Multi-descriptor retrieval in digitalized photographs collections

Neelanjan Bhowmik

▶ **To cite this version:**

Neelanjan Bhowmik. Multi-descriptor retrieval in digitalized photographs collections. Geography. Université Paris-Est, 2017. English. NNT : 2017PESC1037 . tel-01759559

HAL Id: tel-01759559
https://theses.hal.science/tel-01759559

Submitted on 5 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ PARIS-EST

École Doctorale MSTIC

Sciences et Technologies de l'Information Géographique

Neelanjan BHOWMIK

Multi-descriptor retrieval in digitalized photographs collections

Présentée pour obtenir le grade de docteur
de Université PARIS-EST
Novembre, 2017

Jury de Thèse

Vincent ORIA	**Président**
Jenny BENOIS-PINEAU	**Rapporteur**
Liming CHEN	**Rapporteur**
Marin FERECATU	**Examinateur**
Valérie GOUET-BRUNET	**Directeur de thèse**
Gabriel BLOCH	**Encadrant**

This thesis has been conducted at the Nicéphore Cité (Chalon-sur-Saône, France) and at the LaSTIG (Laboratoire en sciences et echnologies de l'information géographique) laboratory of the Research Department of IGN (Institut National de l'Information Géographique et Forestière).

Cette thèse s'est déroulée à Nicéphore Cité (Chalon-sur-Saône, France) et laboratoire LaSTIG (Laboratoire en sciences et technologies de l'information géographique) du Service de la Recherche de l'Institut National de l'Information Géographique et Forestière (IGN).

Nicéphore Cité
34 quai Saint-Cosme
71100 Chalon-sur-Saône FRANCE
Téléphone : 03 85 42 06 55

Laboratoire LaSTIG
Institut National de l'Information Géographique et Forestière
73 avenue de Paris
94165 Saint-Mandé cedex FRANCE
Téléphone : 01 43 98 80 00

Abstract

Content-Based Image Retrieval (CBIR) is a discipline of Computer Science which aims at automatically structuring image collections according to some visual criteria. The offered functionalities include the efficient access to images in a large database of images, or the identification of their content through object detection and recognition tools. They impact a large range of fields which manipulate this kind of data, such as multimedia, culture, security, health, scientific research, etc.

To index an image from its visual content first requires producing a visual summary of this content for a given use, which will be the index of this image in the database. From now on, the literature on image descriptors is very rich; several families of descriptors exist and in each family, a lot of approaches live together. Many descriptors do not describe the same information and do not have the same properties. Therefore it is relevant to combine some of them to better describe the image content. The combination can be implemented differently according to the involved descriptors and to the application. In this thesis, we focus on the family of local descriptors, with application to image and object retrieval by example in a collection of images. Their nice properties make them very popular for retrieval, recognition and categorization of objects and scenes. Two directions of research are investigated:

Feature combination applied to query-by-example image retrieval: The core of the thesis rests on the proposal of a model for combining low-level and generic descriptors in order to obtain a descriptor richer and adapted to a given use case while maintaining genericity in order to be able to index different types of visual contents. The considered application being query-by-example, another major difficulty is the complexity of the proposal, which has to meet with reduced retrieval times, even with large datasets. To meet these goals, we propose an approach based on the fusion of inverted indices, which allows to represent the content better while being associated with an efficient access method.

Complementarity of the descriptors: We focus on the evaluation of the complementarity of existing local descriptors by proposing statistical criteria of analysis of their spatial distribution. This work allows highlighting a synergy between some of these techniques when judged sufficiently complementary. The spatial criteria are employed within a regression-based prediction model which has the advantage of selecting the suitable feature combinations globally for a dataset but most importantly for each image. The approach is evaluated within the fusion of

inverted indices search engine, where it shows its relevance and also highlights that the optimal combination of features may vary from an image to another.

Additionally, we exploit the previous two proposals to address the problem of cross-domain image retrieval, where the images are matched across different domains, including multi-source and multi-date contents. Two applications of cross-domain matching are explored. First, cross-domain image retrieval is applied to the digitized cultural photographic collections of a museum, where it demonstrates its effectiveness for the exploration and promotion of these contents at different levels from their archiving up to their exhibition in or ex-situ, and their linking with other categories of contents, such as geographical mapping contents. Second, we explore the application of cross-domain image localization, where the pose of a landmark is estimated by retrieving visually similar geo-referenced images to the query images.

Keywords: *Content-based image retrieval (CBIR), Feature extraction, Interest points, Feature combination, Bag-of-Features (BoF), Inverted index, Spatial complementarity, Cultural heritage photography, Data linking, Image exploration, Cross-domain image retrieval, Image-based localization.*

Résumé

La recherche d'images par contenu (CBIR) est une discipline de l'informatique qui vise à structurer automatiquement les collections d'images selon des critères visuels. Les fonctionnalités proposées couvrent notamment l'accès efficace aux images dans une grande base de données d'images ou l'identification de leur contenu par des outils de détection et de reconnaissance d'objets. Ils ont un impact sur une large gamme de domaines qui manipulent ce genre de données, telles que le multimedia, la culture, la sécurité, la santé, la recherche scientifique, etc.

Indexer une image à partir de son contenu visuel nécessite d'abord de produire un résumé visuel de ce contenu pour un usage donné, qui sera l'index de cette image dans la collection. En matière de descripteurs d'images, la littérature est désormais trés riche: plusieurs familles de descripteurs existent, et dans chaque famille de nombreuses approches cohabitent. Bon nombre de descripteurs ne décrivant pas la même information et n'ayant pas les mêmes propriétés d'invariance, il peut être pertinent de les combiner de manière à mieux décrire le contenu de l'image. Cette combinaison peut être mise en oeuvre de différentes manières, selon les descripteurs considérés et le but recherché. Dans cette thése, nous nous concentrons sur la famille des descripteurs locaux, avec pour application la recherche d'images ou d'objets par l'exemple dans une collection d'images. Leurs bonnes propriétés les rendent très populaires pour la recherche, la reconnaissance et la catégorisation d'objets et de scènes. Deux directions de recherche sont étudiées:

Combinaison de caractéristiques pour la recherche d'images par l'exemple: Le coeur de la thèse repose sur la proposition d'un modèle pour combiner des descripteurs de bas niveau et génériques afin d'obtenir un descripteur plus riche et adapté à un cas d'utilisation donné tout en conservant la généricité afin d'indexer différents types de contenus visuels. L'application considérée étant la recherche par l'exemple, une autre difficulté majeure est la complexité de la proposition, qui doit correspondre à des temps de récupération réduits, même avec de grands ensembles de données. Pour atteindre ces objectifs, nous proposons une approche basée sur la fusion d'index inversés, ce qui permet de mieux représenter le contenu tout en étant associé à une méthode d'accès efficace.

Complémentarité des descripteurs: Nous nous concentrons sur l'évaluation de la complémentarité des descripteurs locaux existant en proposant des critères statistiques d'analyse de leur répartition spatiale dans l'image. Ce travail permet de mettre en évidence une synergie en-

tre certaines de ces techniques lorsqu'elles sont jugées suffisamment complémentaires. Les critères spatiaux sont exploités dans un modèle de prédiction à base de régression linéaire, qui a l'avantage de permettre la sélection de combinaisons de descripteurs optimale pour la base considérée mais surtout pour chaque image de cette base. L'approche est évaluée avec le moteur de recherche multi-index, où il montre sa pertinence et met aussi en lumière le fait que la combinaison optimale de descripteurs peut varier d'une image à l'autre.

En outre, nous exploitons les deux propositions précédentes pour traiter le problème de la recherche d'images inter-domaines, correspondant notamment à des vues multi-source et multi-date. Deux applications sont explorées dans cette thèse. La recherche d'images inter-domaines est appliquée aux collections photographiques culturelles numérisées d'un musée, où elle démontre son efficacité pour l'exploration et la valorisation de ces contenus à différents niveaux, depuis leur archivage jusqu'à leur exposition ou ex situ, ainsi que leur interconnexion avec d'autres catégories de contenus, tels que ceux de l'information géographique. Ensuite, nous explorons l'application de la localisation basée image entre domaines, où la pose d'une image est estimée à partir d'images géoréférencées, en retrouvant des images géolocalisées visuellement similaires à la requête.

Mots Clés: *Recherche d'image par contenu, Extraction de caractéristiques, Points d'intérêt, Combinaison de caractéristiques, Sac de mots, Index inversé, Complémentarité spatiale, Photographie du patrimoine culturel, Interconnexion de données, Exploration d'images, Recherche d'images inter-domaines, Localisation basée image.*

To my parents

Acknowledgements

I take this opportunity to express my sincere gratitude to Dr. Valérie Gouet-Brunet for guiding and sailing me through this long and enduring journey, called PhD. I feel fortunate to have her as my supervisor who helped me to learn not only the technical aspects but also the integrity of this degree. I am grateful to Gabriel Bloch, my co-supervisor, for his constant guidance and help through the entire process. I am very much thankful to my colleagues at Nicéphore Cité, On-Situ, and Nicéphore Niépce museum for their support. I would like to thank Nicéphore Cité, Institut national de l'information géographique et forestière (IGN) and Agence Nationale de la Recherche (ANR) for financial support.

To my friends and colleagues at MATIS, IGN - thank you for your kind and warm help. Of course, not the least, my friends across the globe - thank you for being with me throughout the journey.

Finally, to my family - you are my inspiration. You always motivate and encourage me. Thank you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context

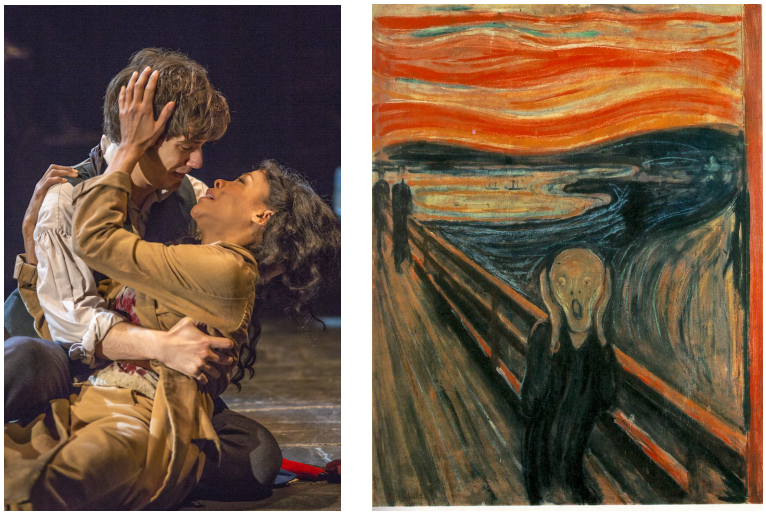
'A picture is worth a thousand words.'

We have all heard the above cliché and it is true. We, humans, are visual creatures. There is a real value in using images. Images grab our attention quickly, it helps us to understand complex concepts and algorithms, it helps us to portray beautiful sceneries, horrific incidents or joyous expressions. It is not invariably possible to describe an incident or event or image just with texts or words. Think about the painting 'Starry Night' by Van Gough or Edvard Munch's 'The Scream' or 'Dog playing Poker' series by Cassius Coolidge (see Figs. 1.1 and 1.2). With images is it always possible to describe the expression or feelings of the actors playing 'Les Misérables' (see Fig. 1.2)? Certain visual impressions are beyond words.

(a) (b)

Figure 1.1: (a) 'Starry Night' by Van Gough (b) 'Dog playing Poker' by Cassius Coolidge.

Another aspect is, a human brain can process visual content much faster compared to text-based communication. Anthropologically speaking, we, humans, started communication over 30,000 years ago, but the first written language was found around 3200 BC. Over a long period of time, we communicated through visual messages, without using written scripts. It helped our brain



(a) (b)

Figure 1.2: (a) Actors playing 'Les Misérables' (b) 'The Scream' by Edvard Munch.

to develop a capacity to process visual data instantaneously. When we consider vision power of humans, we can process images almost instantaneously. Part of our brain can interpret images or familiar objects within 13 milliseconds [Potter et al., 2014], which is pretty fast. When we see an image, we analyze it within a very short snippet of time, knowing the meaning and scenario within it immediately. Thus, images need to be explored as images or searched as images by content, by style.

Image interpretation is one facet. Another important aspect needs to be considered is the humongous volume of media. The volume of multimedia data, such as images, videos, etc., are increasing at an alarming rate due to huge technical advancement in data acquisition, capturing, visualization, etc. To put it into perspective, Instagram[1], an online mobile photo-sharing site, hosts approximately 34.7 billion photos in total and 52 million photos are shared in each day. 300 hours of videos are uploaded on YouTube[2], a video-sharing website, in every minute. The numbers are staggering. The content, the nature, the style, the genre of these images and videos are exceptionally diverse. Therefore, an image organization mechanism, which can efficiently and automatically manage the image databases, is desired.

Not only these are fascinating statistics, at the same time image plays a pivotal role in many ways in our life. Famous American photographer Ansel Adams once said - "Photography, as a powerful medium of expression and communications, offers an infinite variety of perception, interpretation and execution." We all know, some images such as 'The vulture and the little girl' by Kevin Carter or Nick Ut's 'The Napalm Girl' leave a deep impact on us. Images in different formats (digital or print) are used extensively in various other fields such as medical imaging, education, forensic investigation, journalism, visual adverts, cultural restorations, museum archives, etc. Not only that, nowadays authorities in different countries are taking im-

[1] https://www.instagram.com/
[2] https://www.youtube.com/

portant decisions for the betterment of citizen's life by analyzing images shared on social sites [Rudinac et al., 2017]. Thus, there is an expanding demand for efficient image search, analysis techniques.

In recent years, museums, media archive companies have conducted large pictures digitization campaigns. The rapid development of means of production of image digitization generates a huge amount of images which are difficult to manage manually. From research to mining through the data management, image databases are becoming more and more complex and dependent on the end users. In this context, the indexing of images is an essential stage in the life of a database. Indexing means linking meta data or text tags or short description with each image with the aim of using them to retrieve this content quickly for a given purpose. However, this task remains long and complicated in its implementation. In the most scenarios, the process is manual. Therefore, it depends on the professional or cultural of the individual in-charge. Standard of indexing and thesaurus used in the image databases or archives are different from those of archivists. These two scopes are unknown to the end users. These factors often create a couple of problems in image search services which are designed on traditional text tag-based search. First, the interpretation of texts could be different from one person to another. When the user is searching for images of 'Apple' using text as a query, the query can be interpreted either as a fruit or electronic gadgets produced by Apple Inc. The text-based image search engines look for the similar keywords of the query to the associated with images in their database and return the images with matched keyword. This could introduce irrelevant search outcomes, which are not desired by the user. Second, the text-based search depends on the image indexing which relies on annotation or meta data associated with each database image. The annotation part is a normally manual process, which requires human labor. Manual annotation is time consuming process and it may incorporate indecorous and irrelevant descriptions or at least, descriptions usable for a given task and not for another one. Hence, an alternative image search tool, which is capable of capturing user's actual search intention by visually analyzing the search specification, is required.

