only the on-the-fly models, but also the quality of the pre-computed hierarchical organization of the collection.

	Precision ($t < 20$)	
	HEAT	HEAT, user-logs
(a): 60K image collection	0.78	0.88
(b): 1M image collection	0.68	0.77

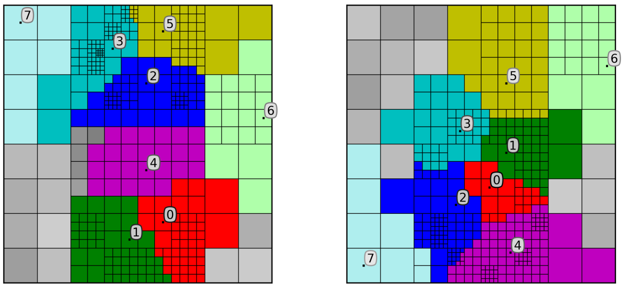
Table 8.1: Retrieval performance. Here are a few discrete values read from Figure 8.2.



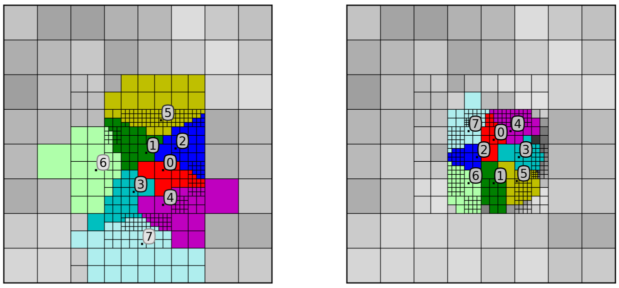
$t = 0$ (initial) $t = 0$ (expand)



$t = 1$ (collapse) $t = 1$ (expand)



$t = 2$ (expand) $t = 3$ (expand)



$t = 4$ (expand) $t = 5$ (expand)

Figure 8.3: Evolution of the comprehensive system for the synthetic collection, when searching for images with points close to the center. At iteration 0, the trace is initialized randomly. At each iteration, the zoom factor is estimated, the trace is collapsed and expanded, the probabilities of relevance are updated, and then the new images to be shown are selected. After 5 iterations, the trace concentrates mostly on the intended region.

system benefits even more than the original system since the log-based similarity metric improves not only the on-the-fly models, but also the quality of the pre-computed hierarchical organization of the collection as pointed out in §8.1.

Table 8.1 has a few discrete values read from Figures 8.2.(a-b) that characterize the retrieval precision after 20 iterations of the HEAT vs. optimized systems. The optimized system performs consistently better, and saturates about 10% higher.

HEAT framework with mass-zoom extension

The retrieval performance of the HEAT system in combination with the mass-zoom extension is shown in Figure 8.4 among other system combinations. We can see that their integration improves their individual performances for both small and large collections, which means that they complement each other as well.

Total integration

For an intuitive illustration of the system behavior, we set up the comprehensive system for the synthetic collection, and we take again the case of searching for images with points close to the center. Figure 8.3 shows how the comprehensive system evolves at each iteration, and how the image collection is sampled at different resolutions in different regions in combination with the mass-zoom extension.

The retrieval performance of the comprehensive system that integrates all our contributions is shown in Figure 8.4. Both individual integrations of the mass-zoom extension and the log-based similarity metric improve the retrieval performance of the HEAT system. Furthermore, they complement each other, and their combination significantly improves the overall retrieval performance.

The retrieval performance of the overall system is further analyzed in Figure 8.5. Here the retrieval performance of the overall system is evaluated for three different trace sizes. We can see how the retrieval performance depends on the trace size. The bigger the trace, the better the performance, but the difference between 1,000 vs. 1,500 is smaller than the difference between 500 vs. 1,000. Regarding the scalability properties, we can see that the retrieval performances for the large collection remain intuitively consistent.

Overall, about 50% of the sessions are successfully terminated in less than 5 iterations, and 80% in less than 15 iterations. The system performance remains very reasonable when thinking of the two most extreme cases. In the ideal case, if the collection would be arranged as a tree with 8 branches at each node, the *perfectly-structured* search will need about 3 iterations in average and $\log_8 \|\Omega\| \approx 5$ iterations at maximum.[1] In the worst case, if the collection would be totally unstructured, the *uniformly-random* search will need $\|\Omega\| / (\|D\| \cdot (L+1)) \approx 12$ iterations

---

[1] $\|\Omega\| \approx 60,000$, $\|D\| = 8$, $L \approx 600$ are the sizes of the image collection, the display set, and the semantic category.

in average and $\|\Omega\| / \|D\| - \lceil L / \|D\| \rceil \gg 100$ iterations at maximum.

The computational effort of the overall retrieval system is shown in Figures 8.6(a-b). Here, we have the computational cost of three HEAT systems with different trace sizes, for which we show the system response timing in seconds as the users experienced it during the evaluations. We can see how the computational effort of the HEAT system depends on the trace size, and has roughly $\mathscr{O}(\|\mathscr{T}\| \cdot \log \|\mathscr{T}\|)$ complexity.

As visible in the plots, the computation is higher in the first iterations and this is due to the intensive operations of refining the trace, namely the collapse/expansion operations. In the later iterations, the trace becomes relatively stable, and there are fewer collapse/expansion operations. Still, the computation increases slowly because the computation from scratch of the probabilities of relevance increases with the number of iterations.

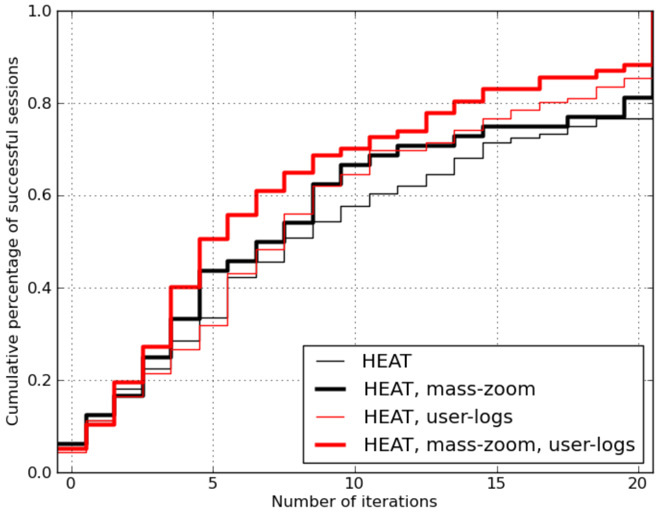
Regarding the scalability properties, we can see that the timings for the large collection shown in Figure 8.6.(b) remain comparable with the ones for the small collection. The computational effort of our system is decoupled from the collection size, and depends mainly on the trace size.

8.2.4 Informal discussions

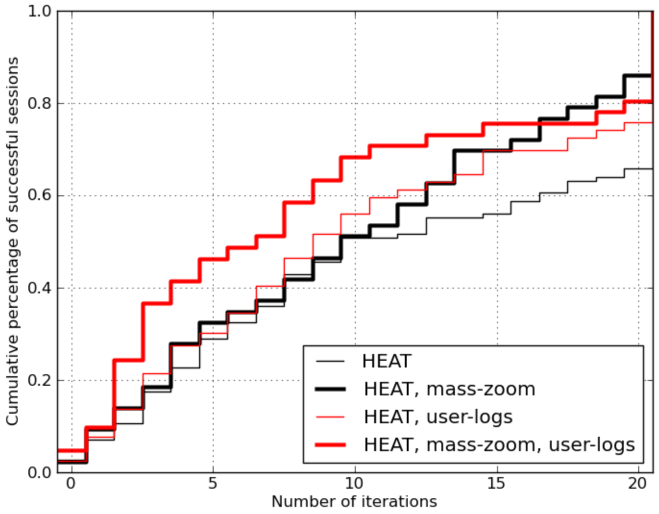
Although we did not organize an appraisal questionnaire, we received favorable informal feedback regarding the user experience. The system is unconventional but intuitive, and becomes understood in very short time, even in the first searching session. All users confirmed that the kind of similarity judgments required by the system seems natural. Of course beyond this friendly feedback, we are aware that there is still plenty of room for improvement in coping with the semantic gap of the state-of-the-art similarity metrics.

Suggestions have been made to improve the user experience. In the first couple of iterations, it may happen that none of the displayed images is even vaguely related to what the user is searching for. When the users cannot make reliable similarity judgments, they would rather give *negative feedback* (i.e. none of the images resembles what they are searching for) or, at least, give *no feedback* and just ask for new images. Also, the users would appreciate the possibility to *undo* the last relevance feedback iteration. Such functionalities may be easily integrated in our approach, but they were intentionally not supported in the evaluation scenario.

Although this approach is viable on its own, many users suggested to integrate it in a retrieval pipeline as one of the steps to narrow down the retrieval scope. This can be very well an alternative option. Instead of starting from a heuristic sampling of the collection, the iterative relevance feedback could be offered after an initialization stage via *query-by-keywords*, *query-by-visual-examples*, or any other type of query. Again, such functionality may be easily integrated in our approach, and it makes a lot of sense to further evaluate these mixed retrieval scenarios.



(a): 60K image collection

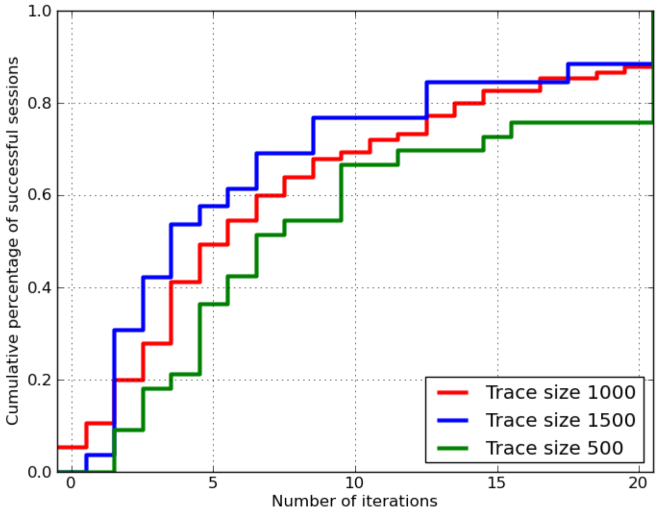


(b): 1M image collection

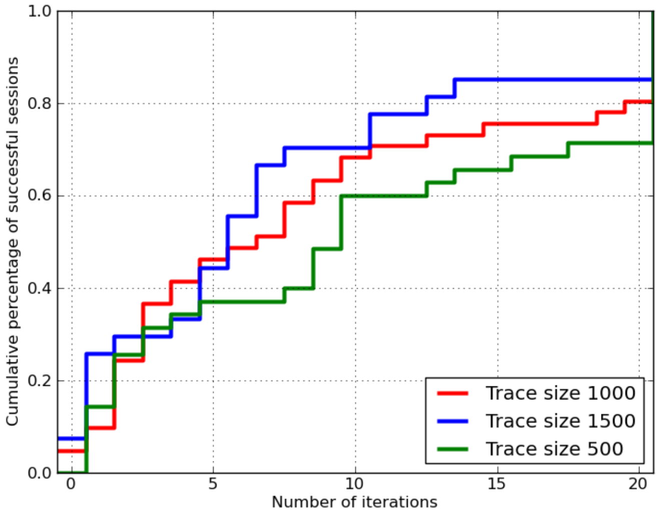
Figure 8.4: Retrieval performance of the comprehensive system that integrates all our contributions. Each of our contributions taken individually, namely the mass-zoom extension and the log-based similarity metric, improves the retrieval performance of the HEAT system. The contributions complement each other, and their combination significantly improves the overall retrieval performance. Furthermore, the behavior of the system is stable and consistent for both small and large collections.

	Precision ($t < 5$)	**Precision** ($t < 10$)	**Precision** ($t < 15$)
(a): 60K image collection	0.42/0.50	0.58/0.61	0.75/0.82
(b): 1M image collection	0.32/0.46	0.51/0.68	0.70/0.75

Table 8.2: Retrieval performance. Here are a few discrete values read from Figure 8.4.



(a): 60K image collection

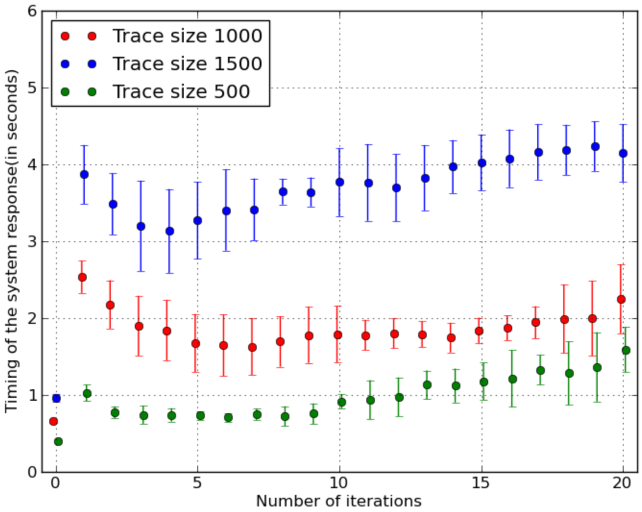


(b): 1M image collection

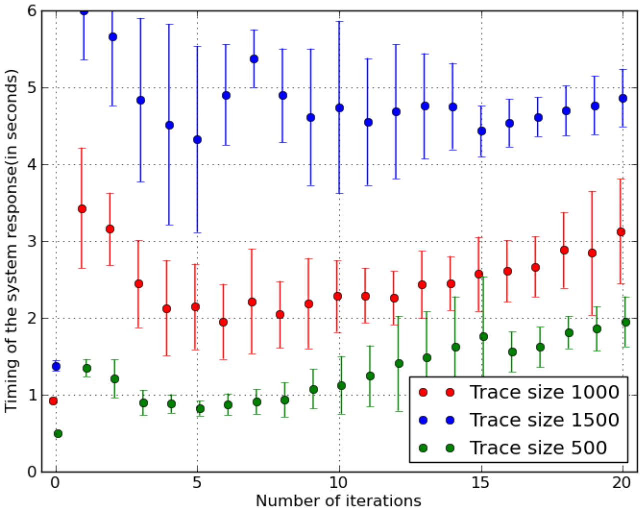
Figure 8.5: Retrieval performance of the comprehensive system for three different trace sizes. The average performances for the small collection are shown in (a): We can see how the retrieval performance depends on the trace size. The bigger the trace, the better the performance, but the difference between 1,000 vs. 1,500 is smaller than the difference between 500 vs. 1,000. The average performances for the large collection are shown in (b): The performances remain intuitively consistent.

Precision	$t < 5$	$t < 10$	$t < 15$
(a): 60K image collection	0.37/0.48/0.59	0.60/0.70/0.77	0.72/0.82/0.83
(b): 1M image collection	0.38/0.44/0.45	0.60/0.68/0.70	0.64/0.76/0.84

Table 8.3: Retrieval performance. Here are a few discrete values read from Figure 8.5.



(a): 60K image collection



(b): 1M image collection

Figure 8.6: Timing of the HEAT system responses (in seconds) as the users experienced them during the evaluations. Here, we compare the computational cost of three HEAT systems with different trace sizes. The timings for the small collection are shown in (a): The computational effort of the HEAT system depends on the trace size, and has roughly $\mathcal{O}(\|\mathcal{T}\| \cdot \log \|\mathcal{T}\|)$ complexity. As visible in the plot, the computation is higher in the first iterations, and this is due to the intensive collapse/expansion operations. In the later iterations, the trace is relatively more stable, but the computation increases slowly with the number of iterations due to the computation from scratch of the probabilities of relevance. The timings for the large collection are shown in (b): The timings remain comparable with the ones for the small collection. The computational effort of our system is decoupled from the collection size, and depends mainly on the trace size.

8.3 Summary

In this chapter, we rounded up our research by integrating three of our contributions together into one comprehensive retrieval system, namely the large-scale HEAT framework, the mass-zoom extension and the log-based image similarity metric. The integration of our adaptive multi-modal contribution remains an open-question for future work.

We evaluated systematically different combinations of these three contributions in the same manner as for each individual contribution, and we got evidence that each contribution complements each other. Finally, we evaluated the retrieval performance of the comprehensive retrieval system, and we shown empirically that the overall integration of our contributions is consistently beneficial.

9 Conclusion and foresight

We have started the work presented in this thesis with the overall purpose of exploring novel ways for an efficient, effective and interactive access to large-scale image collections. We have investigated a query-free retrieval approach that promises an interactive access to image collections of unprecedented size.

In this chapter, we summarize our work, and open new directions for further research. First, we conclude the thesis with a final overview of our contributions, and give along our final remarks. Then, we outline and motivate a few potential directions for future work.

9.1 General summary

The overall goal of our work was to achieve a better understanding of the research field of content-based image retrieval. Furthermore, making use of this understanding, we wanted to identify potential ways for an efficient, effective and interactive access to large-scale image collections.

We advocate the assumption that large-scale collections are not only large, but also inherently un-structured (i.e. lacking any semantic or thematic indexing as in the archived libraries) and continuously out-dated (i.e. images are frequently being added, replaced or removed). Thus, our research affinity was towards retrieval solutions that would potentially accommodate such realistic assumptions. On the one hand, the retrieval solutions should be computationally scalable in both off-line and on-the-fly operations. On the other hand, the indexing information should support incremental updates, without requiring updates from scratch each time something changes.

Our literature analysis guided us towards the iterative relevance feedback mechanisms, and in this thesis we have been investigating a query-free retrieval approach, which relies solely on an iterative relevance feedback mechanism driven by user subjective perception of image similarities. Most of the image retrieval approaches require an initial query before offering relevance feedback tools. The motivation for a query-free retrieval approach comes from the observation that formulating a query might not be the most optimal way of initializing a searching session. User retrieval needs are often difficult to describe in terms of keywords, and relevant images may be easily filtered out.

Our research focused on extending and reshaping various aspects of the relevance feedback mechanism. We have extended the state-of-the-art approach in four complementary aspects: large-scale distributed system architecture, exploration/exploitation trade-off, log-based similarity learning and adaptive multi-modal similarity metric. The user-based evaluations and the informal discussions show that our query-free retrieval system is viable, and has the potential of becoming a commercial application.

9.2 Further research

While a number of novel contributions are made, they are in no way complete solutions. The problems we deal with are relatively open-ended, with a lot of scope for incremental improvement, or even for adopting radically new approaches. Here, we point out a few alternative research directions that we have considered for exploring further, and we wished to work on if time would have allowed us.

Large-scale HEAT framework

The retrieval performance depends greatly on the quality of the hierarchical partitioning. Further research should address the means and criteria for controlling the trade-off between the retrieval performance and the computational effort. Currently, the criterion of collapsing/-expanding operations is based on the children nodes of the next inferior level.

There could be envisioned more sound statistical approaches based on other parameters of the nodes as for example the maximum similarity distance between the images in the node or the concentration of the images. Also, the implementation of efficient updating schemes on parallel and distributed computing architectures would be interesting as well.

Exploration/exploitation trade-off

The heuristics that we used for evaluating the exploration/exploitation trade-off could be questioned, and seen from different perspectives. For example, one could solve the same problem by re-thinking the original algorithm to somehow construct a finer grained Voronoi tessellation, with the granularity determined by the exploration/exploitation trade-off.

One could also investigate the idea of never displaying the image with the highest probability and none of the images in its Voronoi cell. In this way, the system will be pushed a bit more towards exploration, and will never enter a dead-end state where it just displays very close images one after the other. Of course, the size of the display set could also be played with, even in a dynamic way, on-the-fly at each iteration, during the retrieval process.

Log-based similarity metric

The evaluation results give motivation for further investigations when a larger amount of user logs would become available. On the one hand, one could investigate the effect of adapting the metric in the case of leveraging large amounts of user logs, where the efficiency of the optimized metric may fade due to the huge variation among the users. On the other hand, one could investigate the alternative of having personalized similarity metrics for each user separately.

Of course other optimization schemes and other types of adaptive metrics are always worth of investigating. One can think of metrics that are created not via feature weights optimization, but via deriving directly new better image features. As the retrieval system is used, the user logs will gradually cover the entire collection, and the user log-derived image features would gradually get control over the automatically-generated image features.

Multi-modal similarity metric

The system could benefit from other indexing features and similarity metrics. Although evaluated for the bi-modal case with only one type of visual-based and textual-based features, our extension is ready to be applied for the multi-modal case with only minor changes.

For the visual features, one should consider the generic MPEG-7 descriptors [42], as the MPEG-7 standards are largely accepted and used in the research community. For the textual features, one should consider at least some weighting methods such as tf-idf (i.e. term frequency - inverse document frequency), and even better some more advanced methods based on the WordNet semantic hierarchy.

System integration

The integration of our adaptive multi-modal contribution and the large-scale HEAT framework is still an open-question. The HEAT system cannot simply integrate the dynamical weighting since it is based on a pre-computed hierarchical organization of the collection that assumes an invariant similarity metric.

One possible solution is to consider several discrete weightings that covers fairly the weighting dynamic range, and then to have several pre-computed hierarchical organizations corresponding to these pre-defined weights. Then, these pre-computed hierarchical organizations will be inter-changed optimally, on-the-fly at each iteration, during the retrieval process.