Usually, image archive companies or photographic museums keep complete archives of photographers collections digitized versions, such as negatives, contact sheets, prints. The archives can receive these formats of images of the same collection at the different times. The primary task of the archivist is to reorganize the collection and match each format of the archives to other format and classify them systematically. The classification task is difficult to manage and time consuming process as the most of the work is done manually. Not only that, photographer's collections often use in different print media, such as magazines, books, newspapers, etc. Archive companies also preserve the digitized version of the print media. As shown in the Fig. 1.3, different versions, photo contact sheet, print photos and photo used in the magazine, of the same photo or similar scene is found in the archive of Nicéphore Niépce Museum[3], which has a collection of more than hundreds of thousands digitized images. To organize and linking

[3]Nicéphore Niépce museum website: http://en.museeniepce.com

these photos could take several of hours of manual work of exhaustive matching and exploration through different sources. Hence, the manual organization process is not viable and efficient, considering the huge volume of images.

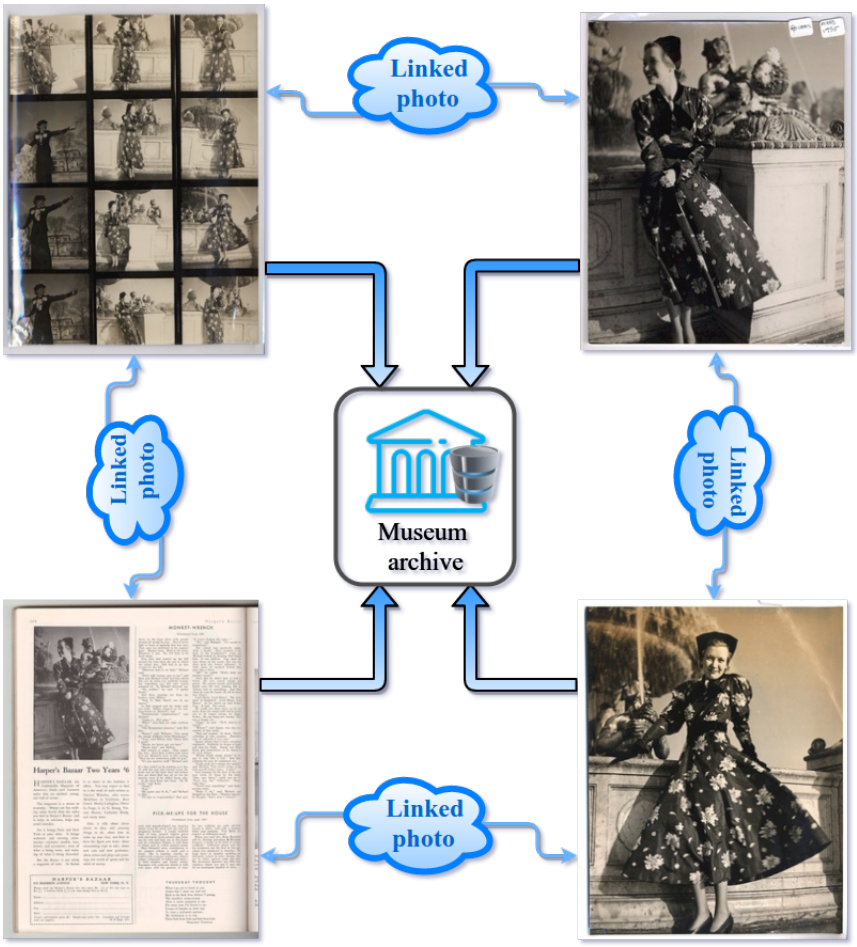


Figure 1.3: Different versions of the same photo or similar scene in the collection of Nicéphore Niépce museum.

Given this background, enhancement of image digitization, for example, publishing or through the organization of an exhibition, creates numerous exchanges between interlocutors of different professions, such as museum curator, editor, picture editor, magazine editor, archivist, etc. Everyone has his/her respective needs. Today, there are no common tools that meet the needs of all potential actors, to facilitate a collaborative work. Hence, an integrated tool for searching, exploration and organization is required to analyze images by content.

Thus, with the hasty growing of the media collection, it is imperative to concentrate on the development or improvement of a search process, such as Content-based Image Retrieval (CBIR) to access voluminous, complex and unstructured visual data efficiently.

1.2 Topic definition

Basically CBIR is used to retrieve the most relevant images which are similar to a query image. In order to pursue to build an intelligent CBIR system which is capable of managing media like a human does, several types of research have been conducted over the time and several paradigms are proposed. Though the vision based research is not new, the initial groundwork has started in the early 1960s. The motivation and fascination behind vision based research are to develop an automated system which significantly increases the efficiency and productivity of the industry. The first vision based automated machine has been built in 1973 [Kashioka et al., 1976] to assemble semiconductor devices. It is one of the most successful computer vision applications due to the intricacies and precisions involved in the mass production of semiconductors, and undeniably object recognition is one of the important modules of vision based applications.

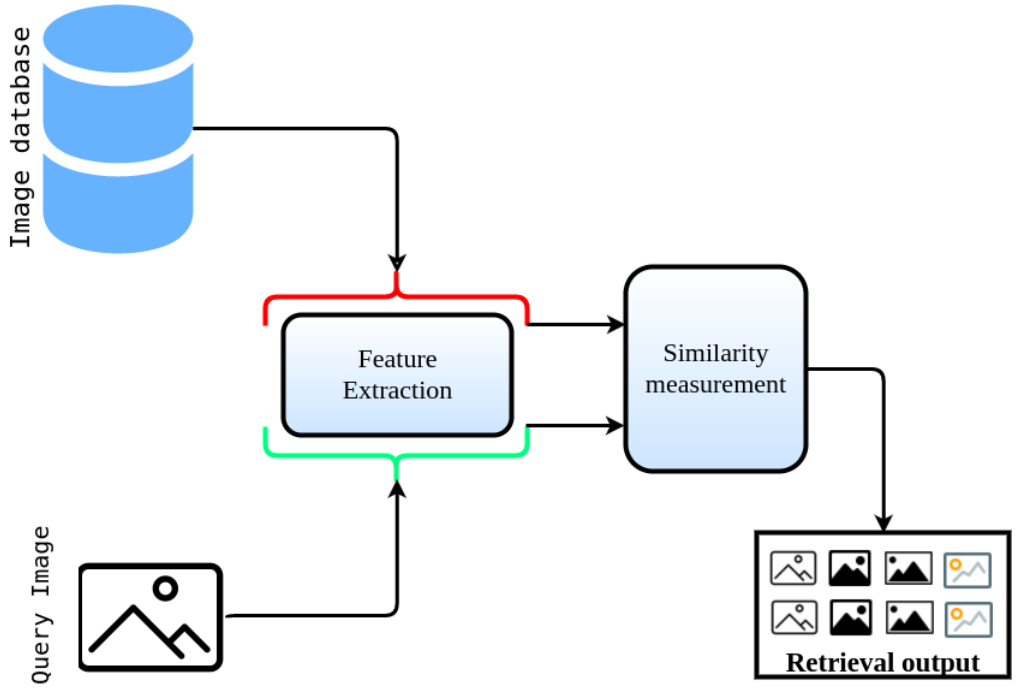


Figure 1.4: A typical scheme for content-based image retrieval (CBIR).

The basic idea of CBIR is to extract the distinguished features (*e.g.,* color, shape, texture, etc.) from the database and then measure their resemblance from the ones of a query or a model, etc. A typical architecture of CBIR is depicted in the Fig. 1.4. There are different models available in CBIR. In 'Query-by-example' (QbE) paradigm [Jing et al., 2004], an example image, which also acts as a query, is provided to the CBIR system in order to search for similar images from the database. In semantic or text-based model [Barnard et al., 2003], the texts or phrases or sentences are used as a query. In hybrid text-visual search model [Bassil, 2012], the query is represented by both texts and example images. In supervised learning approach [Zhang et al., 2011b], which is more about image classification, the machine learning techniques are used for learning to determine image classes. Relevance feedback (RF) [Crucianu et al., 2008] is another

CBIR paradigm, where the user is involved in the retrieval process. The user provides positive or negative feedback to decide the relevancy of the retrieved images and it refines the search result progressively. Among these different CBIR paradigms, QbE is the most popular in CBIR. QbE paradigm is used in various computer vision applications, such as in image exploration, photograph archiving, cross-domain image retrieval, image-based localization, remote sensing, medical applications, etc. This thesis focuses on the QbE paradigm for CBIR.

In QbE paradigm, to index an image from its visual content, it requires producing a visual summary of this content for a given use, which will be the index of this image in the database.

The feature descriptions may become larger in dimension and volume depending on the image contents and the characteristics of the descriptors. This phenomenon is known as 'curse of dimensionality', which was first used in the work of [Bellman, 1961]. It can be defined as follows: the number of descriptions, which are important generate definitive image representation, grows exponentially along with the dimension. For large databases, the high-dimensional representation is not preferable for image retrieval due to several reasons. First, the storage requirement will increase. Second, the image retrieval process will become computationally very expensive and time-consuming. Thus, the curse of dimensionality problem is addressed by introducing several dimensionality reductions approaches, such as principal component analysis [Schölkopf et al., 1998], linear discriminant analysis [McLachlan, 2004] or feature representation models, such as the bag of features (BoF) [Sivic and Zisserman, 2003], Fisher kernel [Jégou et al., 2010], etc. Along with that, different indexing structures, such as inverted indexing [Sivic and Zisserman, 2003], tree-based indexing [Silpa-Anan and Hartley, 2008], joint inverted indexing [Xia et al., 2013], etc., are incorporated to speed up the database access during image retrieval process.

Several surveys [Rui et al., 1999; Smeulders et al., 2000; Müller et al., 2004; Datta et al., 2008; Hu et al., 2011; Alzu'bi et al., 2015] have been conducted for image retrieval due to its large range of applications. At the same time, there is no standard retrieval framework for CBIR exists. Thus, research in the CBIR domain is widespread and expanding in breadth rapidly.

To put in perspective, statistical estimations on published article on CBIR are shown in the Fig. 1.5 and Fig. 1.6. We looked for publications on CBIR in major publication databases: IEEE[4], Elsevier[5], Springer[6], ACM digital library[7]. We limited our search with the keywords combinations 'Content-based image retrieval' and 'CBIR'. The articles, which contain both keywords, are included in these statistics. As observed from the Fig. 1.5, approximately 2000 articles have been published during 2000 and 2009 and over 2500 articles are published in between 2010 and 2016. Although, this is not a pin-point statistics as we only consider particular two keywords, not others such as, 'image matching' or 'image search'. Having said so, by looking at the trend,

[4]http://ieeexplore.ieee.org/
[5]https://www.elsevier.com/
[6]http://www.springer.com/
[7]http://dl.acm.org/

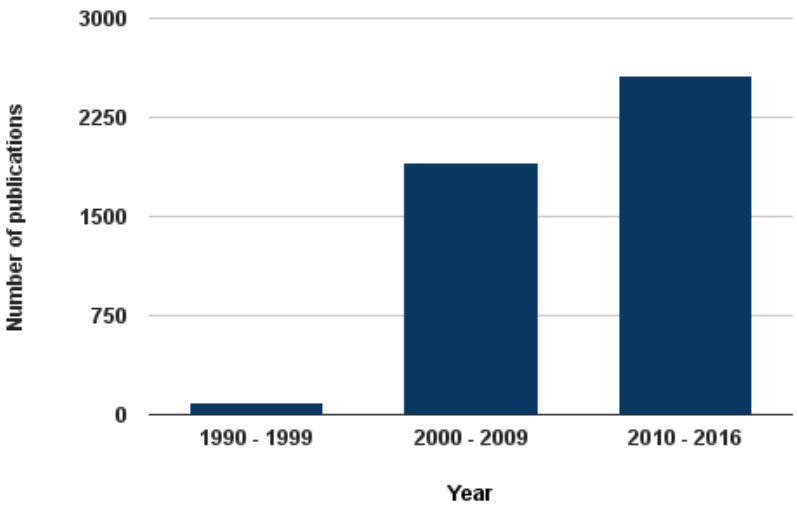


Figure 1.5: Article published in recent years on CBIR.

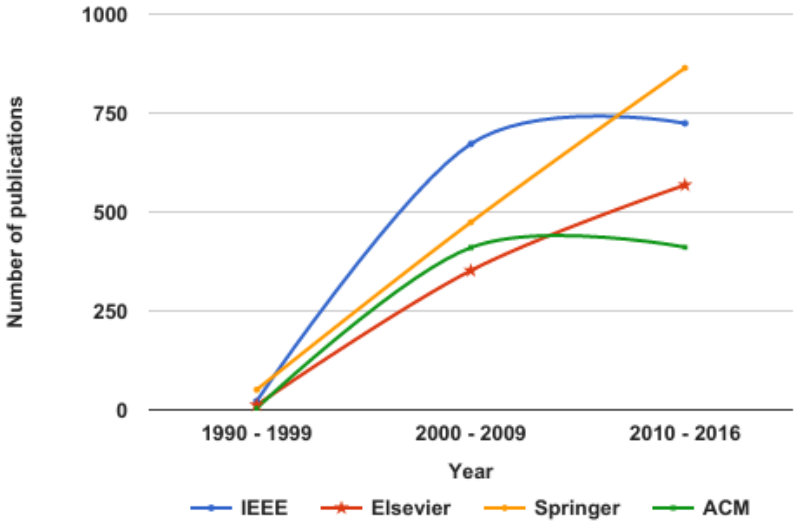


Figure 1.6: Article on CBIR published in different journals.

we can predict that by the end of this decade, this number will be increased by two-folds compared to the last decade. Additionally, in the Appendix A, we list down all notable CBIR search engines/tools available publicly. The CBIR engines/tools are proposed either for commercial use or for research purpose. Among them, 20 such CBIR tools are available for research purpose and these are developed either in the research institutes or in the universities. The commercially available CBIR tools are developed by either privately or publicly held companies.

Although there are several CBIR approached are proposed in the literature, still, there are certain

open issues exist. These issues need to be discussed and reviewed before we put forward our proposal. Let us present the key topics of the thesis in the following.

Single feature represenation or multiple features representation:

Feature extraction from the images is one of the first and key steps in any CBIR approach [Gudivada and Raghavan, 1995; Rui et al., 1998, 1999; Smeulders et al., 2000; Müller et al., 2004; da Silva Torres and Falcão, 2006; Hu et al., 2011; Wan et al., 2014]. The key emphasis is on describing suitable image characteristics, which should coincide with the user's vision and perception of similarity of the images (*i.e.,* to account for the gap between low and high-level semantic concepts), classically known as 'semantic gap'. Over the period of time, considerable research works have conducted to propose different image characteristics [Li and Allinson, 2008; Tuytelaars and Mikolajczyk, 2008; Gauglitz et al., 2011; Mukherjee et al., 2015], such as shape, texture, deep features. In the context of CBIR, we are interested in conventional, also known as hand-crafted, local image features. Local features are well known for their genericity and their robustness to image transformations.

The important question arises at this point: are single feature description is sufficient to represent image contents or there is a need to combine multiple feature descriptions to better encapsulate the essence of the image?

Images can be described with conventional features by two ways:

First, global features which capture the overall essence of the content of the image and represent it with a single feature vector. Normally, color [Gevers and Stokman, 2004], texture [Krishnamachari and Chellappa, 1997] and shape [Gevers et al., 2004], characteristics are considered to construct global feature [Chang et al., 2001].