9.3 Suitable applications

Although our research was focused on a particular type of data, namely images indexed by visual features, and on a particular type of application, namely a retrieval web-service, our retrieval approach is in principle suitable for any type of multimedia retrieval with minor changes.

Adaptation to other user interfaces

The minimalist user interface assumed by our retrieval system seems appropriate for unconventional human-computer interactions [34], as for example voice recognition, eye-gaze tracking or real-time EEG (i.e. electroencephalography) neurofeedback. We thought about two possible applications, but there are certainly more. One application could help disabled people to communicate simple needs. Another application could let doctors to search in medical databases during surgeries, hands-free.

Adaptation to other image collections

Although the retrieval system is generic for any kind of image collection, we believe that it may have a particular applicability in the professional domains, as for example medical, architecture, design or forensic. The system still lacks the retrieval performance necessary for generic commercial applications, but it could work sufficiently well for dedicated collections, where the indexing features could be better tuned than in the generic universal case of non-thematic photos.

Adaptation to other multimedia collections

All types of multimedia data could benefit greatly from our retrieval approach, and there are several cases in which they could benefit in special from our multi-modal approach. Movie retrieval could be an interesting application, as raw textual information can be derived from the speech transcript [58] and this textual information could complement greatly the visual content. Songs with lyrics in music databases constitute a similar case [47]. Another useful application could be for the medical databases that associate x-ray images and reports. As clinical diagnoses benefit from comparing similar (but not identical) cases [46], such a retrieval system as ours could be highly appreciated.

9.4 Closing note

In this thesis, we have investigated a query-free retrieval approach with full searching capabilities that promises an interactive access to image collections of unprecedented size. The iterative relevance feedback mechanism scales up one order of magnitude above most of the state-of-the-art iterative mechanisms. The adaptive mass-zoom system encompasses both regimes of exploration and exploitation, and the adaptive similarity metric increases the alignment between the system and the user.

User-based evaluations show that our approach extends the retrieval capabilities of the original framework, and give evidence that the approach is intuitive and able to deal with the user subjectivity in making similarity judgments. Moreover, the minimalist user interface is effortless and self-explanatory.

While a number of novel contributions are made, they are in no way complete solutions. The problems we deal with are relatively open-ended, with a lot of scope for incremental improvement, or even for adopting radically new approaches. We hope that our experimental findings give motivation for continuing this research direction.

A Web-application

The retrieval system[1] has been implemented as a web-application[2], which is distributed to the public under the AGPL[3] Version 3 open-source license. Besides the advantage of permanent availability for demos and evaluations, this implementation encourages the adherence to a realistic system architecture.

In this appendix, we describe briefly the web-application. First, we give an overview at the functional level. Second, we point out our high-level implementation choices. Further details are available in the software documentation included in the open-source release itself.

A.1 Functionality

The web-application implements the retrieval framework and all our contributions, and provides a versatile infrastructure for conducting user-based evaluations. Figures A.2-A.4 show a few screenshots of the web-interface. Figure A.2 shows the web-interface for searching. There is support for inter-changing the image collections and the system configurations. Figure A.3 shows the interface for testing. The active test configuration is hidden to the user for anonymity reasons. The target task is shown in words at the top of the page, and an image example is shown on the right side panel. Figure A.4 shows the interface for plotting the usual statistics.

The computational effort depends of course on the configuration. For a collection of 33,000 images with the original approach, where the computation depends linearly on the size of the collection, it takes 1 second per iteration and uses 300KB cache memory per user between iterations. For a collection of 1,000,000 images with the HEAT approach, where the computation depends linearly on the size of the trace, it takes 2-5 seconds per iteration and uses 20-50MB cache memory per user between iterations. These figures corresponds to our demo web-server running as a virtual server on a PC hardware machine with a dual-core CPU model Intel Core™-Duo E6700, 2.66GHz, 4MB cache, and 4GB of RAM.

---

[1] http://imr.idiap.ch/

[2] http://www.idiap.ch/software/imr/

[3] http://www.gnu.org/

Test management

There is infrastructure for managing the user tests and recording the searching sessions. A test consists of a set of searching sessions. Each session is performed on one image collection, and is configured with one system configuration and one target task. Therefore, a test is configured as a set of triplets of (image collection, system configuration, target task) data. The system configuration data define all the algorithmic settings as in Table A.1. The target task data provides the user with what she has to search for as in Table A.2.

During the testing, the test manager assigns the triplets randomly in an anonymized fashion. The test keeps track of its progress, and continues until the user performs all the triplets included in the test. The searching session data is accumulated iteration by iteration, and is stored in the database as in Table A.3.

name	the label and the display properties
TRACE	the trace size and the refinement type
SYS_THARGS	the calibration threshold parameters
SYS_WEIGHTS	the image feature default weights and the adaptation type
SYS_CONSISTENCY	the consistency estimation type
SYS_VERSION	the algorithm version – original, HEAT, random
datetime	the time of the saving in the database

Table A.1: System configuration data.

query	the text query
target	the target image set

Table A.2: Target task data.

user	the user that performed the session
config	the system configuration
task	the target task given to the user
iterations	the number of iterations (redundant)
evaluation	the evaluation given by the user
actionH	the actions performed on the input data
thargsH	the history of the image similarity thresholds
weightsH	the history of the image feature weights
consistencyH	the history of the consistency scores
dispfdbkH	the displayed image sets and the corresponding relevance feedback given by the user
traceH	the evolution of the size of the trace
timingH	the timing of system response and user feedback
datetime	the time of the saving in the database

Table A.3: Searching session data.

User registration

The user registration provides functionality to differentiate between anonymous, regular and staff users:

- The anonymous users can access only the minimal functionality to perform searching sessions.
- The regular users can access additional functionality to see the training examples, and perform evaluation tests.
- The staff users can access extra functionality to manage (create/delete/modify) the users, system configurations and target tasks, view the statistics of the system performance, and watch the user searching session logs saved in the database.

A.2 Implementation

Our web-application relies on a client-server architecture as shown in Figure A.1, and uses open standards and software. The implementation decouples the user interface from the core indexing and retrieval algorithms, and consists of four modules:

- the Django-based client with the graphical user interface
- the Django backend for client-server communication
- the database that stores the raw image data and the indexing information
- the server that performs the retrieval algorithms

The web-application is developed in Python, and is based on the Django[4] platform. The implementation has approximatively 12,000 lines of code, and complies with the PEP8[5] style standard. The software documentation is generated by using Sphinx[6]. The modular design allows easily new algorithm extensions and user interface enhancements.

Web-server deployment

The web-application is powered by the Apache[7] web-server. The current implementation supports multiple users logged in from different machines, but it does not support yet multiple searching sessions in parallel in the same browser. The application makes use of the cache-per-session backend mechanism, and requires that cookies are enabled in the browser.

---

[4] http://www.djangoproject.com

[5] http://http://www.python.org/dev/peps/pep-0008/

[6] http://sphinx.pocoo.org/

[7] http://www.apache.com

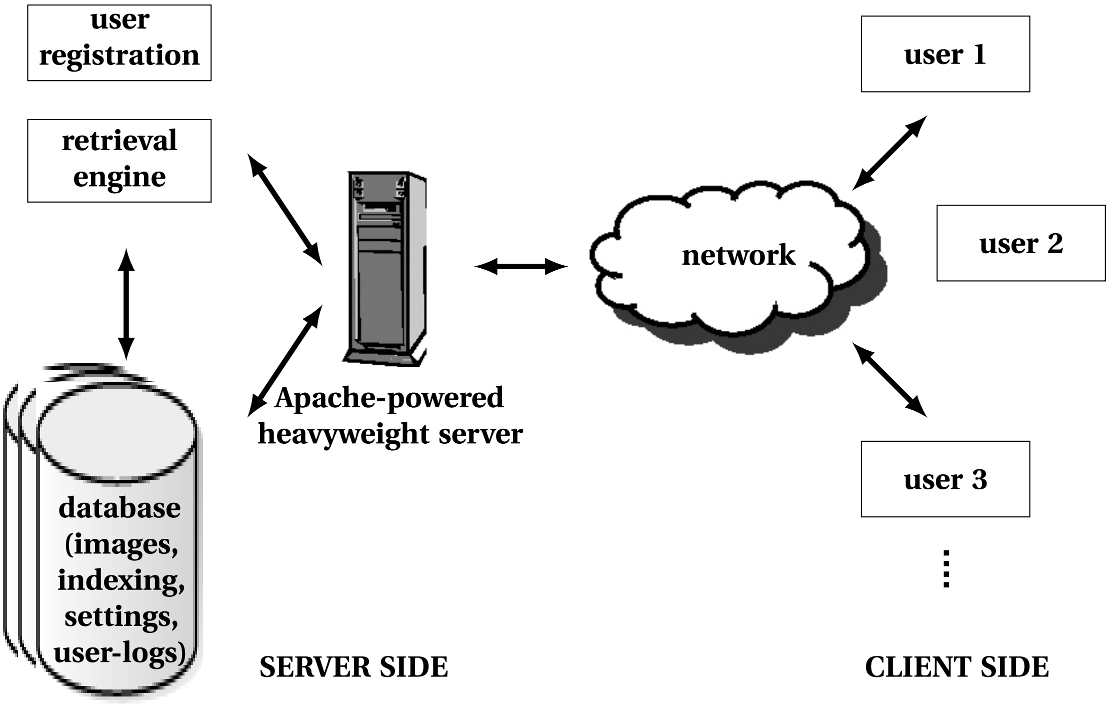
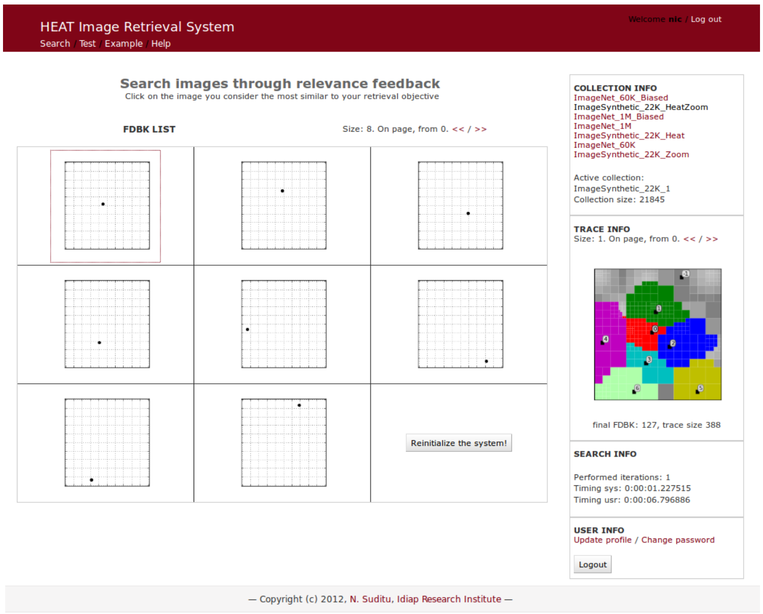

user registration
retrieval engine
database (images, indexing, settings, user-logs)
Apache-powered heavyweight server
network
user 1
user 2
user 3
SERVER SIDE
CLIENT SIDE

Figure A.1: Client-server architecture of the retrieval system. The retrieval system is powered by the Apache server, and this server supports multiple searching sessions via user cookies. The users interact with the retrieval system via web-interfaces like in Figures A.2-A.4.


HEAT Image Retrieval System

Search / Test / Example / Help

Welcome **nic** / Log out

Search images through relevance feedback

Click on the image you consider the most similar to your retrieval objective

FDBK LIST

Size: 8. On page, from 0. << / >>

Reinitialize the system!

COLLECTION INFO

ImageNet_60K_Biased
ImageSynthetic_22K_HeatZoom
ImageNet_1M_Biased
ImageNet_1M
ImageSynthetic_22K_Heat
ImageNet_60K
ImageSynthetic_22K_Zoom

Active collection:
ImageSynthetic_22K_1
Collection size: 21845

TRACE INFO

Size: 1. On page, from 0. << / >>

final FDBK: 127, trace size 388

SEARCH INFO

Performed iterations: 1
Timing sys: 0:00:01.227515
Timing usr: 0:00:06.796886

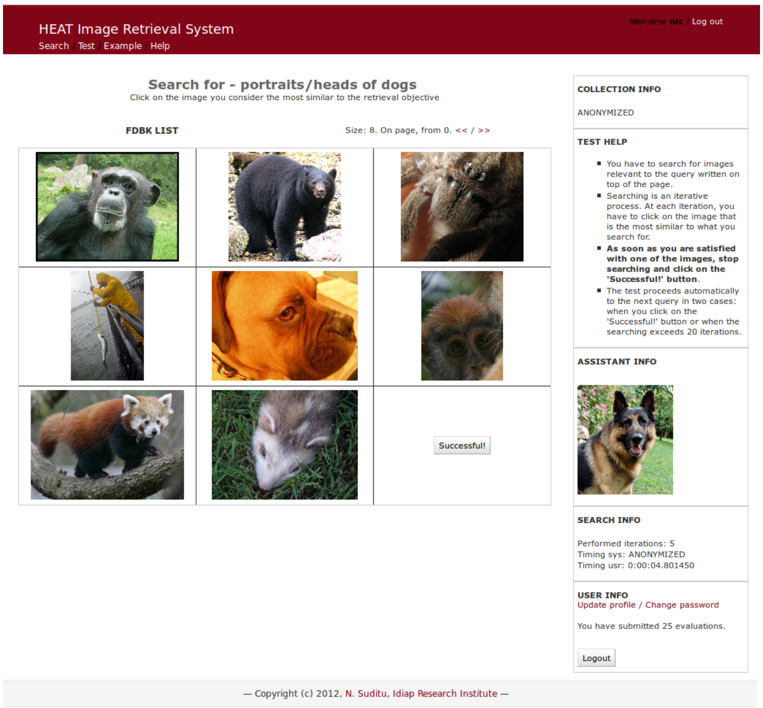
USER INFO

Update profile / Change password

Logout

— Copyright (c) 2012, N. Suditu, Idiap Research Institute —

Figure A.2: Screenshot of the web-interface for the searching application. Here, the interface is activated for the synthetic collection. Besides the searching functionality, there is a panel that provides additional information, and allows to switch between different collections.


HEAT Image Retrieval System
Search / Test / Example / Help
Welcome nic / Log out
FDBK LIST
Size: 8. On page, from 0. << / >>
Successful!
Logout

Search for - portraits/heads of dogs

Click on the image you consider the most similar to the retrieval objective

COLLECTION INFO

ANONYMIZED

TEST HELP

- You have to search for images relevant to the query written on top of the page.
- Searching is an iterative process. At each iteration, you have to click on the image that is the most similar to what you search for.
- **As soon as you are satisfied with one of the images, stop searching and click on the 'Successful!' button.**
- The test proceeds automatically to the next query in two cases: when you click on the 'Successful!' button or when the searching exceeds 20 iterations.

ASSISTANT INFO

SEARCH INFO

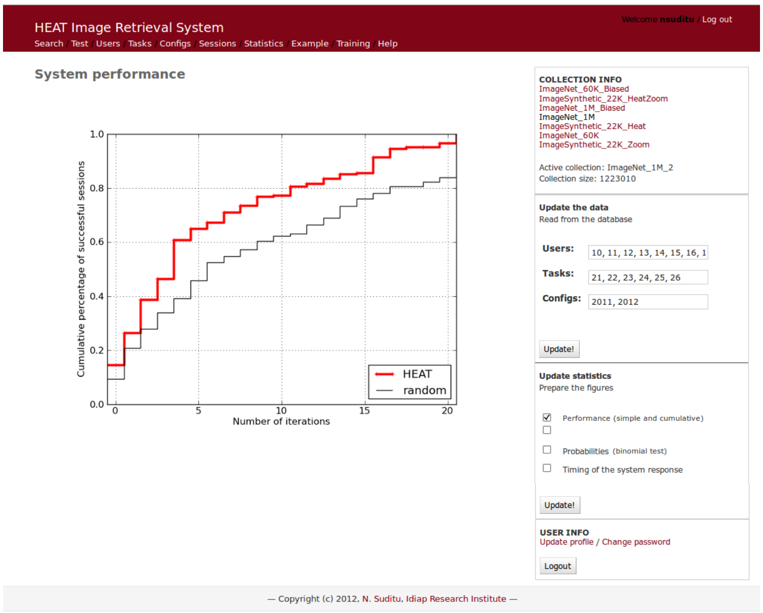
Performed iterations: 5
Timing sys: ANONYMIZED
Timing usr: 0:00:04.801450

USER INFO
Update profile / Change password

You have submitted 25 evaluations.

— Copyright (c) 2012, N. Suditu, Idiap Research Institute —

Figure A.3: Screenshot of the web-interface for the testing application. During the testing, the active collections and configurations are hidden to the users for anonymity reasons. The active target task is shown in words at the top of the page, and an image example is shown on the panel on the right side. The panel also provides guidance as well as progress information.


HEAT Image Retrieval System
Search / Test / Users / Tasks / Configs / Sessions / Statistics / Example / Training / Help
Welcome nsuditu / Log out
System performance
Cumulative percentage of successful sessions
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
Number of iterations
HEAT
random
Update!
Update!
Logout

COLLECTION INFO
ImageNet_60K_Biased
ImageSynthetic_22K_HeatZoom
ImageNet_1M_Biased
ImageNet_1M
ImageSynthetic_22K_Heat
ImageNet_60K
ImageSynthetic_22K_Zoom

Active collection: ImageNet_1M_2
Collection size: 1223010

Update the data
Read from the database

Users: 10, 11, 12, 13, 14, 15, 16, 1
Tasks: 21, 22, 23, 24, 25, 26
Configs: 2011, 2012

Update statistics
Prepare the figures

☑ Performance (simple and cumulative)
☐
☐ Probabilities (binomial test)
☐ Timing of the system response

USER INFO
Update profile / Change password

— Copyright (c) 2012, N. Suditu, Idiap Research Institute —

Figure A.4: Screenshot of the web-interface for the statistics application. This interface allows easy access to the usual statistical plots. One can select the plots of interest, and also can filter the data of certain users, system configurations or target tasks to be included in the plots.

Relational database

At the heart of the system, there is a relational database based on MySQL[8]. For each image collection, there are several dedicated tables:

- the table with the image data (e.g. the local directory path or the remote web-address of the raw image data), the indexing features and other meta-data (e.g. annotation keywords)
- the table with the hierarchical tree-like organization of the images
- the system configurations table
- the target tasks table
- the table with the user searching session logs

The user accounts are also handled in one table in the relational database. The table with the user searching session logs has cross-references (i.e. foreign keys) to the tables of users, system configurations and target tasks in order to enforce the data consistency.

Core-algorithm library

The core-functionality was implemented in modules aiming towards flexibility to try-out alternative retrieval algorithms. The retrieval algorithms are isolated from the logic of the relational database and the user interface via interface classes. For computational efficiency, the low-level routines are optimized by using Cython[9] and C++.

Pre-processing operations

The operations for extracting the image features and creating the indexing information (e.g. the pre-computed similarity distances for the original framework and the hierarchical organization for the HEAT framework) are implemented to run efficiently in parallel processes on a distributed architecture managed by a Sun's Grid Engine (SGE[10]).

The gradient descent algorithm for computing the optimized weighting vector for the log-based similarity metric is implemented as a simple routine, although it could be implemented to take advantage of the parallel processing as well.

---

[8] http://www.mysql.com

[9] http://www.cython.org/

[10] http://gridengine.sunsource.net

A.3 Remarks

We believe that we succeeded to adhere to a realistic system architecture. The object-oriented implementation design reflects all our knowledge and anticipation of the possible alternative development and research requirements. We hope that our application prototype adds transparency to our research work, facilitates the reproducibility of our experiments, and offers a good development platform for pursuing the forthcoming research.

Although it is not just a one-press of a button, it is quite straight-forward to set up the application for new collections and new evaluation scenarios. We provide the routines to set up the application for the image collections that we have used in our evaluations, namely ImageNet, Corel stock photo library and our synthetic collection, and these routines can be easily adapted for other image collections.

As we said already, we distributed our code to the public under the AGPL Version 3 open-source license. In particular, this license requires any further contributions and deployments to be distributed under the same terms as the original source-code, i.e. the AGPL terms, which is not the case for the standard GPL license. Choosing the AGPL license, we aimed to encourage the sharing of future extensions made by others.

B Test platform

We have developed a test platform for running the web-application programmatically, without human interaction. This platform implements an automatic user that interacts with the web-application in exactly the same way a human user does. This platform is useful for code verification and optimization, and thus is convenient in getting confidence before organizing the time-costly user-based evaluations.

In this appendix, we describe briefly this test platform. First, we give an overview at the functional level. Second, we explain how this platform can be used to provide abstract performance evaluations that resemble the user-based evaluations to a certain extent. For further reference, the test platform is part of our software package[1] distributed under the AGPL[2] Version 3 open-source license.

B.1 Functionality

The test platform is implemented on top of the web-application, and interacts with it in the same way a human user does. Besides being helpful for code testing and debugging purposes, the test platform can be used to perform automatic tests that resemble the real evaluations to a certain extent.