Second, local features refer to a distinct patterns which are different from its surroundings. Shape, texture, intensity, etc. are considered to define a distinct characteristic which can be a point or small image patches. In general, the local feature extraction is two step process. First, the distinguished image patches or points are identified by the local detector. Then these detected points or regions are described by the local image descriptor. The descriptions, in form of vectors, are the representation of the image content. The advantages of this feature are to invariance to the geometry, such as rotation and scale, occlusion, change in viewing condition, illumination changes. Also, the local feature generation process is computationally efficient. Thus, it can be useful for robust image representation. This leads to the usage of local features in a variety of computer vision applications, such as in CBIR, object detections, image registration, classification, motion estimation, tracking, 3D reconstruction, etc.

In this work, we do not concentrate much on the recently developed deep learning features [Sermanet et al., 2013; Oquab et al., 2014], which is a whole new ball game. Although we revisit this type of features in the Chapter 2.

All these local image descriptors have their respective advantages and drawbacks depending on the targeted applications. For example, scale invariant feature transformation feature [Lowe, 2004] is suitable for object, scene matching and recognition related applications. When it comes to pedestrian detection task, histogram of oriented gradient feature [Dalal and Triggs, 2005] performs better than scale invariant feature transformation features. These are two different applications. Understandably, features are developed in the literature to perform some specific tasks.

Now, consider only image retrieval related tasks. Nowadays, image databases are very diverse in nature and contain heterogeneous image contents, such as scenery, facades, textures etc. All the image descriptors do not pose the same properties, hence they do not describe the same information. To give it in perspective, for similarity search task, where the image contains objects or scene, shape contexts (SC) [Belongie et al., 2002] produces high performance. However, if the image content is texture type, then the same SC descriptions do not perform reliably. Instead, local binary pattern (LBP) [Ojala et al., 1996] tends to perform better.

From now on, the literature on local image descriptors is very rich. Several families of local descriptors exist and in each family, a lot of approaches live together, with different properties. As a consequence, they do not describe the same information from an image. Therefore, fusion or combination of local image descriptors may be propitious to better describe the image content for image retrieval related tasks. This leads to the core of this work, *i.e.,* feature fusion strategy in CBIR. It has already been demonstrated that combining different descriptions is propitious to better describe image contents in image retrieval related applications [Hiremath and Pujari, 2007; Chatzichristofis et al., 2010b; Zhang et al., 2012b; Choudhary et al., 2014] due to better performances compared to using a single feature.

Strategy for feature fusion:

In the thesis we focus on the proposal of a strategy for combining low level and generic descriptors in order to obtain a descriptor of higher semantic level adapted to a given use case while maintaining genericity in order to be able to index different types of visual contents.

The emerging concerns at this point are:

1. The availability of a generic strategy of the feature fusion in the literature, and
2. How efficiently the fusion strategies can deal with the curse of dimensionality problem in the context of large volume of image content representation.

Several descriptor fusion approaches exist in the literature of CBIR (see Chapter 2). The two main observations we can make in the literature on the fusion strategies are:

First, fusion can take place at the beginning of the process, i.e., just after the feature extraction from the images and before starting the image retrieval process. This type of fusion strategies

is termed as early fusion. Usually, this strategy refers to the combination of the features into a single representation by feature concatenation before similarity comparison step. One of the advantages of this type of fusion approach is that the image retrieval related steps are performed only once. However, there are certain complications involved:

1. The single representation of the combined multiple features requires a considerable amount of memory, as it generates the very high dimensional feature vector. This is not desirable in many image retrieval tasks due to the accretion in retrieval time.

2. The representation of the different descriptor vectors is not the same, *i.e.,* range of the description vectors differs from one to another. Therefore, it induces the problem of combining multiple features into a common feature space.

These drawbacks also restrict the usage of the multiple numbers of features to combine during retrieval.

Second, fusion can take place at the last stage of the entire retrieval process. This type of fusion strategies is categorized as late fusion. In late fusion, the final retrieval output can be achieved by aggregating the combined multiple ranked outputs of the multiple descriptors. In general, either ranked-based or score-based late fusion strategies are applied to result in the final response. As each individual feature type is used separately for retrieval, the high dimensionality of the features does not constitute issues, contrary to the previous fusion methods. The deficiency of this type of fusion strategies lies in the expensive computational cost due to the involvement of the multiple retrieval steps for each feature type. Hence, there is a slight restriction on the number of the features to fuse.

Thus, in this work, a simple yet powerful feature fusion strategy is proposed. Considering the target application is query by example in content-based image retrieval, we consider following crucial aspects:

1. Robust and generic feature fusion strategy to accommodate multiple features considering the diverse nature of image contents.

2. How to tackle the curse of dimensionality problem in order to reduce the complexity of the fusion strategy and efficiency towards retrieval time.

3. Scalability of the proposal bearing in mind different applications within the context of image retrieval.

Among all the existing solutions for describing image contents and organizing the extracted features in order to deal with large dataset, we propose a feature fusion strategy called 'Fusion of Inverted Indices' (FII). Considering the cons and pros of the different fusion strategies, which was discussed earlier, we have chosen a middle ground for fusion, *i.e.,* intermediate fusion.

Inverted multi-index data structure [Babenko and Lempitsky, 2012] suggests a data structure to combine multidimensional features efficiently. It decomposes the image descriptor space into n desired subspaces. Then, the best responses to a query in each subspace are retrieved and combined into one response that ensures better result than the traditional approaches based on classical inverted indices [Sivic and Zisserman, 2003]. We propose a novel fusion method for efficiently combining multiple descriptors for image retrieval, based on the inverted multi-index data structure. The proposed fusion strategy allows combining any number of multidimensional image descriptors by integrating their responses to a query in finer subdivisions.

In general, the descriptors combined are selected a priori. They are selected for a particularly targeted task or for a given database based on their presupposed compatibility. In this scenario, compatibility implies whether the selected features represent the complementary information about the image or not. Thus, a complementarity evaluation between the features would be profitable before evaluating the whole similarity search engine at large scale.

Descriptors fusion based on spatial complementarity evaluation:

We have already discussed that a substantial number of local detectors and descriptors are proposed in the literature of computer vision and CBIR. Each of this descriptor has respective advantages and drawbacks. The diversity in the local feature descriptors makes it arduous to determine the most relevant descriptors for a given application and a given dataset. Therefore, the concern arises during feature fusion: how do we determine the most befitting descriptors to combine for image retrieval related tasks?

We focus on local image descriptors, where the extraction of feature points plays an imperative part in the process. The existing different local descriptions used in the fusion strategies may not have similar importance due to the diversity in the image contents and the attributes of the descriptors. Without considering the complementarity information during the fusion could lead us to the following problem. During feature extraction, different detectors might detect the very similar points or regions from an image. These interest points are described by descriptors and then the combined descriptions represent the images. As the combined representation might consist of repeatable and similar descriptions, it does not fulfil the true purpose of the feature fusion, *i.e.,* represent the images by combining distinctiveness information. Thus, it is imperative to consider complementarity information.

In the most of the existing fusion strategies, the complementarity between the descriptors are largely ignored. The descriptors combined are chosen a priori, according to their presupposed complementarity. This assumption may not effective for different image retrieval scenario. Also, in the existing fusion works, the descriptors combination is considered globally for an entire database. It ignores the fact that each image content may be different. As a consequence, the features are not combined locally for each image. Therefore, we believe, a framework is required to evaluate the effectiveness of given descriptors on a specific dataset. It is also

possible to learn the best combination of given descriptors from a representative dataset. We think that it is important to appraise the complementarity between such local features. This part of work focuses on the spatial complementarity of the detected interest points in the image, by exploiting statistical criteria of spatial analysis, in order to give the possibility to combine several descriptors.

Several spatial complementarity criteria of the local feature detectors (which measure different properties of the detectors) are reviewed in the 4. Among them, three suitable criteria, *e.g.,* spatial distribution [Ehsan et al., 2013], contribution [Gales et al., 2010] and cluster-based measure [Mikolajczyk et al., 2005], are selected for the evaluation. We propose a regression model that involves these complementarity measurement criteria of spatial analysis of feature detectors. The regression model is integrated into the FII image search framework. Mean average precision (mAP), which is the evaluation measure of the quality of the content description, is incorporated to train the regression model and then anticipates the optimal detector combinations not only globally for a new dataset, but for each new query image.

1.3 Contributions of the thesis

The contributions of this work are primarily in two-folds. In the below, we summarize these contributions:

First: Due to the large diversity of existing feature descriptors in content-based image retrieval, the image contents can be better represented by the joint use of several descriptors in order to explore their potentially complementary characteristics. The first part of this work presents and discusses a strategy for the fusion of different multidimensional features involved, based on inverted multi-indices and dedicated to similarity search. Image descriptors are quantized separately and efficiently through dimension reduction techniques, before being combined in the inverted multi-indices. To exhibit its effectiveness, the proposal is evaluated on several datasets having different contents and sizes, facing several state-of-the-art approaches of image descriptor fusion. The obtained results reconfirm that the joint use of several descriptions improves similarity search, and show that our fusion proposal outperforms other solutions while manipulating lower or similar volumes of features.

Second: With a large number of local feature detectors and descriptors in the literature of CBIR, in the second part of this work, we propose a solution to predict the optimal combination of features, for improving image retrieval performances, based on the spatial complementarity of interest point detectors. We review several spatial complementarity criteria of detectors and employ them in a regression based prediction model, designed to select the suitable detectors combination for a dataset. The proposal can improve retrieval performance even more by se-

lecting optimal combination for each image (and not only globally for the dataset), as well as being profitable in the optimal fitting of some parameters of image search engine. The proposal is appraised on three state-of-the-art benchmarks to validate its effectiveness and stability. The experimental results highlight the importance of spatial complementarity of the features to improve retrieval and prove the advantage of using this model to optimally adapt detectors combination and some parameters.

The global overview of this work is depicted in the Fig. 1.7. Three different parts of this work are presented and discussed in the upcoming chapters.

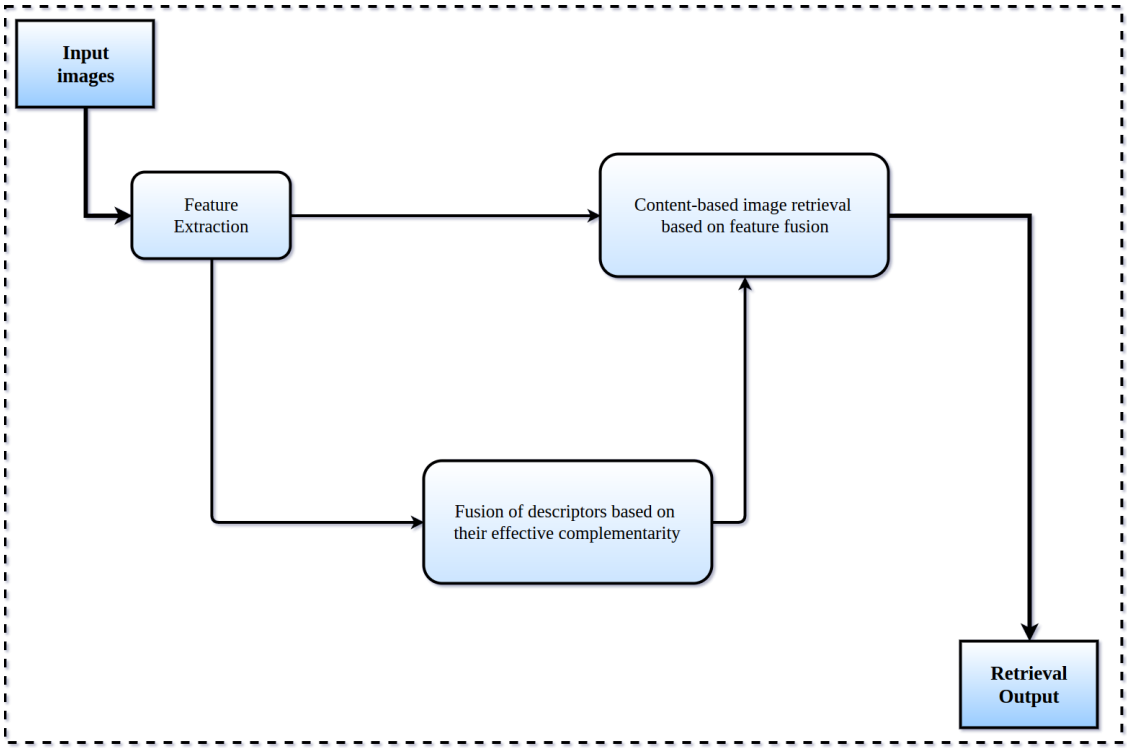


Figure 1.7: Global overview of the thesis proposal.

Additionally: We apply our contributions to the problem of cross-domain image retrieval. Cross-domain matching or retrieval can be defined as the visual similarity matching between two characteristically different images. In cross-domain image retrieval, the query images may be different in characteristics from the database images, such as postcards vs. street view images. Nowadays, the image datasets are becoming more versatile and complex. The image archive companies, museums collect a wide range of images which are acquired by different sources or modalities and at different times. Therefore, the retrieval of visually similar images by CBIR across different image domains poses a challenging task. The successful strategy to deal with this problem depends on several factors. Definitely, the selection of the suitable features based on the image content is very important due to the fact that the similarity comparisons are conducted between two different characteristics of images. We focus on the two main applications of cross-domain retrieval.

1. Image exploration in the photographic museum collections: The FII image search framework is applied to the cultural photographic collection of a French photographic museum, Musée Nicéphore Niépce. It demonstrates its added value for the exploration and promotion of these contents at different levels from their archiving up to their exhibition *in* or *ex situ* by intra/inter linking the image contents.

2. The problem of cross-domain image localization, *i.e.,* the ability to estimate the pose of a landmark from visual content acquired under various conditions, such as old photographs, postcards were taken at a particular season, etc. by using a CBIR framework. The ability to localize cross-domain content opens the opportunity to inter linking the content across different domains and to improve its added value within contents, frameworks and applications relying on a geographical location in the environment.

1.4 Thesis organization

The rest of the thesis is structured in five chapters. The contents are summarized as bellow:

Chapter 2: This chapter is dedicated to the literature review on CBIR with a focus on the particular topis addressed in the thesis. It begins with the discussion on the first step of CBIR, *i.e.,* feature extractions from the images. The primary focus is on the local image features. The discussion continues with the presentation of the representation models of the extracted features. It is followed by a detailed presentation on the feature fusion strategies in CBIR.

Chapter 3: This chapter presents the detail proposition of the CBIR search engine based on feature fusion for query by example application. The core of the thesis rests on the proposal of a model for combining low level and generic descriptors and adapted to a given use case while maintaining genericity in order to be able to index different types of visual contents. We introduce the concept of different inverted indexing structures. The implementation details of the proposal, *i.e.,* fusion of inverted indices image search engine, is presented afterwards. The experiments and evaluation details are presented before concluding the chapter.

Chapter 4: The main focus of this chapter is to predict the optimal combination of detectors, for improving image retrieval performances, based on the spatial complementarity of interest point detectors. After introducing the concept, several spatial complementarity criteria of detectors are reviewed. These criteria are employed in a regression based prediction model which is designed to select the suitable feature combinations for each image and as well as globally for a dataset. This is followed by several experiments and evaluation which highlight the importance of adaptive selection of the features and parameters for fusion of inverted indices search engine to improve the image retrieval performance.