The main challenge of this test platform is to implement an automatic user that models the relevance feedback actions given normally by a human. In the current implementation, the automatic user embodies simply the ideal behavior of an oracle:

- knows precisely the target set S to search for, although in reality it exists only as a vaguely defined image category in the mind of the user
- always chooses the image $x_t^* \in D_t$ that is the closest to the target set S in the similarity metric used by the system, although in reality there is a semantic gap

---

[1] http://www.idiap.ch/software/imr/

[2] http://www.gnu.org/

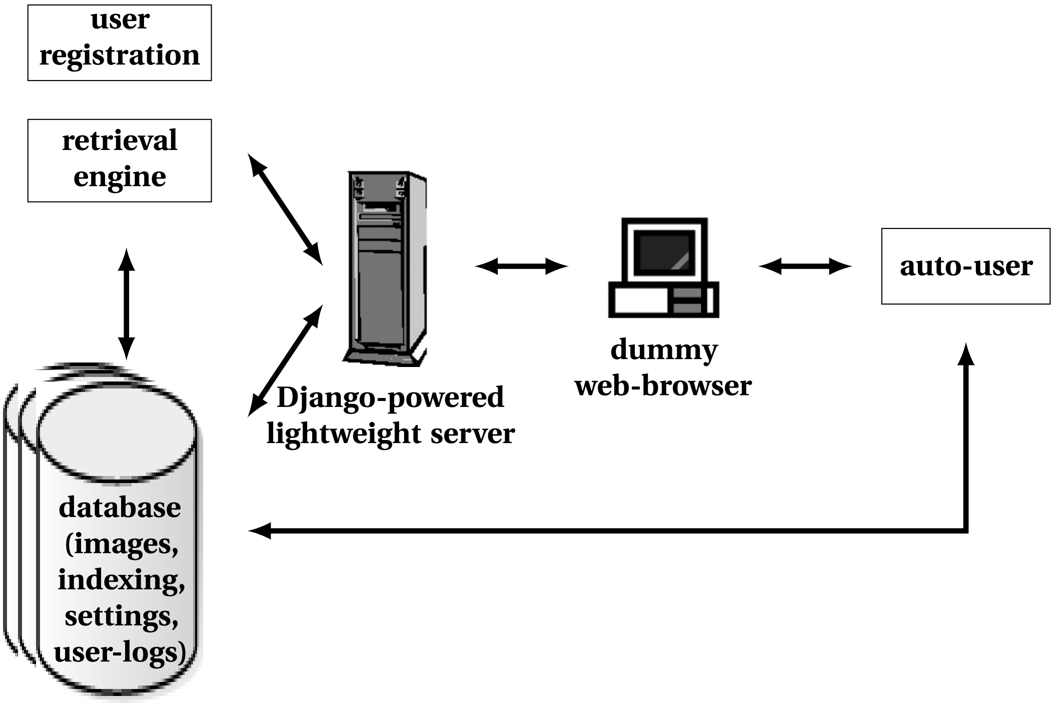
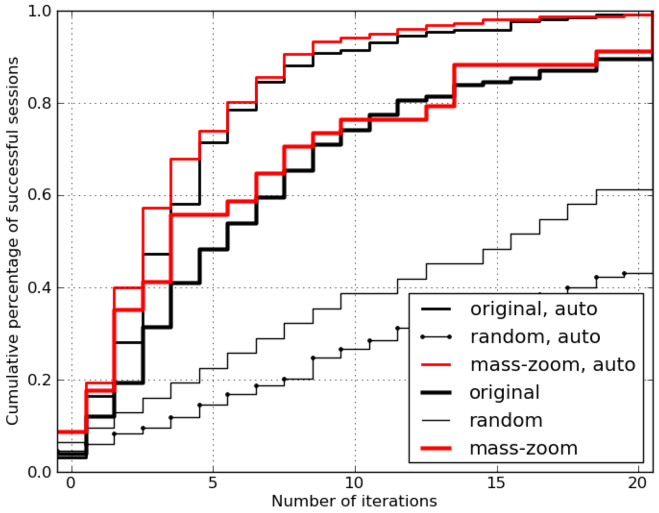

user registration
retrieval engine
database (images, indexing, settings, user-logs)
Django-powered lightweight server
dummy web-browser
auto-user

Figure B.1: Architecture of the test platform. The retrieval system is powered by the Django server, and this lightweight server running on the local machine is accessible via a dummy web-browser. The automatic user interacts with the retrieval system via the dummy web-browser in the same manner as a real user does.

B.2 Implementation

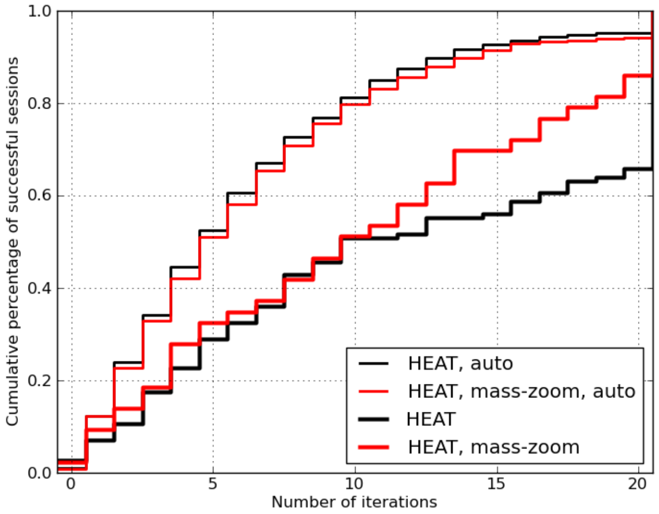
Our test platform emulates a client-server architecture as shown in Figure B.1 by using the test client class provided by the Django framework, which acts as a dummy Web browser. The test client class allows to test the web-interface (i.e. the Django-views), and therefore to interact with the Django-powered web-application programmatically. The implementation consists of three modules:

- a class that coordinates the workflow of the automatic user actions and the system responses. This class corresponds to the real context of accessing the web, logging in and proceeding with the tests.
- a class that encapsulates the interactions with the Django-view and the Html-page. This class corresponds to the user intelligence to use the application: what is the action required at a certain moment, and how to trigger this action.
- a class that encapsulates the similarity judgments on the displayed images given the target set. This class corresponds to the user intelligence to choose the closest image to what she is searching for.



60K image collection (ImageNet)

Figure B.2: Retrieval performance of the automatic tests in comparison with the user evaluations for the original system and the mass-zoom extension. The automatic user was assigned with target sets of size of about 1% of the 60K image collection, aiming to match the retrieval difficulty faced by the real users. The automatic tests out-perform the user evaluations as expected. One can see that the mass-zoom extension is helping very little the automatic user.



1M image collection (ImageNet)

Figure B.3: Retrieval performance of the automatic tests in comparison with the user evaluations for the HEAT system and the mass-zoom extension. The automatic user was assigned with target sets of size of about 1% of the 1M image collection, aiming to match the retrieval difficulty faced by the real users. The automatic tests out-perform the user evaluations as expected. One can see that the mass-zoom extension is a little bit counteracting for the automatic user.

We implemented the test platform in such a way that one can run efficiently multiple test instances in parallel processes on a distributed architecture managed by a Sun's Grid Engine (SGE[3]). Thus, the MySQL database is accessed in parallel by multiple instances of the retrieval application, as in the real web-server architecture.

B.3 Automatic tests

The automatic tests are organized in the same manner as the real tests, and they follow the same evaluation scenario. The only difference from the real tests is that the target tasks have to define the target explicitly as a set of images instead of just a semantic description in words.

First, the automatic tests are set up to include certain combinations of image collections, target tasks and system configurations. Then, the automatic users are launched to perform the test. During the automatic testing, the test manager serves the searching sessions as during the real tests. The automatic users are only given the target tasks, but not the system configurations, and they are not biased in any way by the internal state of the retrieval system.

The automatic tests are useful in getting confidence before organizing the time-costly user-based evaluations. As we explain next, the automatic tests help in assessing the retrieval performance of the web-application. Still, the interpretations should be done with caution.

The automatic tests help in assessing the retrieval performance of the web-application. As the automatic user embodies the ideal user behavior, the automatic tests give a hint on the potential theoretical performance of the retrieval system. Still, the automatic tests should not be mistaken with the real tests who bring in the big challenge of user subjective perception in judging the image similarities.

Figures B.2-B.3 show the retrieval performance of the automatic tests in comparison with the user evaluations for several system combinations. Figure B.2 corresponds to the 60K image collection, and Figure B.3 corresponds to the 1M image collection. The automatic user was assigned with target sets of size of about 1% of the image collections, aiming to match the retrieval difficulty faced by the real users. The automatic tests out-perform the user evaluations in all cases as expected.

One can see that the mass-zoom extension is helping very little the automatic user, and is even a little bit counteracting for the 1M image collection. Maybe the mass-zoom heuristic is too aggressive for the ideal user who never does relevance feedback mis-judgments. This is one example when the interpretation of the abstract retrieval performance of the automatic user should be done with caution.

---

[3] http://gridengine.sunsource.net

B.4 Remarks

We made extensive use of test platform in order to evaluate our alternative ideas in their early incipient stages. The test platform is useful in getting confidence before organizing the time-costly user-based evaluations, since it provides important clues almost for free (i.e. by simply running automatic tests for a few hours).

The interpretation of the abstract retrieval performance of the automatic user should be done with caution. Currently, the automatic user implements simply the ideal abstract behavior. One could implement more realistic behaviors by introducing for example some controlled randomness that would eventually model better the human behavior.

C Image collections

There are many public shared photography collections, some older as Getty Images or Corbis, and some recent as Flickr or Facebook, and there are even more specialized professional photography collections, as for example medical archives or designer portfolios.

Still, ever since the birth of the digital era, there has been a growing need for standardized data in the image and vision research communities. The motivation for standardized data is clear. Good research needs good resources, and it would be tremendously helpful if there were image databases accepted by the research community for reference and collaboration.

In our research experiments and evaluations, we have set up our retrieval system for three collections, namely the Corel stock photo library [45, 53], the ImageNet dataset [18] and a synthetic collection generated by ourselves. Here, we present briefly these collections and their characteristics.

C.1 Corel stock photo library

Corel Stock Photo Library has been used in many publications to demonstrate the performance of content-based image retrieval systems and it was the *defacto* standard in the field for quite some time [45, 53]. Corel Stock Photo Library offers a collection of more than 800 Photo CDs, each of them containing about 100 images. The images are grouped into semantic themes, and each image is associated with a few keywords from a controlled vocabulary. In Figure C.1, we show a few examples of images and their associated keywords in order to provide a general impression of the collection.

Recently, the interest for this collection has decreased drastically due to some obvious weaknesses. First, the collection is divided in over 800 small subsets, and there is no one single compact repository that group all of them and that can be referred to. Even people of the same research lab often use different subsets. Second, the collection is somehow out-dated by the evolution of multimedia technologies in storage capacity, processing power and on-line accessibility. A collection of less than 100,000 images is not large-scale in the present terms.

Geese And Chicks bird
geese chicks water

single Thunderbird
Flying Upside Down
plane jet f-16 sky

The sun sets behind the
anchored schooner sunset
boat water horizon

snowy Foothills mountain
trees snow sky

one hand pole flip
Lake Tahoe Usa
people ski snow sport

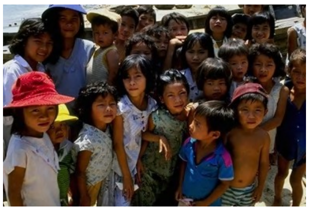
Niagara Falls American Side
waterfall rocks mist cascade

Rooftops of Prague
rooftops buildings trees
chimneys

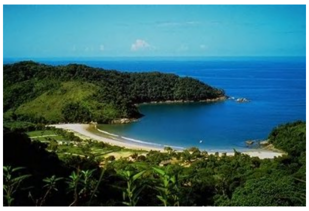
Marketplace Oaxaca
Mexico market place
people flowers plants

African Elephants
trunks ground trees

Entranceway To The
Olympic Stadion Olympia
stone wall arch entrance



children on the beach
people clothing
children hat



small bay between Sao
Paolo and Rio de Janeiro
bay beach water trees

Figure C.1: Examples of images and their associated keywords from the Corel stock photo library. They are related to the 12 semantic categories that we used in our multi-modal experiments in Table 7.2. Here, one can observe how the textual information match the visual content of the images, and still is incomplete in various ways.

We have used a subset of 35,000 images of the Corel stock photo library for our evaluations of the multi-modal extension in §7. As elaborated in §7, our multi-modal system employs visual features based on SIFT (Scale Invariant Feature Transform) [41], and textual features based on LSA (latent semantic analysis) [17].

C.2 ImageNet dataset

ImageNet[1] is an image dataset organized according to the WordNet hierarchy of *synsets* [18]. WordNet[2] is a large lexical database of English [43, 22] that is considered to be the most important resource available to researchers in computational linguistics and text analysis, and is also freely available. It superficially resembles a thesaurus, in that it groups words together based on their meanings. Each meaningful concept in WordNet, possibly described by multiple words or phrases, is called a *synonym set* or *synset.* There are more than 100,000 synsets in WordNet, and they are interlinked by means of conceptual-semantic and lexical relations.

ImageNet aims to provide on average 1,000 images to illustrate each synset. Images of each concept are quality-controlled and human-annotated via the Amazon Mechanical Turk. In its completion, ImageNet will offer tens of millions of cleanly sorted images for all the concepts in the WordNet hierarchy. At the time of the 2010 release, ImageNet was covering 1,000 synsets, each synset having 500–2,500 images. ImageNet is continuously updated, and one can inspect its progress via the on-line navigation interface based on the WordNet hierarchy of synsets as shown in Figure C.2.

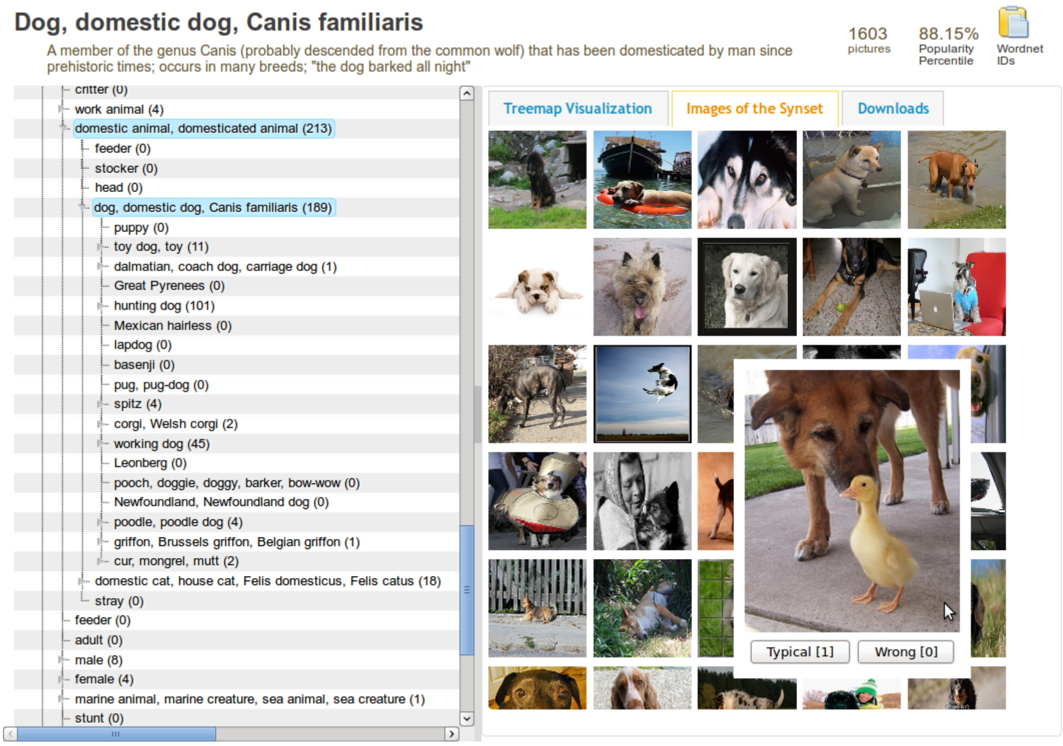
ImageNet provides densely sampled SIFT features that are also quantized in bags-of-words of dimension $1,000$. The feature extraction is fully explained in [18]. The images are resized to have a maximum side length of no more than 300 pixel. SIFT descriptors are computed on 20×20 overlapping patches with a spacing of 10 pixels. The images are also further downsized (to $1/2$ the side length and then $1/4$ of the side length) and more descriptors are computed. Next, k-means clustering is performed with a random subset of 10 million SIFT descriptors in order to derive a vocabulary of 1,000 visual words. Each SIFT descriptor is quantized into a visual word using the nearest cluster center.

We have used the ImageNet dataset as it was released in 2010 for most of our evaluations in §4-6. Considering all the images provided with pre-computed SIFT features (Scale Invariant Feature Transform) [41] and with valid url at that date, we obtained a large collection including about 1,054,000 images. Then, we sampled uniformly a small collection of 33,000 images (i.e. 3% of the large collection), and another one of 60,000 images (i.e. 6% of the large collection).

---

[1] http://www.image-net.org

[2] http://wordnet.princeton.edu



Dog, domestic dog, Canis familiaris

A member of the genus Canis (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds; "the dog barked all night"

Figure C.2: Screenshot of the ImageNet exploration tool web-interface. This interface allows to navigate through the ImageNet based on the WordNet hierarchy of synsets. ImageNet aims to provide on average 1,000 images to illustrate each synset from WordNet hierarchy. Images of each concept are quality-controlled and human-annotated via the Amazon Mechanical Turk. In its completion, ImageNet will offer tens of millions of cleanly sorted images for all the concepts in the WordNet hierarchy.

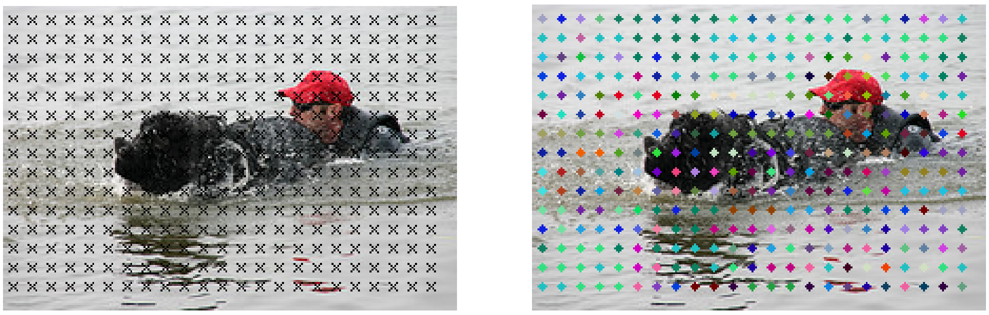
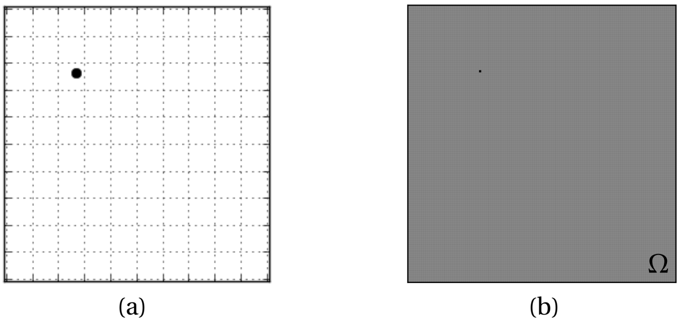


Figure C.3: Visual features provided by ImageNet are SIFT-based bags-of-words of dimension 1,000 derived from densely sampled SIFT descriptors. ImageNet provides all the details of their feature extraction pre-processing in [18]. Besides the quantized bags-of-words, they provide the raw SIFT descriptors as well as the spatial coordinates of each descriptor.



(a)

(b)

Figure C.4: Abstract representation with a synthetic collection. (a): Each image has as visual content one single point in the 2D Cartesian space, and the indexing features are the corresponding coordinates of that point. The similarity distances between images are the Euclidean distances between their corresponding points. (b): The duality between points and images is used to represent the entire collection. Each point in this abstract representation corresponds to an image in the collection. Additionally, the grey-levels of the points tell the probabilities of relevance of their corresponding images.

C.3 Synthetic collection

We have created a synthetic image collection in order to accompany the rigorous mathematical formulations, and to offer an intuitive illustration of the system behavior. The collection has 22,000 images made up of points at different locations. In Figure C.5, there are a few examples of actual images from the synthetic collection.

One key characteristic of this synthetic collection is that there is no semantic gap between the low-level abstract similarity metric and the high-level visual meaning. The indexing image features are the corresponding coordinates of that point inside the image, and the similarity distances between images are the Euclidean distances between their corresponding points. Thus, the user similarity judgments cannot be more straight-forward than this: the closer the points, the more similar the images.

Another key characteristic of this synthetic collection is the duality between points and images. The point locations have been chosen in such a way that it allows a nice smooth symmetric abstract representation. As explained in Figure C.4, each point in this abstract representation corresponds to an image in the collection. Additionally, the grey-levels of the points tell the probabilities of relevance of their corresponding images.