Chapter 5: This chapter is dedicated to the cross-domain image retrieval and its applications. The chapter begins with the introduction to the cross-domain terminology and how the cross-domain image retrieval tasks can be addressed by the fusion of inverted indices search engine with query adaptive feature fusion framework, which is presented in the Chapters 3 and 4. It continues with the presentation of the experiments and evaluations, which are conducted on the cross-domain dataset. Once the effectiveness of the proposal is established, one of the applications, *i.e.,* image exploration in the photographic museum collections by intra and inter linking the image contents, is presented. The fusion of inverted indices image search engine demonstrates its effectiveness for the exploration and promotion of the museum collections. Several cross-domain image retrieval scenarios in the context of museum collections are explored and presented in this chapter. It follows by presenting another application, *i.e.,* cross-domain image localization by retrieving visually similar images using proposed query adaptive image search engine. We explore several image-based localization examples, where we inter link the image contents, using geo-referenced image dataset before concluding this chapter.

Chapter 6: This chapter concludes this work with the global assessment and provides short, medium and long term perspectives for this work.

Appendices: This thesis consists of several appendices. These appendices illustrate the different important steps of the proposal with examples.

Appendix A presents a list of CBIR tools available for commercial and research purpose.

Appendix B presents several examples of the main steps of the fusion of inverted indices search engine, which is proposed in the Chapter 3. The appendix is organized as follows:

Appendix B.1 presents the example of the creation of inverted unique indices.

Appendix B.2 is for the example of search k-nearest neighbor.

Appendix B.3 describes candidate list creation example.

Appendix B.4 is dedicated to the example of MultiSequence pair algorithm.

Appendix B.5 focuses on the voting example.

Appendix C presents the spatial complementarity evaluation criteria, which is presented in the Chapter 4. The organization of this appendix is as follows:

Appendix C.1 presents the spatial coverage complementarity examples.

Appendix C.2 illustrates the examples of contribution measure between interest points detectors.

Appendix C.3 is dedicated for the cluster based complementarity measurement examples.

Appendix D presents the list of articles published in the journal, conferences, etc.

Chapter 2

Related work on content-based image retrieval

2.1 Introduction

We are witnessing a huge upsurge in the volume of multimedia collection such as images, videos, etc. At the same time, the multimedia data are increasingly complicated as we are living in a massively expanding digital world. Thus the focus is on developing an effectual content based image retrieval (CBIR) system to manage and organize the voluminous, complex and unstructured data. The goal of any CBIR is to retrieve visually images similar to a query from image database. We are interested in QbE paradigm in CBIR. QbE model in CBIR finds similar images by analyzing only the content of the images, without considering metadata associated with images.

A typical architecture of QbE image retrieval and its different components are depicted in the Fig. 2.1. Several distinguished image features are extracted from the database and the query images, and the similarity are measured between the images. The features could be shape, color, texture or other information which can be used to represent an image.

In general, several steps are involved in any image search engine, such as offline process and online process:

- Offline process: This is related to the preparation of the image database. This included several pre-processing steps. These steps depend on the target applications and the methodologies or algorithms used in the image retrieval systems. To begin with, noise removal from the images, resizing, rescaling of the images, image segmentation, salient region detection could be the part of the pre-processing steps. As stated before, feature extraction from the images is one of the important steps and also the key step in the offline process. The characteristics of the database images can be represented by different image

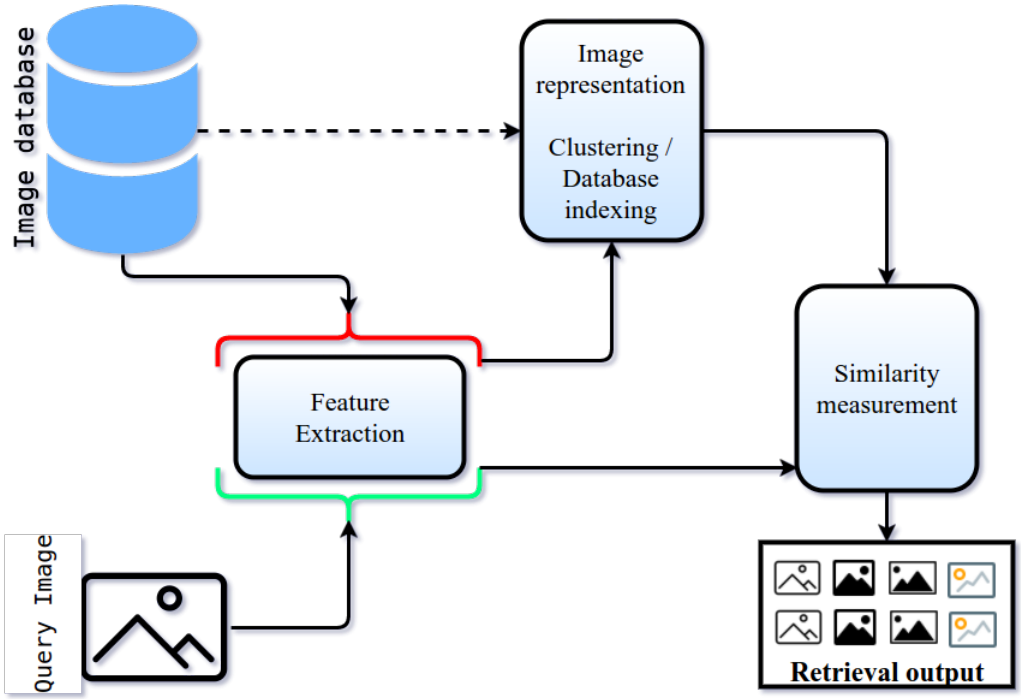


Figure 2.1: Different components in typical CBIR schema dedicated to query by example retrieval.

features, such as local features [Lowe, 2004; Morel and Yu, 2009; Rublee et al., 2011], global features [Turk and Pentland, 1991; Zhuo et al., 2013], or deep features [Sermanet et al., 2013; Donahue et al., 2013; Zeiler and Fergus, 2014]. These extracted features are used to measure the similarity with the features extracted from the query images. However, the exhaustive search for each feature in the database and the query may not be the efficient solution, considering the huge volume of images needs to be tackled. Additionally, to reduce the semantic gap between high-level concept and machine described features, the features tend to larger in size and dimension. Therefore, to reduce the dimensionality, the features are clustered or represented by different models, such as bag of features [Sivic and Zisserman, 2003], BossaNova [Avila et al., 2013], manifold learning [Gashler et al., 2007], Fisher kernel [Rahulamathavan et al., 2013], bag-of-bags of words [Ren et al., 2014], etc. Not only that, database indexation is very crucial to accelerate the retrieval process by accessing the database quickly and efficiently. Hence, different indexing strategies, such as inverted index [Sivic and Zisserman, 2003], inverted multi-indices [Babenko and Lempitsky, 2012], hash-based indexing [Shao et al., 2012], latent semantic indexing [Liu et al., 2009], joint inverted indexing [Xia et al., 2013] are proposed in the literature of CBIR. Once the database is ready by performing all or some of these offline steps, the online process comes into the picture.

- Online process: In QbE image retrieval [Carson et al., 2002; da S. Torres et al., 2009; Huang et al., 2010], the online process deals with the query image and its matching with the database images. The extraction of the distinguished features from the query is the initial step. The system compares the query image features with the database features

which are represented by different models. The similarity scores are measured between the query and the database images and the relevant images are retrieved and usually presented as a ranked list. In other CBIR paradigm, such as relevance feedback (RF), considers the user's feedback, while deciding relevancy of the retrieved images to increase the search accuracy. Several RF strategies, such as support vector machine (SVM) based [Zhang et al., 2012a], genetic programing based [Ferreira et al., 2011], etc. are proposed in the literature. In the context of classification, the similarity between the images can be learned by different machine learning algorithms, such as SVM. The use of SVM can be found in the numerous works, such as in [Ji et al., 2013; Zhang et al., 2011b; Aubry et al., 2014].

Over the period of time, several strategies for CBIR have been proposed depending on the target applications. The goal is to invent a more efficient and powerful CBIR system although there is no universally accepted standard strategy. We have seen a huge upsurge in research and it is continuously and strongly growing. We presented a statistics on the published articles in the Sec. 1.2 of Chapter 1. Not only that, CBIR tools, which are available for research and commercial purposes, are numerous and various. They are presented in the Appendix A.

In CBIR system, the images can be represented by single feature or fusion of multiple features. In the Chapter 1, we argued on the use of the single feature and multiple features. Nowadays, literature on image descriptors is very rich [Bay et al., 2008; Tuytelaars and Mikolajczyk, 2008; Sande et al., 2010; Abdel-Hakim and Farag, 2006; Dalal and Triggs, 2005; Alahi et al., 2012; Leutenegger et al., 2011], providing several families to describe different image characteristics. Combining different descriptions is propitious to better describe image contents and this might be advantageous for CBIR. Thus, in this chapter, the key focus is on the different feature fusion strategies in CBIR. We systematically study, review and present the different processes, such as feature extractions, image representation models, fusion strategies, etc., involved in a CBIR system.

type of distance measure that will be used to compare their similarity

This chapter is organized as follows: in the Sec. 2.2, we focus on feature extraction approaches, such as, local, global, convolution features, Sec. 2.3 describes different feature representation models, in the Sec. 2.4, we present different fusion methods in CBIR.

2.2 Feature extraction

Visual feature extraction from the images is the elementary and crucial step in content based image retrieval. The distinguished image characteristics are identified from the local image patches or globally from an image. Then the image content is described or represented by image feature descriptors. The key emphasis is on the design of visual feature descriptors that

encapsulate the image content according to the application. However, it is always difficult to address the semantic gap problem between low and high-level semantic concepts. Therefore, the most arduous task is to propose image features which are proficient in producing illustrious information of the image content. Ideally, the image features should coincide with the user's observation of the particular image. It should also define the similarity measures close to the user's perception of similarity [Datta et al., 2008; Lew et al., 2006; Bay et al., 2008; Belongie et al., 2002].

Over the period of time, precisely since the 1990s, we have seen several strategies of feature extraction in the literature [Tuytelaars and Mikolajczyk, 2008; Datta et al., 2008; Mukherjee et al., 2015; Razavian et al., 2014]. We can broadly categorize these approaches into two main domains:

1. Conventional features and
2. Deep learning features

The conventional features or the hand-crafted features are the most commonly used features where images can be described either low level representation or higher level visual concepts. These type of features are developed or proposed to solve various of computer vision applications such as CBIR, image classification, object detection, face recognition, image localization, 2D-3D reconstructions, etc. The conventional feature types could be further classified as global features and local features.

The later ones, deep learning features, are gradually gaining popularity in recent years to tackle the above-mentioned computer vision tasks. In deep features, feature representation is learned by feature hierarchy using convolution neural networks.

A quick overview of the different types of features is depicted in the Fig. 2.2. In this section, our primary focus is on the conventional features, to be specific - image representation by local features, in the context of CBIR.

2.2.1 Conventional features

In the literature of CBIR, remarkable works have been done on different strategies to extract several features from the images. Images can be described either by low-level features that represent information about the signal, such as color, texture, shapes, regions, interest points, etc. or by higher level visual concepts such as trained and recognized objects or specific objects related to a domain (*e.g.,* faces). Therefore, images can be described with:

1. Global feature: produces one feature vector for all the image content.
2. Local feature: extracts the information on local salient regions from an image.

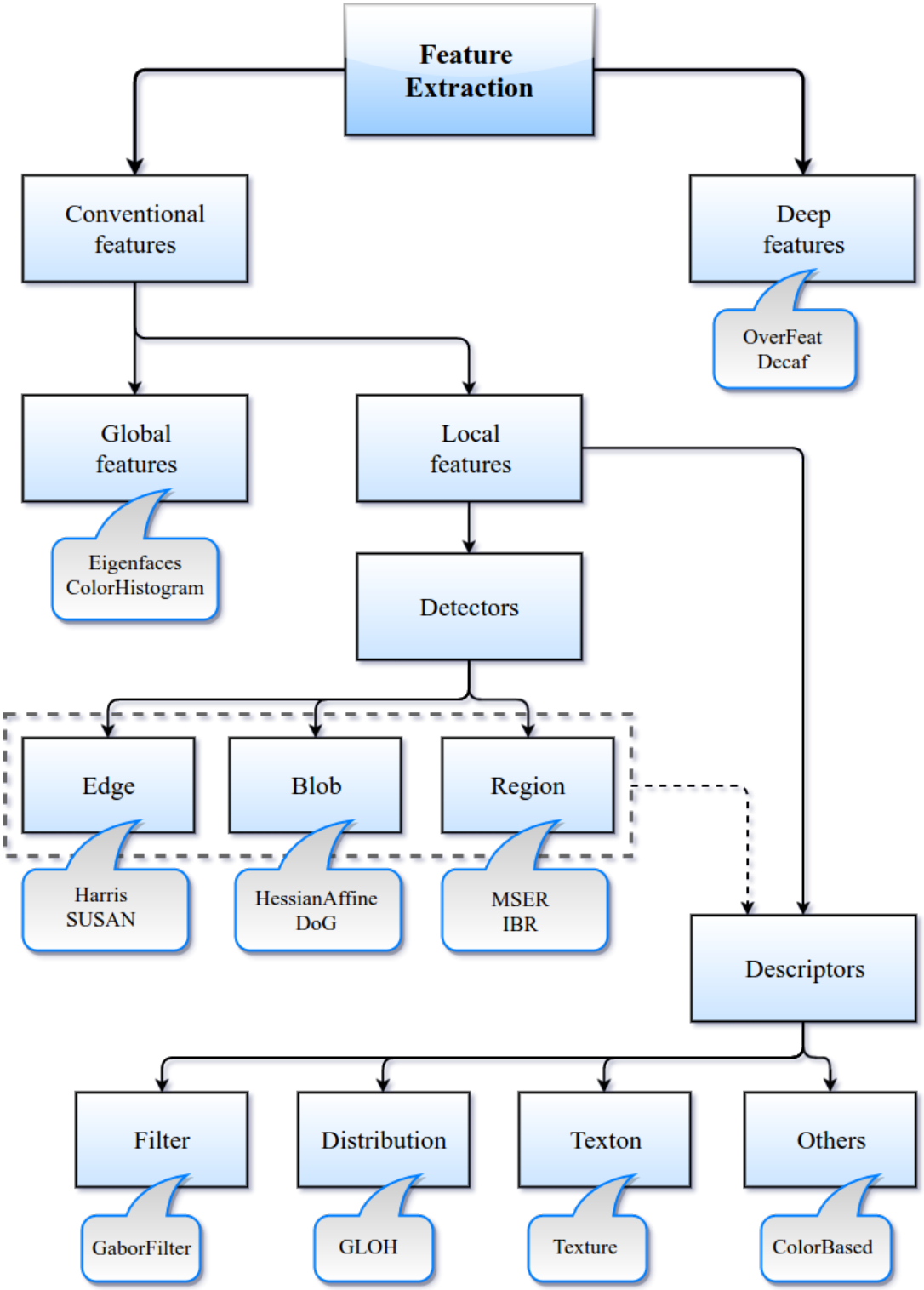


Figure 2.2: Types of features in the literature of content-based image retrieval: OverFeat [Sermanet et al., 2013] Decaf [Donahue et al., 2013], Eigenfaces [Turk and Pentland, 1991] ColorHistogram [Swain and Ballard, 1991], Harris [Schmid and Mohr, 1997] SUSAN [Smith and Brady, 1997], HessianAffine [Mikolajczyk and Schmid, 2004] DoG [Lowe, 2004], MSER [Matas et al., 2002] IBR [Tuytelaars and Van Gool, 2004], GaborFilter [Daugman, 1980], GLOH [Mikolajczyk and Schmid, 2005], Texture [Varma and Zisserman, 2005], ColorBased [van de Weijer and Schmid, 2006].

The scope of the global or the local features depends on the target application. In this section, we briefly introduce global features and then the main focus will be on the image representation by local features.

2.2.1.1 Global representation of the content

A global descriptor describes the image as a whole and it resumes in one feature vector all the image content. Global features include contour representations, shape properties, texture properties, color representation of the content. They have the advantage of well characterizing the main aspect of images and encapsulating some global semantics or ambience [Oliva and Torralba, 2001; Chang et al., 2001]. Global representation of the content requires a low amount of data to describe it as it generates only one high-dimensional vector per image.

Several global descriptors are proposed in the literature [Swain and Ballard, 1991; Oliva and Torralba, 2001; Zhuo et al., 2013; Penatti et al., 2012]. In this context, color indexing [Swain and Ballard, 1991] was proposed for object representation where histogram intersection was introduced for real-time indexing of a large database. A new color indexing strategy is proposed in the work of [Stricker and Orengo, 1995], where the index stores only the dominant features in the color distribution. Cell histogram [Stehling et al., 2003] is another global representation technique where it represents how the color is distributed among the grid of cells. In the work of [Deng et al., 2001], a compact color description is proposed for image retrieval. The colors are clustered, in a region, into a small number of representative colors to represent the image content. Several other global color descriptors are also proposed in the literature, such as in [Nallaperumal et al., 2007; Utenpattanant et al., 2006; Williams and Yoon, 2007; Paschos et al., 2003].

GIST descriptor [Oliva and Torralba, 2001] is one of the well-known global descriptors proposed for scene recognition. In GIST, A set of perceptual dimensions, such as naturalness, openness, roughness, expansion, ruggedness, represents the dominant spatial structure of a scene. Other notable global descriptors [Chang et al., 2001], such as MPEG-7 color layout descriptor (CLD), scalable color descriptor (SCD), edge histogram descriptor (EHD), belong to the MPEG-7 or multimedia content description interface family. As the names suggested, these descriptors represent different properties of the image. For example, CLD represents the spatial color distribution of an image in YCbCr color space. EDH describes the spatial distribution of one non-directional edge and four directional edges in an image. SCD is a color histogram in HSV color space. Some other global features [Turk and Pentland, 1991; Murase and Nayar, 1995; Basilio et al., 2011; Zhuo et al., 2013] are proposed in the literature for different tasks, such as object retrieval, image recognitions, etc. One of the drawbacks of this representation model is the incapability to differentiate foreground from background in the image and often it combines the information from both parts.

2.2.1.2 Local representation of the content

Local features in an image are image patterns which are distinct from its neighborhood. These local patterns or structures can be a point, small image patch, corners.

The literature on the local description of the content is very rich, see for example surveys [Tuytelaars and Mikolajczyk, 2008; Yaghoubyan et al., 2016]. As the local descriptors first usually require to define a local support of extraction of the information in the image, we split this presentation into two parts: first, we discuss the detection of the support in the image, and then we revisit the feature description part.

Local feature detectors: Feature detection is the primary step to obtain local feature descriptors. A detector detects supports in the image, *i.e.,* the most distinctive regions or points, particular pixels in the image. Generally, it is capable of replicating similar echelons of performances to human observers in locating substantial features in an image. Several computer vision applications [Mikolajczyk and Schmid, 2004; Lowe, 2004; Alahi et al., 2012] use feature detection as the primary step. Hence a several numbers of detectors have been developed. These feature detectors can broadly be categorized into the following groups:

1. Edge detectors: It detects points in an image where image brightness changes sharply or in the high curvature region or intersection of edges. Examples are Harris [Schmid and Mohr, 1997], SUSAN [Smith and Brady, 1997].

2. Blob detectors: It detects local interest regions which differ in properties such as illumination, color, etc. Examples are HessianAffine [Mikolajczyk and Schmid, 2004], DoG [Lowe, 2004].

3. Region detectors: It detects homogeneous image regions based on intensity, segmentation, etc. Examples are MSER [Matas et al., 2002], IBR [Tuytelaars and Van Gool, 2004].

In the following, we discuss several important detectors from the different categories mentioned above.

Harris detector: Different variant of Harris corner detector, such as in [Harris and Stephens, 1988; Schmid and Mohr, 1997], or color Harris detector [Montesinos et al., 1998] enhance the Moravec's corner detector [Moravec, 1977] by bringing new invariant types. The foremost enhancements are:

1. It performs an analytic expansion on the shift origin to involve all possible small shifts.

2. It substitutes a smooth circular window by a binary rectangular window.

3. It develops the corner response metric to take into account the intensity changes with the direction of shifts.

The detected key points are invariant to rotation although it cannot deal with scaling.

Harris-Laplace detector: Harris-Laplace region detector [Mikolajczyk and Schmid, 2004] combines the Harris corner detector and function of Laplace to be invariant to scale changes. This algorithm starts by simulating changes in scale product convolution between the original image and Laplacian of Gaussian (LoG) at different scales. It detects potentially significant points which are invariant to scales with Harris corner detectors.

Hessian-Laplace detector: Hessian-Laplace detector [Mikolajczyk and Schmid, 2004] is a modification of Harris-Laplace detector. It accelerates the speed by calculating Hessian matrix instead of Harris corners. This makes it robust to the viewpoint change. The number of regions can be controlled by thresholding Hessian determinant.

Harris-Affine and Hessian-Affine detector: These two improved detectors [Mikolajczyk and Schmid, 2004] are proposed to make them invariant to affine transformation. These detectors instigate by determining the interest points of Harris-Laplace or Hessian-Laplace detectors. The main steps involved are:

1. The affine shape is estimated by an ellipse with the matrix of second-order moments.

2. The elliptical region is normalized to obtain a circle.

3. The new position is detected and the new scale of the normalized image.

4. Replicate the first step if the eigenvalues of the matrix of second-order moments of the normalized image are different from those of the matrix of the first stage.

Maximally Stable Extremal Region Detector: The maximally stable extremal region (MSER) [Matas et al., 2002] is a feature detector which extracts a number of covariant regions from an image. Two main properties of this detector are:

1. All intensities in each MSER are either lower (dark extremal region) or higher (bright extremal region) than intensities outside its boundary.

2. Each MSER is affine invariant for both geometrical and photometrical transformations.

Difference of Gaussian operator and SIFT detector: Difference of Gaussian (DoG) method [Crowley and Parker, 1984] selects the scale-space extrema in a series of DoG images by convolute an image with DoG functions. The convolution process occurs in different local scales and the detected points are candidate key points. This existing method is extended in scale invariant feature transformation (SIFT) detector [Lowe, 2004] in order to deal with scale invariant feature transformation. Few main steps of this process are:

1. The scale-space extrema detection.
2. The key points localization: This step produces more stable key points by discarding low contrast candidate points.
3. The orientation assignment: Assign dominant orientation to a detected key point.

Speeded up robust features: Speeded up robust features (SURF) [Bay et al., 2008] is a scale and rotation invariant feature detector which is inspired by SIFT. This detector is developed by relying on integral images for image convolutions. It is achieved by using existing Hessian matrix based measure for the detector and then it simplifies these methods.

Binary robust invariant scalable key-points detector: Binary robust invariant scalable keypoints (BRISK) [Leutenegger et al., 2011] detector is proposed for fast image matching tasks without sufficient prior knowledge on the scene and camera pose. The keypoints are detected in octave layers using saliency criteria over the image and scale dimensions. The quadratic function fitting is used to obtain the location and scale of the keypoints. The detected points can be described by BRISK descriptor (see BRISK descriptor in the Local feature descriptors section).

Color symmetry: The color symmetry detector [Reisfeld et al., 1995] and its generalization to the color [Heidemann, 2004] are developed on the concept of focus points on the objects. The focus points must be stable to rotation, noise and change in lighting, distinctive or salient and usable for feature extraction. It uses color symmetry map and color based gradient detection to detect symmetries in the content.

Oriented FAST and Rotated BRIEF detector: The Oriented FAST and Rotated BRIEF (ORB) detector [Rublee et al., 2011] is developed from modified feature accelerated segment test (FAST) [Rosten and Drummond, 2006] detector. The missing component in FAST, *i.e.,* orientation is introduced in ORB keypoints using intensity centroid. The corner's intensity is offset from its center, thus intensity centroid is used to attribute orientation during ORB keypoint detection.

Center surround extremas: Center surround extremas (CenSure) [Agrawal et al., 2008] detector is developed for visual odometry application based on following criteria: stability of the features across viewpoint change and consistent localization of features which are invariant to viewpoint changes. CenSure features are:

1. computed at the extrema of the center-surround filters using original image resolution at scale.
2. approximation to the scale-space Laplacian of Gaussian.

We have presented several local feature detectors which belong to different detector families. These detectors detect different interest points or regions in an image depending on their characteristics as well as on the image contents. Therefore, Finding the appropriate features to define the image content is one of the basic keys for an effective CBIR system. In this context, an example is illustrated in the below Fig. 2.3.

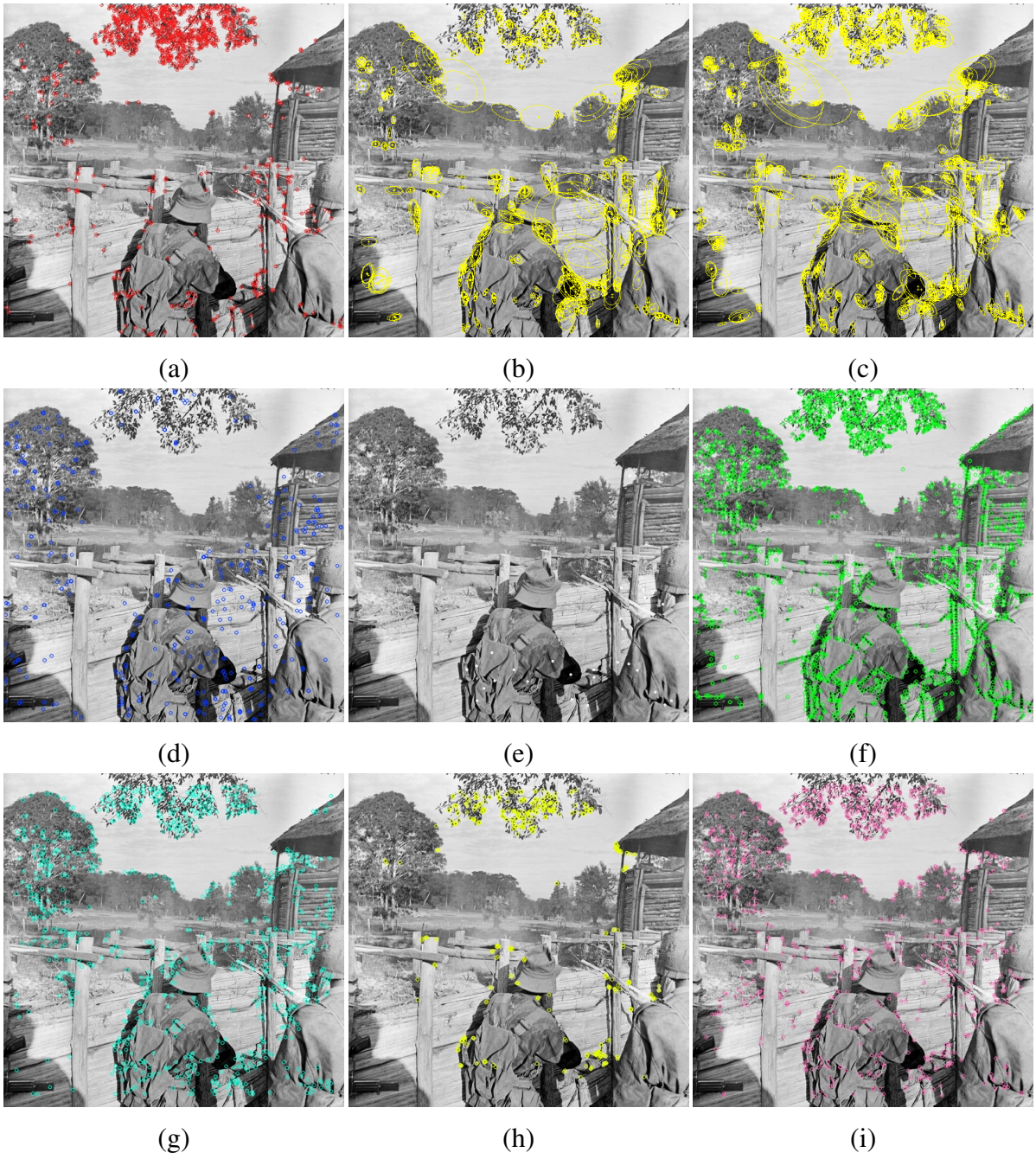


Figure 2.3: Interest points/regions detected by different detectors in an image from Musée Nicéphore Niépce collection: (a) Harris (b) Hesaff Affine (c) Harris Affine (d) MSER (e) Color symmetry (f) SIFT (g) CenSure (h) ORB (i) BRISK.

The observation we can make in the Fig. 2.3 is that the detected interest points or regions by the different genre of detectors are different. All these different detectors involved may or may not have the same level of relevance. The distinctiveness may be different from one content to another. Thus, the combined features of an image by the different complementary detectors may enhance the image representation. Certainly, it is relevant to consider the spatial complementarity evaluations between such local features. A detail discussion on this direction is presented in the Chapter 4.

Local feature descriptors: The local keypoints are described by signatures or description vectors which are referred as a local descriptor. In literature, we can categorize the local descriptors of the content on the basis of the different approaches:

1. *Filter-based approaches:* Gabor Filters [Daugman, 1980].
2. *Distribution-based approaches:* SIFT [Lowe, 2004], GLOH [Mikolajczyk and Schmid, 2005], RIFT [Lazebnik et al., 2005].
3. *Textons approaches:* Texture descriptor [Varma and Zisserman, 2005].
4. *Others approaches:* Color descriptor [van de Weijer and Schmid, 2006].

Another way to classify the local descriptors is: *Geomatric relation* (*e.g.,* Curveture descriptor [Awrangjeb and Lu, 2007]) and *Pixel interest region* (*e.g.,* SIFT [Lowe, 2004], BRISK [Leutenegger et al., 2011]).

All these proposals have the ambition of being more distinctive and invariant or robust to many kinds of geometric and photometric transformations. In the following, we present the details of some notable descriptors of the literature.

Scale invariant feature transformation: Scale Invariant Feature Transformation (SIFT) [Lowe, 2004] is a combination of a detector with a descriptor. The SIFT detector is presented in the local feature detector section. It is one of the most popular feature extraction processes in which image is transformed into scale invariant coordinates. SIFT is one of the best robust local descriptors due to its invariance in scaling, rotation, distortion and translation. Generally, the Euclidean distance metric is used for similarity measurement. The four main steps of SIFT description are:

1. Detection of scale-space extrema
2. Key-point localization
3. Orientation assignment and
4. Local image descriptor, which encapsulates, into a histogram, the distribution of the gradient orientations in the neighborhood of the interest point.