This synthetic collection proved to be truly helpful for intuitive analysis of the system behavior. We explain in Figures C.6-C.7 the different representations that allow to visualize the internal state of the retrieval system at different stages. In Figure C.6, one can see how the displayed images are selected, and how the Voronoi tessellation algorithm grows the clusters. In Figure C.7, one can see the relevance feedback given by the user, and the probabilities of relevance.

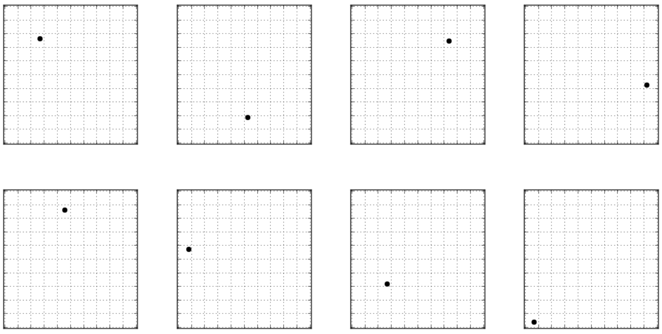
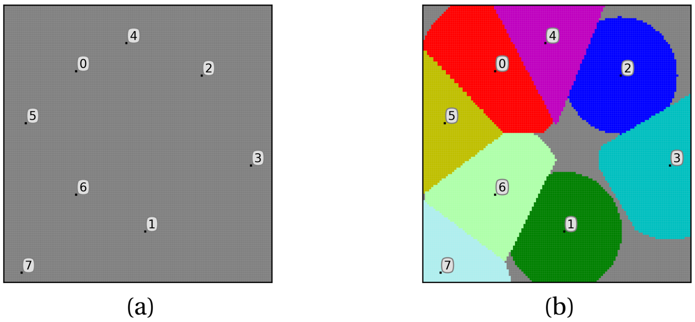
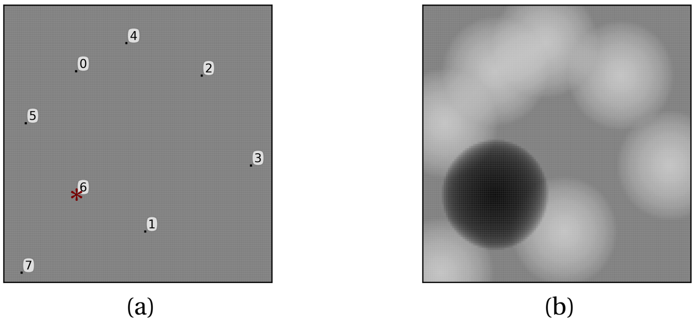


Figure C.5: Examples of images in the synthetic collection. Actually, these 8 images are the images in the display set D_0 that is used consistently as the first display set in all the illustrated searching sessions.



(a) (b)

Figure C.6: Abstract representations of D_0 shown above in Figure C.5. (a): The displayed images D_0 are shown, and the order in which they where selected by the sampling algorithm is indicated. (b): The clusters grown by the Voronoi tessellation algorithm are shown in colors.



(a) (b)

Figure C.7: Abstract representations of D_0 shown above in Figure C.5. (a): The displayed images D_0 are shown, and the relevance feedback given by the user x_0^* is indicated. (b): The probabilities of relevance are encoded by the grey-levels of the corresponding points.

Bibliography

[1] James Abello, Stephen G. Kobourov, and Roman Yusufov. Visualizing large graphs with compound-fisheye views and treemaps. In *Graph Drawing*, volume 3383 of *Lecture Notes in Computer Science*, pages 431–441. 2005.

[2] Jeffrey R. Bach, Charles Fuller, Amarnath Gupta, Arun Hampapur, Bradley Horowitz, Rich Humphrey, Ramesh C. Jain, and Chiao-Fe Shu. The Virage image search engine: An open framework for image management. In *Proceedings of Symposium on Electronic Imaging: Science and Technology – Storage and Retrieval for Still Image and Video Databases IV, IS&T/SPIE*, pages 76–87, 1996.

[3] Kobus Barnard, Pinar Duygulu, David A. Forsyth, Nando de Freitas, David M. Blei, and Michael I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[4] Benjamin B. Bederson. PhotoMesa: A zoomable image browser using quantum treemaps and bubblemaps. In *Proceedings of the 14th ACM symposium on User interface software and technology*, pages 71–80, 2001.

[5] Adam L. Buchsbaum and Jefferey R. Westbrook. Maintaining hierarchical graph views. In *Proceedings of the 11th ACM-SIAM symposium on Discrete algorithms*, pages 566–575, 2000.

[6] I. Campbell and K. van Rijsbergen. The ostensive model of developing information needs. In *Proceedings of the International Conference on Conceptions of Library and Information Science: Integration in Perspective (CoLIS)*, pages 251–268, 1996.

[7] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik. BlobWorld: A system for region-based image indexing and retrieval. In *Proceedings of the 3th International Conference on Visual Information Systems*, volume 1614, page 660, January 1999.

[8] Edward Chang, Kwang-Ting Cheng, Wei-Cheng Lai, Ching-Tung Wu, Chengwei Chang, and Yi-Leh Wu. PBIR: Perception-based image retrieval – A system that can quickly capture subjective image query concepts. In *Proceedings of the 9th ACM International Conference on Multimedia*, pages 611–614, October 2001.

[9] Edward Chang and Beitao Li. MEGA–The maximizing expected generalization algorithm for learning complex query concepts. *ACM Transactions on Information Systems*, 21(4):347–382, October 2003.

[10] Shi-Kuo Chang and Arding Hsu. Image information systems: Where do we go from here? *IEEE Transactions on Knowledge and Data Engineering*, 4(5):431–442, October 1992.

[11] Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11:1109–1135, March 2010.

[12] Scott Counts and Eric Fellheimer. Supporting social presence through lightweight photo sharing on and off the desktop. In *Proceedings of the ACM SIGCHI international conference on Human factors in computing systems*, pages 599–606, 2004.

[13] Ingeman J. Cox, Matthew L. Miller, Thomas P. Minka, Thomas V. Papathomas, and Peter N. Yianilos. The Bayesian image retrieval system, PicHunter: theory, implementation, and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–37, January 2000.

[14] Michel Crucianu, Marin Ferecatu, and Nozha Boujemaa. Relevance feedback for image retrieval: a short survey. In *State of the Art in Audiovisual Content-Based Retrieval, Information Universal Access and Interaction including Datamodels and Languages*, 2004.

[15] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2):1–60, April 2008.

[16] Marc Davis, Simon King, Nathan Good, and Risto Sarvas. From context to content: leveraging context to infer media metadata. In *Proceedings of the 12th ACM international conference on Multimedia*, pages 188–195, 2004.

[17] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American society for information science*, 41(6):391–407, September 1990.

[18] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[19] Pinar Duygulu, Kobus Barnard, J. F. G. de Freitas, and David A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of the 7th European Conference on Computer Vision, ECCV - Part IV*, volume 2353, pages 349–354, January 2002.

[20] M. Emre Celebi and Y. Alp Aslandogan. Human perception-driven, similarity-based access to image databases. *Proceedings of the Artificial Intelligence Research Society Conference*, pages 245–251, May 2005.

[21] Yuchun Fang and Donald Geman. Experiments in mental face retrieval. In *Proceedings of the 5th International Conference on Audio and Video-based Biometric Person Authentication*, pages 637–646, July 2005.

[22] Christiane Fellbaum. WordNet: An electronic lexical database. *Cambridge, MA: MIT Press*, 1998.

[23] Marin Ferecatu. *Image retrieval with active relevance feedback using both visual and keyword-based descriptors.* PhD. Thesis, INRIA-University of Versailles Saint Quentin-en-Yvelines, France, 2005.

[24] Marin Ferecatu and Donald Geman. Interactive search for image categories by mental matching. In *Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–8, October 2007.

[25] Marin Ferecatu and Donald Geman. A statistical framework for image category search from a mental picture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1087–1101, June 2009.

[26] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: The QBIC system. In *IEEE Computer*, volume 28, pages 23–32, September 1995.

[27] Anjan Goswami, Ruoming Jin, and Gagan Agrawal. Fast and Exact Out-of-Core K-Means Clustering. In *IEEE International Conference on Data Mining*, pages 83–90, November 2004.

[28] J. Han, K.N. Ngan, Mingjing Li Li, and Hong-Jiang Zhang. A memory learning framework for effective image retrieval. *IEEE Transactions on Image Processing*, 14:511–524, April 2005.

[29] Donna Harman. Relevance feedback revisited. In *Proceedings of the 15th international ACM SIGIR conference on Research and development in information retrieval*, pages 1–10, 1992.

[30] Daniel Heesch. A survey of browsing models for content based image retrieval. *Journal of Multimedia Tools and Applications*, 40(2):261–284, 2008.

[31] Chu-Hong Hoi and Michael R. Lyu. A novel log-based relevance feedback technique in content-based image retrieval. In *Proceedings of the 12th annual ACM International Conference on Multimedia*, MULTIMEDIA '04, pages 24–31, 2004.

[32] Chu-Hong Hoi, Michael R. Lyu, and R. Jin. A unified log-based relevance feedback scheme for image retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 18:509–524, April 2006.

[33] Yoshiharu Ishikawa, Ravishankar Subramanya, and Christos Faloutsos. MindReader: Querying databases through multiple examples. In *Proceedings of 24rd International Conference on Very Large Data Bases*, pages 218–227, August 1998.

[34] Alejandro Jaimes and Nicu Sebe. Multimodal human computer interaction: A survey. *Computer Vision in Human-Computer Interaction*, 3766:1–15, September 2005.

[35] Jiwoon Jeon, Victor Lavrenko, and Raghavan Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th ACM SIGIR international conference on Research and development in informaion retrieval*, pages 119–126, 2003.

[36] Mohammed L. Kherfi, Djemel Ziou, and Alan Bernardi. Image Retrieval from the World Wide Web: Issues, Techniques, and Systems. *ACM Computing Surveys*, 36(1):35–67, 2004.

[37] Marco La Cascia, Saratendu Sethi, and Stan Sclaroff. Combining textual and visual cues for content-based image retrieval on the world wide web. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, 1998.

[38] Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain. Content-based multimedia information retrieval: State-of-the-art and challenges. *ACM Transactions on Multimedia Computing, Communication and Applications*, 2(1):1–19, 2006.

[39] Jia Li and James Z. Wang. Real-time computerized annotation of pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):985–1002, 2008.

[40] Yen-Yu Lin, Tyng-Luh Liu, and Hwann-Tzong Chen. Semantic manifold learning for image retrieval. In *Proceedings of the 13th annual ACM International Conference on Multimedia*, MULTIMEDIA ’05, pages 249–258, 2005.