Speeded-up robust features: Speeded-up robust features (SURF) descriptor [Bay et al., 2008] is developed based on SIFT properties. The complexity is reduced in this approach. Two steps of this method are:

1. Orientation assignment and

2. Descriptor components.

In the first step, reproducible orientations have been fixed on the basis of the information from a circular region around the interest point. Then the SURF descriptors are extracted from the constructed square region aligned to the selected orientation. The similarity is measured using the Euclidean or Mahalanobis distance.

Affine SIFT: Affine SIFT (ASIFT) [Morel and Yu, 2009] is an affine invariant feature descriptor which consists of both a detector and a descriptor. While SIFT is fully invariant with respect to scaling, rotation, distortion and translation, this method, ASIFT, deals with remaining two parameters namely latitude and longitude. It simulates all image views obtainable by varying the two camera axis orientation parameters, *i.e.,* the latitude and the longitude angles. Then it also calculates the other four parameters of the SIFT. The Euclidean distance is used to measure similarity.

Color SIFT: Color SIFT [Abdel-Hakim and Farag, 2006] is a color local invariant feature descriptor for the purpose of combining both color and geometrical information in object description. It is the adaptation of the SIFT method to colors images. Hence similarity between the images is measured considering color and shape information. This method builds the SIFT descriptors to the color invariant space. The SIFT is applied on the three color channels *i.e.,* red, green and blue. Therefore the descriptor vector is three times larger than the SIFT vector. Again, for similarity measurement, the Euclidean distance is used.

Histogram of oriented gradient: Histogram of oriented gradient (HOG) [Dalal and Triggs, 2005] is a feature descriptor commonly used in people detection and other object detection purpose. It avoids hard decisions compared to the edge-based features. This window based descriptor is developed on the occurrences of the gradient orientation in the detected patches (or localized portions) of an image. Histogram or Bhattacharya distances are used for similarity measurement. The basic principle behind HOG descriptor is that the shape and appearances can be well characterized by the local intensity gradients distributions. It is not necessary to know the corresponding positions of gradients. This can be implemented by dividing an image into several spatial regions which are known as a cell. Then one-dimensional gradient direction histogram (edge orientations) is collected over the pixel of each cell. The collective histograms generate the final representation. The important five steps of HOG algorithm are:

1. Gradient computation
2. Orientation binning
3. Descriptor blocks and

4. Block normalization.

Local binary pattern: Local binary pattern (LBP) [Ojala et al., 1996] is one of the popular texture features operators due to its computational simplicity. The histogram of the binary patterns is calculated over a region and it is generally used for texture description. It describes each pixel by the relative grey levels of its neighboring pixels. For each neighboring pixel, the result will be set to one if its value is no less than the value of the central pixel, otherwise, it will be set to zero. The Mahalanobis distance is used for similarity measurement. Two main captivating properties of LBP are:

1. It is robust to monotonic gray-scale changes due to illumination disparities.

2. Its discriminative power for defining texture structure.

An effective dimensionality reduction method for LBP, named as orthogonal combination of local binary patterns (OC-LBP), is proposed in the work of [Zhu et al., 2013].

Shape contexts: Shape contexts (SC) [Belongie et al., 2002] descriptor is proposed to measure the similarity between shape and also used for object recognition task. In this descriptor, a shape is represented by a discrete set of points. It considers a vector originates from a reference point to all other sample points on a shape. The SC descriptor is invariant to scaling, translation and small geometrical distortions and occlusion. Shape distance, which is the weighted sum of three terms, *i.e.,* shape context distance, image appearance distance and bending energy, is used as distance metric and used in different applications.

Fast retina key-point descriptor: Fast retina keypoint (FREAK) [Alahi et al., 2012] is a key-point descriptor which is proposed recently inspired by human visual system or retina. The main steps involved to generate this binary descriptor are:

1. Retinal sampling pattern is generated using retinal sampling grid, which is circular in pattern with higher density of the points near the centre (similar to the spatial distribution of ganglion cells in eye ratina).

2. A coarse to fine descriptor is formed by a sequence of a one-bit difference of Gaussians (DoG).

3. Saccadic search is used to select relevant features and the orientation of the keypoints are estimated by summing the local gradients over selected points. It uses Hamming distance.

Oriented FAST and Rotated BRIEF descriptor: ORB descriptor [Rublee et al., 2011] is a binary descriptor which used ORB keypoints for feature description. It uses the modified version of binary robust independent elementary features (BRIEF) descriptor [Calonder et al., 2010]. The ORB descriptor provides a learning method for de-correlating BRIEF features which are invariant to rotation. Hamming distance is used as distance metric.

Binary robust invariant scalable keypoints descriptor: Binary robust invariant scalable keypoints (BRISK) is a binary descriptor [Leutenegger et al., 2011] which is proposed for fast image matching. The BRISK keypoint detection is explained in the local feature detector section. The sampling pattern of the oriented BRISK is applied at the neighborhood of each keypoint. It processes the local gradient intensity and the direction the feature characteristic is determined. Finally, the pairwise brightness comparison is incorporated in the description using sampling pattern. For descriptor matching, Hamming distance is used.

As presented in this section, the literature on local image descriptors is very rich. This provides several families to describe different image characteristics for different targets. For example, local binary pattern descriptor is best suitable for the images with texture contents. On the other hand, in general, SIFT performs better with the images which contain objects or shapes. Thus, the combining local image descriptors to obtain a descriptor of higher semantic is propitious to better represent image contents. Hence, descriptors fusion could be useful for CBIR.

So far, these hand-crafted features enjoyed remarkable success in various vision-based tasks. Let us trun our focus on the learning-based features. The learning-based features or deep features have grabed research communities attention in a very short period of time. In the following section, these features are presented.

2.2.2 Deep learning features

In the last decade, the conventional hand-crafted features are challenged by the new domain of machine learning, deep learning, which has a positive impact on the wide range of computer vision applications, such as classification, detection, recognition, retrieval, localization, etc. The deep learning features are learned automatically from the dataset using Convolution Neural Network (CNN). Therefore, these features are called as CNN features or deep features. The idea behind the deep features is originated from the neural network working concept. It is a representation learning method with multiple layer perceptron which consists of many hidden layers of abstraction. The features can be obtained by composing non-linear modules at one layer to represent the higher abstract level. In this fashion of composition, complex concepts can be learned and represented. In this context, Backpropagation algorithm, which was invented in the 1980s, is one of the useful ways to configure the module parameter et each layer. However, Backpropagation is not always efficient as it can get trapped in poor local optima. Therefore, the performance is deteriorated with the increasing number of layers in a network.

The introduction of the convolution neural network (ConvNet) [Krizhevsky et al., 2012] made a huge impact on the deep learning process, especially when the research in the direction of deep learning or neural network was fallen out of favor. The ConvNet architecture is illustrated in the Fig. 2.4.

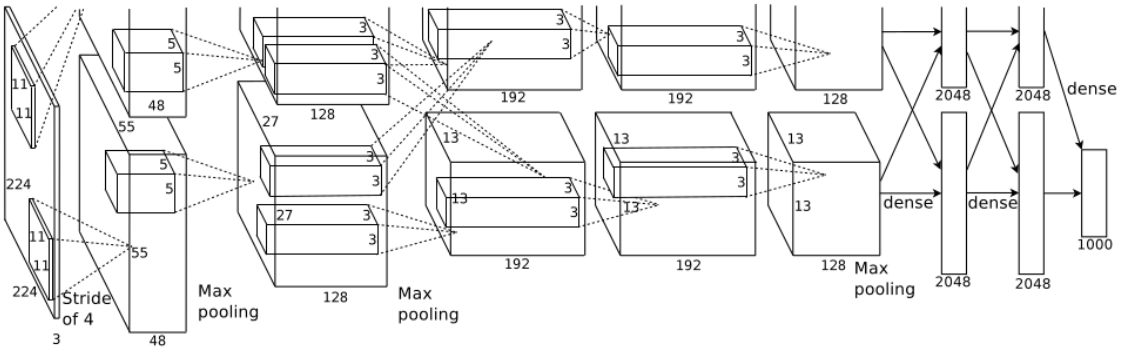


Figure 2.4: A ConvNet architecture as proposed in [Krizhevsky et al., 2012], where it is showing the delineation of responsibilities between the two graphics processing units. The features can be extracted from the different layers.

The ConvNet relies on the primary four properties of signals, *i.e.,* use of multiple layers, local connections, shared weights and pooling. Since then, ConvNet is widely used to address a large computer vision problems, such as classification, face recognition, segmentation, image retrieval, natural language understanding, speech recognition, etc. Although ConvNet was quite successful to solve computer vision tasks, it took another 14 years to gain attention from the research communities. The deep convolutional neural network [Krizhevsky et al., 2012] is proposed in 2012 for image classification where it enjoyed spectacular accuracy. The network was trained with 1.2 million images of ImageNet LSVRC-2010[1], consists of 1000 classes. The network has five convolution layers, 60 million parameters and 650000 neurones. Since then, all the big players, such as Google, Facebook, Microsoft, Twitter, etc., as well as growing number of start-ups in the computer vision industry are focusing heavily on the deep learning research. Not only that, the hardware giants, such as Intel, NVIDIA, Qualcomm, Samsung, etc., are developing ConvNet chips to solve several vision-based problems. Another notable work, called deep belief networks (DBNs), proposed in work of [Hinton et al., 2006], which is an efficient and unsupervised learning algorithm. It is greedy multiple layers learning processes which optimize network parameters at each layer. The DBNs learning algorithm is quite effective for unlabeled data. Additionally, the overfitting and underfitting problems of deep networks are also addressed in this proposal.

In general, CNNs are trained with larger image collections of diverse contents such as, ImageNet [Russakovsky et al., 2015], Caltech101 [Fei-Fei et al., 2007], etc. CNN learns rich deep features from these diverse images. With the success of ConvNet, several learning strategies are proposed in the literature [Donahue et al., 2013; Girshick et al., 2014; Simonyan and Zisserman,

[1]http://www.image-net.org/challenges/LSVRC/2012/

2014; Liu et al., 2015]. Not only that, several open source deep learning libraries are also developed for research purpose. Few of these libraries are TensorFlow [2], Caffe[3], Theano[4], Torch[5], Keras[6], mxnet[7], DIGIT[8], deepy[9], Deeplearning4j[10]. Using the convolution network, three vision tasks, such as image classification, localization and detection, are addressed in the work of [Sermanet et al., 2013]. A fast and accurate deep feature extractor, named OverFeat, is proposed in this work. A multiscale and sliding window strategy is implemented within ConvNet, where it is trained with 1.2 million ImageNet-2012 images. In the work of [Donahue et al., 2013], several vision tasks, such as scene recognition, domain adaptation, and fine-grained recognition, are tackled using deep convolution features. The extracted deep features are applied in a semi-supervised multi-task framework where an auxiliary large labelled object database is used to train a deep convolutional architecture. Object detection and semantic segmentation problem are tackled in the work of [Girshick et al., 2014] by using deep learning method. In this work [Girshick et al., 2014], the multiple low-level features are combined with high-level conceptual CNNs. The features are learned from ImageNet-2012 dataset with image-level annotation by Caffe library. Another ConvNet based approach [Oquab et al., 2014] is proposed for image classification. The proposal in this work is to use of internal layers of the CNN to extract mid-level image representations. The network is trained on ImageNet and produces a improvements accuracy on the Pascal VOC [Everingham et al., 2010] for object and action classification tasks as well as object and action recognition task. CNN is also used to solve different vision-based medical applications, such in the work of [Chaabouni et al., 2016; de San Roman et al., 2017].

Certainly, there is a boom in using deep learning techniques to solve image classification, detection, recognition problems. Thus, deep learning could be useful for content-based image retrieval problem. The primary emphasis in CBIR is to describe the suitable image characteristics, which should coincide with the user's vision and perception of similarity of the images, *i.e.,* to reduce the gap between low and high-level semantic concepts. In deep learning, the high-level concepts are modelled by employing multi-layer convolution neural networks. A general schema for CBIR using deep learning is illustrated in the Fig. 2.5 [Wan et al., 2014].

In the work of [Wan et al., 2014], a deep learning framework for CBIR is proposed. The approach consists of two main steps, *i.e.,* train large-scale deep learning model and uses the learned model to represent the features for image retrieval. The training dataset is used ImageNet-2012 which consists of 1000 classes of images. Similarly, inspired by the ConvNet, [Liu et al., 2015] proposed an image retrieval strategy which uses deep features into classical inverted indexing

[2]https://www.tensorflow.org/
[3]http://caffe.berkeleyvision.org/
[4]http://deeplearning.net/software/theano/
[5]http://torch.ch/
[6]https://keras.io/
[7]http://mxnet.io/
[8]https://developer.nvidia.com/digits
[9]http://deepy.readthedocs.io/en/latest/
[10]https://deeplearning4j.org/

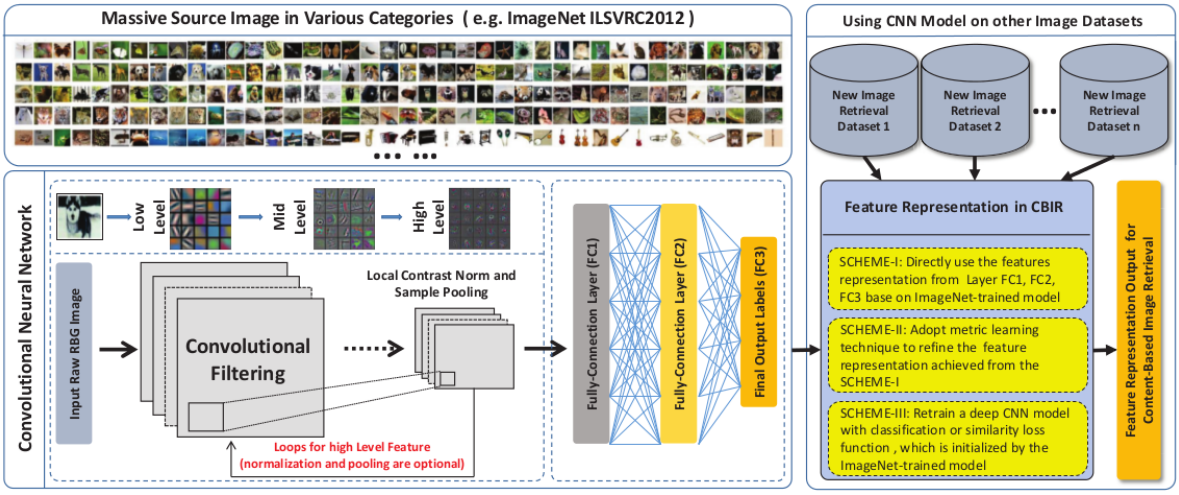


Figure 2.5: A framework of deep learning with application to CBIR [Wan et al., 2014].

structure. Additionally, a DeepIndex framework is proposed which incorporate multiple deep features from different layers. [Ng et al., 2015] exploits the local features which are learned from deep networks for image retrieval application. The model is trained with OxfordNet [Simonyan and Zisserman, 2014] and GoogLeNet [Szegedy et al., 2015] deep networks. The features are extracted from the different layers of the network and VLAD [Arandjelovic and Zisserman, 2013] encoding scheme is adopted to represent the features into a single vector. Several other deep learning based approaches [Alzu'bi et al., 2016; Zhou et al., 2016; Gao et al., 2015; Yang et al., 2016a] are also proposed for image retrieval.

As the popularity of deep learning is upsurging, performance comparisons between deep and conventional hand-crafted features are reported in the work of [Fischer et al., 2014; Liu et al., 2016a; Yan et al., 2016; Schönberger et al., 2017] for different applications. In the work of [Liu et al., 2016a], several conventional features are compared with CNN features for image retrieval application and the favorable outcome reported for hybrid keypoint detector and CNN descriptors. SIFT [Lowe, 2004] features and ImageNet trained deep features are compared for matching task in the work of [Fischer et al., 2014] and the deep features outperform SIFT features. Although, deep features perform quite efficiently for different computer vision tasks, but the computational cost is significantly higher compared to conventional features due to the involvement of several layers of learning stages. Therefore, it opens another important research direction, *i.e.,* are these approaches implementable using low processing power or low-powered embedded system? Few embedded architectures, such as in CEVA-XM6[11], [Yang et al., 2016b; Chen et al., 2017], are proposed to implement deep convolution neural networks computer vision algorithms. However, the available solutions in this direction are very limited and not well explored. Therefore, in-depth research needs to be carried out in this direction.

[11] http://www.ceva-dsp.com/CEVA-XM6

2.3 Image representation models

We present the details of the feature extractions in the Sec. 2.2. In CBIR, these features are compared in order to return the images involving features similar to the one(s) of the query, with or without using an indexing structure. The comparison step usually comprises exhaustive search process, which may not be productive for a large volume of features. Thus, the attention is given to developing efficient models, which could expedite the search process. Several image representation models [Sivic and Zisserman, 2003; O'Hara and Draper, 2011; Nister and Stewenius, 2006; Philbin et al., 2007; Douze et al., 2009; Vieux et al., 2012], in which features are represented compactly with efficient indexing structures, are proposed in the literature. Bag of features (BoF) model stands out among them and it is widely adopted for several computer vision applications. In this section, we primarily focus on the BoF and its related models.

Bag of features: One of the effective representation models is Bag of Features (BoF) or bag of visual words. The extracted features are encapsulated in a BoF model which is inspired by the Bag of Words (BoW) concept [Sivic and Zisserman, 2003].

BoW model was first mentioned in one of the articles of the distributional structure by Zillig [Harris, 1954]. In BoW model, a text (such as a word or sentence) is assumed to be an unordered collection of word(s), irrespective of the order of the words or grammars. This model is mainly used in documents classification, indexing, spam filtering, topic modelling etc. Similarly, in BoF, visual features are clustered into a vocabulary of visual words [Sivic and Zisserman, 2003]. The main four steps of BoF are:

1. Feature extraction: This step is explained in the Sec. 2.2.

2. Feature quantization and codebook generation: An unsupervised learning process, such as k-nearest neighbors, Hierarchical k-means, etc., is used to quantize each feature in order to get visual words. The codebook is a representation of the all extracted image features. It is a clustering process of the features which are extracted from the image patches. Then these clusters are used as an entry of a unified codebook [Jurie and Triggs, 2005; Csurka et al., 2004]. Each cluster relates to a sub-space in the feature space. The centroid of the cluster is treated as a visual word which represents the closest descriptors. For a given feature point, a visual word id which is closest to visual words has been assigned by feature quantization.

3. Image Representation: After quantization, an image can be viewed as a frequency histogram of bag of visual words. The similarity measurement between the query and each database image is calculated by comparing visual word histograms, which is also known as image vectors. A histogram intersection which is defined by below formula may be

used for similarity measurement:

$$d(Im1, Im2) = 1 - \sum_{i=1}^{n}(min(P(Im1), P(Im2))) \quad (1)$$

where $P(Im1)$ and $P(Im2)$ are the probabilities of the visual words of $Im1$ and $Im2$. BoF based image retrieval models typically use Term Frequency-Inverse Document Frequency ($tf - idf$) weights [Salton and McGill, 1986] which penalize too common terms and emphasize on unique terms. If the vocabulary consists of k words, then each document in the database can be represented by the $tf - idf$, which is computed as shown in the Eq. 2.

$$tf - idf = \frac{n_{id}}{n_d} log \frac{N}{n_i} \quad (2)$$

where n_{id} is the number of occurances of words i in document d, n_i is the number of occurances of i in the entire database, and the N is the total number of documents.

BoF model has emanated as one of the popular visual representation due to compactness or abridged storage requirements and swiftness of search process. However, BoF model faces scalability issue while dealing with the high volume of features. A large number of features leads to a very large vocabulary size, which may not be desirable for many computer vision tasks. Also, BoF model fails to describe the of the spatial relationship between the features. It ignores the semantics of the features.

BossaNova model of BoF: BossaNova [Avila et al., 2013], based on BoF, is a unique representation of images or videos for several computer vision applications. BossaNova is the mid-level representation which is developed on the histogram distance between the descriptors in the images and the codebook. It assimilates different enhancements on BOSSA model [Avila et al., 2011]. It relies on the discriminative local descriptors generated by codebook and aggregation of the quantized descriptors by enhanced pooling strategy. Important steps of the BossaNova model are:

1. New pooling scheme: Pooling step is important as it coagulates information which is extracted by the descriptors, into a feature vector. This produces a mid-level feature and it is expedient for use with classifiers such as support vector machine (SVM). One of the problems of pooling scheme is it introduces vagueness in the codebook. The final codebook is activated by combining all the features (such as invariance to the different background, the position of the object etc.). This combination introduces the ambiguity in the codebook and the different set of codewords overlap excessively. A nonparametric estimation of the descriptor distribution has been proposed to address this problem. It is achieved by preserving more information about descriptor during pooling step. Histogram of distances is computed between the codebook elements and the descriptors in the image.

2. Weighting BoW and BOSSA: During the pooling process, a local histogram is generated for each codeword. The feature vector is computed by concatenating all the histograms. An additional component is introduced for each codeword to count the local descriptors which are falling close to that codeword. A weight factor is proposed to weight each histogram. The model is improved amendment of BoF representation by consolidates more informative pooling function.

3. Localized soft-assignment coding: One of the simplest coding strategies is hard assignment, *i.e.,* to assign a local feature to the closest codeword. However, it often introduces large quantization error. To overcome this drawback, soft assignment coding strategy is adopted. In soft assignment, a local feature is assigned to all visual words. The coding coefficient of the feature is different for each codeword. In this work [Avila et al., 2013], only k codewords of a local feature are considered during the soft assignment and the distance to the remaining codewords are set as infinity. Two main advantages of soft-assignment are, first, it cripples the coding error effect which is introduced by descriptor quantization and second, it produces better result compared to hard assignment without compromising computational efficiency.

4. Normalization: When the vector signatures become too sparse with the increasing number of visual words, a normalization operator is applied on each histogram followed by applying l_2 normalization.

Discriminative codebook learning: With the increasing size of the image database, a vocabulary tree method with hierarchical k-means [Nister and Stewenius, 2006] is more preferred for clustering. It is also very proficient for local feature quantization and is easy to implement. However, the drawback of the unsupervised approach is not to embed the labelling information of training images. Hence semantic contexts are absent and it has a less discriminative ability. To mitigate this problem learning based codebook construction methods have been proposed such as in [Jurie and Triggs, 2005; Moosmann et al., 2008; Perronnin et al., 2006; Lazebnik and Raginsky, 2009]: adaption of the codebook based on the semantic labels, building class-specific codebook, semantic vocabulary construction, etc.

Most of the approaches focus on developing a new codebook based on raw local features. A supervised codebook learning method [Tian and Lu, 2013] is proposed to capture discriminative information for web image search. In this approach, the subspace learning is introduced in codebook construction. The discriminative codebook construction and contextual subspace learning can be learned simultaneously. The main steps are as follows:

1. Feature extraction, such as SIFT, from the training images.

2. Unsupervised codebook generation using the K-means clustering in the feature space.

3. Building discriminative codebook from the training data by using contextual subspace as projection matrix.

Each image is represented as a BoF histogram by quantizing each feature in the new space into the nearest visual word in the discriminative codebook.

Fusion of BoF and fisher linear discriminative analysis: The problem of the semantic gap between low-level visual features and high-level semantic exists in many computer vision applications such as image retrieval, classification, etc. BoF model is used to address the semantic gap problem in the literature [Bosch et al., 2006; Winder and Brown, 2007; Yang et al., 2009]. [Winder and Brown, 2007] proposes hard vector quantization and then pooling for coded features to solve this issue. [Yang et al., 2009] developed a spatial pyramid matching kernel based approach which generalized hard vector quantization to sparse coding using multi-scale spatial max pooling. [Bosch et al., 2006] proposed a strategy to reduce the semantic gap by introducing mid-level by applying probabilistic latent semantic analysis (pLSA) on BoF model. However, in the work of [Bosch et al., 2006], the number of model parameters increases with the growing size of the dataset. To address these issues a fusion algorithm of BoF and fisher linear discriminative analysis (FLDA) [Fukunaga, 2013] is proposed by [Yang and Zhao, 2012]. In this approach, BoF is used as the initial semantic description of images. The FLDA algorithm is applied to get its distribution in a subspace to overcome the shortcoming of pLSA model. Finally, the images are classified by k-nearest neighbor algorithm.

2.4 Feature combination in content-based image retrieval

Fusion of multiple descriptors in image retrieval is gaining popularity in recent years. Image retrieval could be text-based using meta data or it could be content-based by only analyzing the content or the combination of both text and content such in hybrid retrieval. Since neither of text nor content, offers a satisfactory bridge to solve the infamous semantic gap, more and more search engines employ hybrid text and visual representations in order to describe and search multimedia databases. Combined hybrid search yields many times results that are more meaningful compared to the situation when only one modality is used (text or image).

For example, in the work of [Ferecatu et al., 2008], the semantic gap problem is addressed by proposing SVM-based active relevance feedback framework, where feature vectors are generated based on keywords associated with an image. Another approach [Zhou and Huang, 2002] proposed to use a seamless joint querying and relevance feedback framework where the low-level image features are incorporated with keywords. Another strategy of CBIR based on hybrid image information is proposed in [Bassil, 2012]. The visual content is represented by color features and histogram and the textual information is extracted from terms that present in an HTML

document where the image is found. Several other hybrid strategies can also propose in the literature [Dinakaran et al., 2010; Luo et al., 2003; Zhang et al., 2013]. Major search engines, such as Google, offer image search by combing text and content. For more information, we list down available CBIR search tools in the Appendix A.

Coming to the content-based retrieval, we already presented in Sec. 2.2, that several families of descriptors exist and in each family, a lot of approaches live together. Many descriptors do not describe the same information and do not have the same properties. Therefore it is relevant to combine some of them to better describe the image content.

Fusion can be investigated differently according to the involved descriptors, the strategy of combination and the application targeted. Several techniques for the combination of image descriptors have been proposed in the literature of CBIR [Madhusudhanarao et al., 2015; Ji et al., 2013; Neshov, 2013; Dubey et al., 2010; Cao et al., 2010; Bouteldja et al., 2008]. In general, the fusion steps are performed in a different position in the entire process of retrieval. Thus, the fusion strategies can be broadly categorized into two main types:

1. Early fusion and

2. Late fusion

Although the most of the fusion strategies fall into these two categories, we introduce a third category, called *i.e.,* 'Other fusion', which groups remaining fusion strategies, such as intermediate, sequential fusions, etc. In this section, we present many examples from the different fusion categories. In the Secs. 2.4.1, 2.4.2 and 2.4.3, several fusion strategies are presented in details. Before that, the basic concept of the early and late fusion is discussed below.

Early fusion: Early fusion usually refers to the combination of the features into a single representation before comparison or learning or retrieval. After extracting the features from images using multiple descriptors, the features are combined into a single depiction. After combination, the image retrieval and similarity measurement related steps are performed. The early fusion scheme is shown in the Fig. 2.6.

Thus, early fusion integrates the multiple features into a single depiction before applying retrieval, classification steps. Several strategies have been proposed for early fusion in the literature, such as feature concatenation [Yu et al., 2013], weight based early fusion [Yue et al., 2011], etc. Since the features are combined at the beginning of the process, the retrieval steps need to execute only once. On the downside, the spatial representation of the different feature vectors are not same, hence it is challenging to combine the features into a single representation. Another added disadvantage is that the combined representation of the feature generates high dimensional vectors, which may not be desirable during the retrieval process.

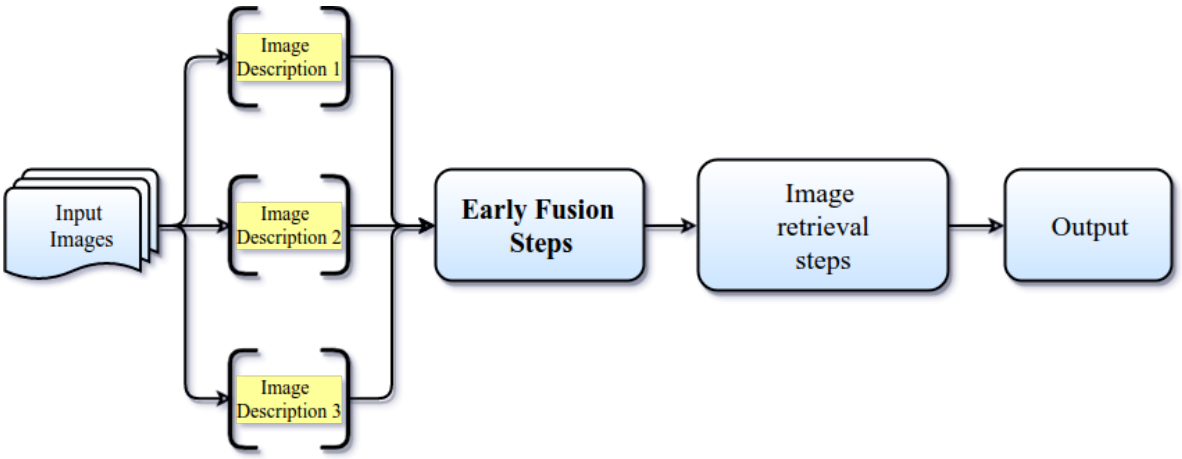


Figure 2.6: General schema of early fusion strategy.

Late fusion: Late fusion refers to the combination, at the last stage, of the responses obtained after individual features comparison or learning. When considering image retrieval, multiple ranked outputs of the multiple descriptors are aggregated to generate another concluding ranked output. This method of fusion can be implemented either score-based [Neshov, 2013] where it combines the different similarities or distances from the query or ranked-based [Neshov, 2013] which considers the combination of the response ranks. The outputs to combine can be weighted to give more importance to particular descriptors, by fixing the weights a priori or, better, by learning them for a given content [Huang et al., 2015b]. While considering image classification, the multiple classifiers are performed on each set of descriptors. The outcomes of the classifiers are combined afterwards to produce a final decision. Hence, late fusion put importance on the discrete strength of each modality. The late fusion scheme is shown in the Fig. 2.7.