[41] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[42] B. S. Manjunath, Philippe Salembier, and Thomas Sikora. *Introduction to MPEG-7: Multimedia Content Description Interface*. John Wiley & Sons, Inc., 2002.

[43] George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38:39–41, 1995.

[44] Florent Monay and Daniel Gatica-Perez. On image auto-annotation with latent space models. In *Proceedings of the 11th ACM international conference on Multimedia*, pages 275–278, 2003.

[45] Henning Müller, Stéphane Marchand-Maillet, and Thierry Pun. The truth about Corel - Evaluation in image retrieval. In *Proceedings of the International Conference on Image and Video Retrieval*, pages 38–49, July 2002.

[46] Henning Müller, Nicolas Michoux, David Bandon, and Antoine Geissbuhler. A review of content-based image retrieval systems in medical applications – Clinical benefits and future directions. *International Journal in Medical Informatics*, 73(1):1–23, 2004.

[47] Nicola Orio. *Music information retrieval: A tutorial and review*, volume 1. Foundations and Trends in Information Retrieval, November 2006.

[48] Till Quack, Ullrich Mönich, Lars Thiele, and B. S. Manjunath. Cortina: a system for large-scale, content-based web image retrieval. In *Proceedings of the 12th ACM international conference on Multimedia*, pages 508–511, 2004.

[49] Yong Rui and Thomas S. Huang. A novel relevance feedback technique in image retrieval. In *Proceedings of the 7th ACM international conference on Multimedia – Part II*, pages 67–70, 1999.

[50] Yong Rui, Thomas S. Huang, and Shih-Fu Chang. Image Retrieval: Current Techniques, Promising Directions, and Open Issues. *Journal of Visual Communication and Image Representation*, 10(1):39–62, 1999.

[51] Young Rui, Thomas S. Huang, Michael Ortega, and Sharad Mehrotra. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Video Technology*, 8(5):644–655, 1998.

[52] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.

[53] Tefko Saracevic. Evaluation of evaluation in information retrieval. In *Proceedings of the 18th ACM SIGIR international conference on Research and development in information retrieval*, pages 138–146, 1995.

[54] Stan Sclaroff, Marco La Cascia, and Saratendu Sethi. Using textual and visual cues for content-based image retrieval from the world wide web. *Image Understanding*, 75(2):86–98, 1999.

[55] Ben Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, January 1992.

[56] Arnold W.M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[57] John R. Smith and Shih-Fu Chang. VisualSEEk: a fully automated content-based image query system. In *Proceedings of the 4th ACM international conference on Multimedia*, pages 87–98, 1996.

[58] Cees G.M. Snoek and Marcel Worring. A review on multimodal video indexing. In *Proceedings of the ICME*, volume 2, pages 21–24, 2002.

[59] Hideyuki Tamura and Naokazu Yokoya. Image database systems: A survey. *Pattern Recognition*, 17(1):29–43, 1984.

[60] J.B. Tenenbaum, V. de Silva, and J.C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, 2000.

[61] J. Urban, J.M. Jose, and C. J. Van Rijsbergen. An adaptive technique for content-based image retrieval. *Multimedia Tools and Applications Processing (MTAP)*, 31:1–28, 2006.

[62] Remco C. Veltkamp and Mirela Tanase. Content-based image retrieval systems: A survey. *Technical Report UU-CS-2000-34, Department of Computer Science, Utrect University, The Netherlands*, October 2000.

[63] Alessandro Vinciarelli, Nicolae Suditu, and Maia Pantic. Implicit human-centered tagging. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1428–1431, June/July 2009.

[64] James Z. Wang, Nozha Boujemaa, Alberto Del Bimbo, Donald Geman, Alexander G. Hauptmann, and Jelena Tešić. Diversity in multimedia information retrieval research. In *Proceedings of the 8th ACM MIR international workshop on Multimedia information retrieval*, pages 5–12, 2006.

[65] Jason Weston, Samy Bengio, and Nicolas Usunier. Large scale image annotation: learning to rank with joint word-image embeddings. *Journal of Machine Learning*, 81(1):21–35, 2010.

[66] Yi-Leh Wu, King-Shy Goh, Beitao Li, Huaxing You, and Edward Y. Chang. The anatomy of a multimodal information filter. In *Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 462–471, 2003.

[67] Atsuo Yoshitaka and Tadao Ichikawa. A survey on content-based retrieval for multimedia databases. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, volume 11, February 1999.

[68] Xiang Sean Zhou and Thomas S. Huang. Unifying keywords and visual contents in image retrieval. *IEEE MultiMedia*, 9(2):23–33, April/June 2002.

[69] Xiang Sean Zhou and Thomas S. Huang. Relevance feedback for image retrieval: A comprehensive review. *Journal of Multimedia Systems*, 8(6):536–544, 2003.

Curriculum Vitae

Personal data

Full name	Nicolae Suditu
Date of birth	18 December 1974
Place of birth	Baia-Mare, Romania
Citizenship	Romanian

Research interests

My research interests lie in the fields of computer vision and machine learning. I enjoy working on practical approximate solutions for computationally intractable problems. My current research is related to large-scale interactive content-based image retrieval with focus on relevance feedback, feature selection, hierarchical clustering and adaptive similarity metrics.

Education

Aug. 2008 – Jan. 2013	École Polytechnique Fédérale de Lausanne (EPFL), Switzerland Doctoral School in Electrical Engineering (EDEE) *PhD in Electrical Engineering*
Apr. 2002 – Mar. 2004	Technische Universiteit Eindhoven (TU/e), The Netherlands Department of Electrical Engineering *Post-master on Engineering Design*
Oct. 1998 – Jul. 1999	Polytechnic University of Bucharest, Romania Department of Engineering Sciences, English stream *Master in Systems Modeling and Simulation*
Oct. 1993 – Jul. 1998	Polytechnic University of Bucharest, Romania Faculty of Electronics and Telecommunications *Bachelor in Telecommunication*

Professional skills

Research experience in multimedia information retrieval and signal processing
Background and course work in statistics, computer vision and machine learning
Work experience in medical image processing and software design
Programming: Python, C, ObjectiveC, C++, Matlab
Web-related: Apache, Django, MySQL, Html, jQuery, pHp
Good in writing scientific articles and software documentation
Languages: fluent English, intermediate French, native Romanian, basic German

Work experience

Aug. 2008 – Jan. 2013	Idiap Research Institute, Switzerland, affiliated to École Polytechnique Fédérale de Lausanne (EPFL) *Research Assistant* My research was focused on interactive large-scale content-based image retrieval. My PhD thesis director was Dr. François Fleuret. A demo web-application is available on-line at http://imr.idiap.ch/ and its source code is available at http://www.idiap.ch/software/imr/. Working with: Python, Cython, Django, MySQL, Sun Grid Engine
Mar. 2003 – Jul. 2008	Philips Medical Systems, The Netherlands Healthcare Informatics – Clinical Advanced Applications *Design Software Engineer* My job was to create prototypes of various medical applications for MRI and CT scanners and to investigate alternative choices together with a few clinicians. I was part of a team of 10 and I was developing software at functional level as well as at user interface level, being responsible for all the design steps including documentation and testing. Working with: ObjectiveC, sometimes C++, rarely C#, UML, GIT
Oct. 1999 – Apr. 2002	Technische Universiteit Eindhoven (TU/e), The Netherlands Department of Electrical Engineering *Research Assistant in Signal Processing Systems Group* My research was related to audio signal processing in a multi-microphone array setup for de-noising, dereverberation and blind source separation. Working with: Matlab, C, C++, sometimes Assembler

Other activities

Personal-development	University and company trainings in various subjects like technical writing, giving presentations, team work, time management
Business-related	Many courses in innovation, management, entrepreneurship and intellectual property attended at TU/e, EPFL, VentureLab and EPO
Science-related	Reviewer or co-reviewer for conferences like NIPS, CVPR and ICCV

Scientific publications

1. **Theses**

   - N. Suditu, “Adaptive relevance feedback for large-scale image retrieval”, PhD thesis, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, 2013.
   - N. Suditu, “Active GRAF: A Framework for Measurement and Segmentation”. The thesis was part of a one-year project at Philips Medical Systems in The Netherlands, 2004.
   - N. Suditu, “Shrinkage Filtering and Wavelet Maxima Filtering Applied in Medical Images”. The thesis was part of a Socrates Mobility Program at the Medical School, University of Patras in Greece, 1999.
   - N. Suditu, “Probabilistic Decision Making for Traffic Routing in ATM Networks”. The thesis was part of a project supported by the Finnish Academy at the Faculty of Information Technology, University of Jyväskylä in Finland, 1998.

2. **Conference papers**

   - N. Suditu and F. Fleuret, “Interactive Image Retrieval with Log-based Similarity Learning”, in Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR), July 2013, *submitted*.
   - N. Suditu and F. Fleuret, “Iterative Relevance Feedback with Adaptive Exploration/Exploitation Trade-off”, in Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM), October 2012, pp.1323–1331.
   - N. Suditu and F. Fleuret, “HEAT: Iterative Relevance Feedback with One Million Images”, in Proceedings of the IEEE International Conference on Computer Vision (ICCV), November 2011, pp.2118–2125.
   - A. Vinciarelli, N. Suditu and M. Pantic, “Implicit Human-Centered Tagging”, in Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), June 2009, pp. 1428–1431.

- N. Suditu and P.C.W. Sommen, "A study towards partial dereverberation methods", in Proceedings of the IEEE Signal Processing Symposium SPS'2002, Leuven, Belgium, March 2002, pp.185–188.
- N. Suditu, J.v.d. Laar, and P.C.W. Sommen, "Evaluation of dereverberation capabilities of a nearfield broadband beamformer", in Proceedings of ProRISC'2001, Veldhoven, The Netherlands, November 2001, pp. 656–661.
- N. Suditu and P.C.W. Sommen, "On the convergence of a partitioned frequency domain adaptive filter", in Proceedings of ProRISC'2000, Veldhoven, The Netherlands, December 2000, pp. 531–536.

3. **Technical reports**

   - N. Suditu, A. Vinciarelli and F. Fleuret, "Query-Free Interactive Image Retrieval Based on Visual and Textual Features", Idiap Research Institute internal research report, October 2010.

4. **Patents**

   - R. Habets, N. Suditu, S. Lobregt and F. Gerritsen, Title: "An image analysis system and an imaging system for an object mapping in a multi-dimensional dataset", Applicant: Koninglijke Philips Electronics N.V., WO 2005/106793, 2005.

5. **Open-source code**

   - N. Suditu, http://www.idiap.ch/software/imr/, software release copyrighted by Idiap Research Institute, available under the AGPL Version 3 open-source license, 2010–2012.
   - N. Suditu, http://imr.idiap.ch/, on-line demo web-application, 2009–2012.