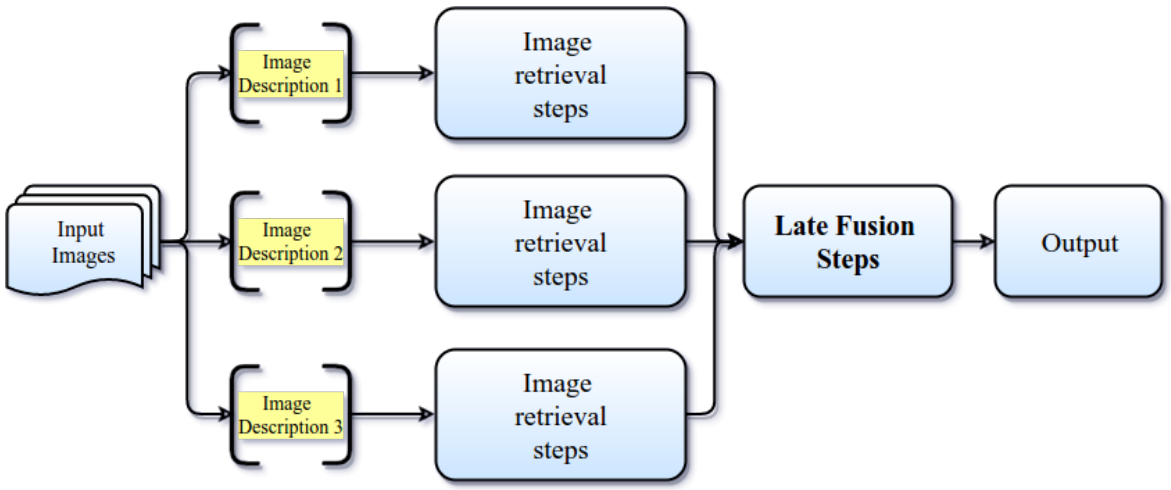


Figure 2.7: General schema of late fusion strategy.

Thus, late fusion integrates the retrieval outputs, which are generated for each individual feature, at the end of the process. In late fusion, the multiple retrieval or comparison steps need to be performed multiple times. This computationally expensive compared to early fusion strategy

where the retrieval step is executed only once. In several applications, late fusion scheme could produce superior result compare to early fusion, but at the cost of increased computational efficiency.

2.4.1 Early fusion strategies

There are several early fusion strategies that have been proposed in the literature of CBIR. In the following, we present the details of some notable early fusion strategies.

Feature integration by concatenation in image retrieval: One of the most widespread solutions in early fusion is to concatenate the feature vectors into a single vector, such as in [Yu et al., 2013] with SIFT [Lowe, 2004], HOG [Dalal and Triggs, 2005] and LBP [Ojala et al., 2002] features. In the work of [Yu et al., 2013], the features are described by SIFT, LBP and HOG descriptors. The SIFT-LBP and SIFT-HOG features are concatenated respectively to generate two high-dimensional local semantic descriptors. The BoF histogram model is applied for image retrieval. The schematic diagram of this proposal is illustrated in the Fig. 2.8.

Following steps are performed during the image retrieval:

1. 128-dimensional SIFT descriptor ($SIFT_i$) is used to describe each keypoint.

   $$SIFT_i = [SIFT^1 \quad SIFT^2 ... SIFT^{128}]$$

2. Similarly, LBP descriptor computes 64-dimensional description vectors.

   $$LBP_i = [LBP^1 \quad LBP^2 ... LBP^{64}]$$

3. SIFT and LBP description vectors are integrated by concatenating LBP description at the end of SIFT descriptions. This generated 192 ($= 128 + 64$) dimensional vector.

   $$SIFT - LBP_i = [SIFT_i \quad LBP_i] \tag{3}$$

4. The codebook is generated using k-means clustering on the integrated description and BoF histogram model is applied for image retrieval.

Features concatenation in CBIR: In the work of [Choudhary et al., 2014], color moment [Yu et al., 2002] and LBP [Ojala et al., 2002] features are concatenated for CBIR. The basic idea is to extract color moment and LBP features from the images and integrate them in a single description before applying an exhaustive search process between the query database image features. The performed steps are described below:

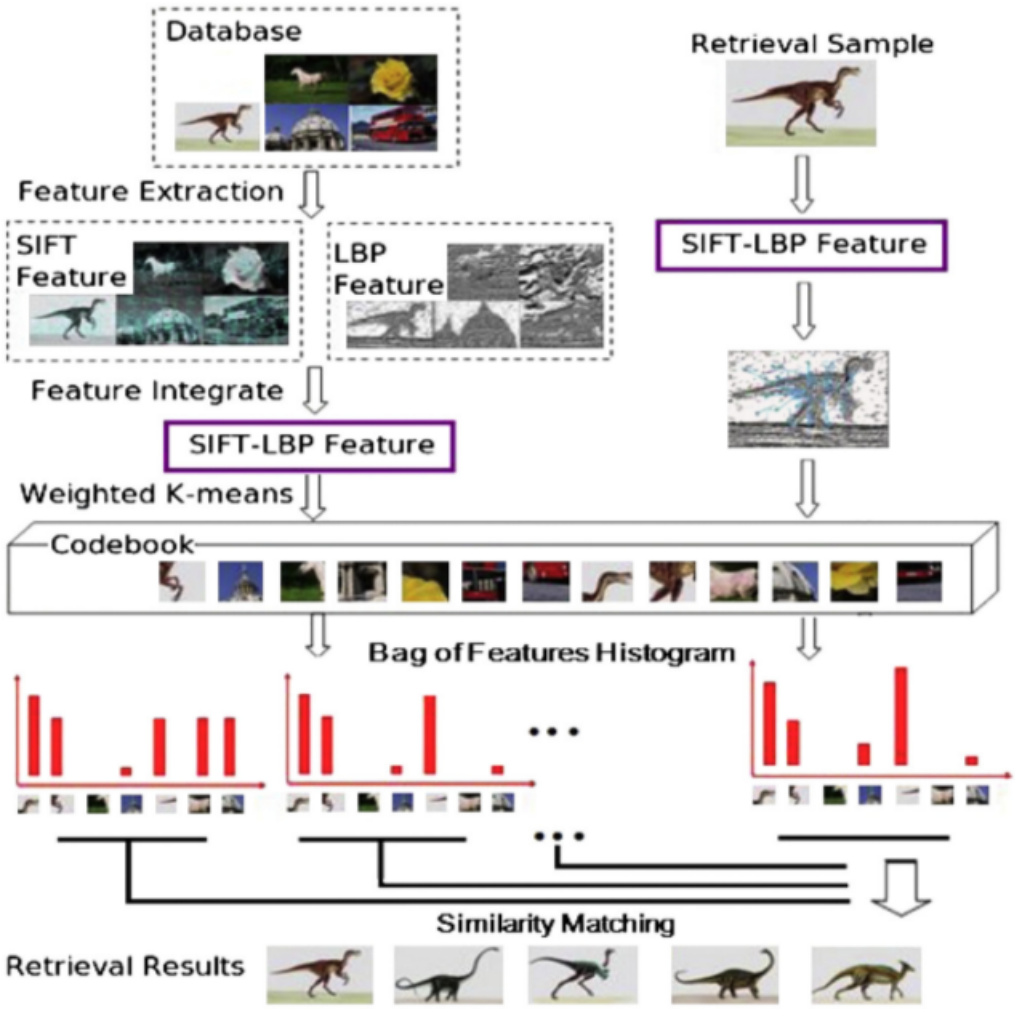


Figure 2.8: The schema for image retrieval based on the BoF model using the SIFT-LBP feature integration proposed in [Yu et al., 2013].

1. Preprocessing steps: Several preprocessing steps are performed on both the database and the query image. Preprocessing steps include image enhancement, segmentation, etc. to adjust the image contrast, segmentation based on color.

2. Feature extraction: Two types of local features, such as color moment and LBP features are extracted from the images. The LBP [Ojala et al., 2002] texture features are extracted from (3 *by* 3) pixel image patches. The LBP feature generation is presented in the Sec. 2.2.1.2. Color moments [Yu et al., 2002] features are computed by measuring the color distribution in an image. The most of the color distribution information is contained in the lower order moments. Hence, three color moments, *i.e.,* mean, standard deviation and skewness are computed. The first order moment, mean, is interpreted by averaging the color in an image. The standard deviation can be computed by taking the square root of the variance of the color distribution. Skewness determines the shape of the color distribution by measuring asymmetricity in the distribution. The computed color moment features are invariant to scaling and rotation.

3. Feature concatenation: In this step, the final feature vector is generated by concatenating color moment and LBP features.

4. Image retrieval: To measure the similarity to a query, the exhaustive search process is applied between each database features and the query features.

Although the complexity of the proposed fusion strategy is quite simple, this strategy is not suitable for the image retrieval where a larger volume of features needs to be manipulated.

CBIR using color and texture fused features: In the work of early fusion approach proposed in [Yue et al., 2011], color and texture features are combined for CBIR. These features are extracted from the query and the each image in the database. Then the features are combined based on the weight given for each feature before using them for similarity measurement. We present the steps involved in this approach in the following.

1. Extracting color histogram: Color features, which are invariant to rotation, translation and scale, are commonly used for image retrieval applications. To calculate the color histograms, the HSV color space is divided into several small ranges. The color space is quantified using the Eq. 4.

$$H = \begin{cases} 0 & h \in [316, 360] \\ 1 & h \in [1, 25] \\ 2 & h \in [26, 40] \\ 3 & h \in [41, 120] \\ 4 & h \in [121, 190] \\ 5 & h \in [191, 270] \\ 6 & h \in [271, 295] \\ 7 & h \in [295, 315] \end{cases} \quad S = \begin{cases} 0 & s \in [0, 0.2] \\ 1 & s \in [0.2, 0.7] \\ 2 & s \in [0.7, 1] \end{cases} \quad V = \begin{cases} 0 & v \in [0, 0.2] \\ 1 & v \in [0.2, 0.7] \\ 2 & v \in [0.7, 1] \end{cases} \tag{4}$$

   Each feature value is counted to generate the histogram. Although it is quite simple to generate, the spatial distribution of the color is lost in this process.

2. Extracting texture features: To compute the texture features, the color images are converted to a grey-scale image as shown in the Eq. 5.

$$Y = 0.29 * R + 0.587 * G + 0.114 * B \tag{5}$$

   Where Y is grey-scale value and R, G, and B are red, green and blue components. The grey-scale images are quantified before computing the four texture parameters, capacity, entropy, moment of inertia and relevance using co-occurrence matrices, which are calculated in four directions, 0°, 45°, 90° and 135°. Texture feature components are generated

by taking means and standard deviations of each parameter. The Gaussian normalization is applied to make each feature with the same weight.

3. Fusion of color and texture features: These two features are combined based on the weight assigned to each feature. In order to find the optimal weight values, the weights are varied in the range of (0,1). This implies that the weight value for one feature is varied from 0 to 1 and the weight value for the second feature is varied from 1 to 0. It is shown that the equal weight values for each feature give best retrieval accuracy.

4. Once the features are combined based on their respective weight values, Euclidean distances are calculated between the query features and database features. The similar images are retrieved depending on the calculated distances.

In this proposal, the exhaustive search process involved and hence, it is not suitable for the image retrieval at a large scale.

Genetic programing framework for descriptor combination in CBIR: In the work of [da S. Torres et al., 2009], several shape descriptors [Arica and Vural, 2003; da S. Torres et al., 2004] are combined to create a composite descriptor for CBIR. The Genetic Programing (GP) [Koza, 1992] is used to find a suitable combination function for the descriptors. The concept of GP in artificial intelligence is inspired by the principles of biological inheritance and evolution. In the context of CBIR, no prior work has been done using GP. It consists of several key components. These components are:

- Terminal: These are the leaf nodes (x, y as shown in the Fig. 2.9) in a tree structure.

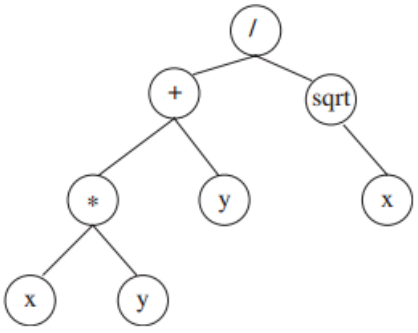


Figure 2.9: Sample tree representation [da S. Torres et al., 2009].

- Functions: Numerical operators, such as $+$, $-$, $*$, $/$, etc. use to combine leaf nodes.

- Fitness function: These are functions which need to be optimized using GP. It is very important to obtain the best descriptor combination. Different fitness functions [Fan et al., 2004], such as FFP1, FFP2, FFP3, FFP4, CHK and LGM, are used in this proposal [da S. Torres et al., 2009].

- Operators: Three operators, such as reproduction, mutation, and crossover, are used to modify the population of combination functions.

The GP is used to combine the similarity obtained from individual descriptors and to create a combined similarity function for the composite descriptor. In this proposal, two approaches are put forward. One approach considers only training set and another one considers training and validation set while finding appropriate combination function. The use of validation sets avoids the overfitting problem which occurs if the model parameters fit training data overly well. These two approaches are almost identical except the extra steps are performed for the validation set. Therefore, the GP framework with validation set is presented below.

Algorithm 1 – GP FRAMEWORK WITH VALIDATION SET

1. *Let T be a training set*
2. *Let V be a validation set*
3. *Let S be a set of pairs (i, $fitness_i$), where i and $fitness_i$ are an individual and its fitness, respectively.*
4. $S \leftarrow \varnothing$
5. $P \leftarrow$ *Initial random population of individuals ('similarity trees')*
6. *For each generation g of Ng generations do*
7. *For each individual $i \in P$ do*
8. $fitness_i \leftarrow fitness(i, T)$
9. *Record the top N_{top} similarity trees and their fitness values in S_g*
10. $S \leftarrow S \cup S_g$
11. *Create a new population P by*
12. *Reproduction*
13. *Crossover*
14. *Mutation*
15. $F \leftarrow \varnothing$
16. *For each individual $i \in S$ do*
17. $F \leftarrow F \cup (i, fitness(i, V))$
18. *BestIndividual* $\leftarrow$ *SelectionMethod*(F, S)
19. *Apply the 'best individual' on a test set of (query) images*

It is an iterative process. At line 5, the population starts with randomly created individuals/descriptors. It evolves using GP operations. The fitness function is used to select the best descriptor combination. At line 11, GP operators are applied to create better-performing descriptor combination. combinations. Finally, the best performing descriptor combination is selected by averaging the performance in both training and validations sets and this combination is used for the test sets.

Bag of features using mutliple feature combination: BoF model is used for several object/scene recognition related tasks [Wu et al., 2009; Botterill et al., 2008; Aldavert et al., 2009]. In the work of [Cho et al., 2011], BoF signature model is used for object retrieval. The features are described using invariant region descriptors before using them in BoF model. Following steps are performed in this approach [Cho et al., 2011].

1. Set of region descriptors: The features are extracted by dense sampling method where regions are selected randomly regardless of image distribution. Three-level of pyramid image and a half overlapping regions of size $48by48$ is used in order to extract the variety of information and invariant features. The features from the set of regions are described by the region descriptor using radial partitioning and multi-level features [Cho et al., 2010]. Each region is divided in 8-level radial and 8-level angular partitioning. The extracted multi-level average intensity descriptions are used for BoF signature generation.

2. Vocabulary tree and BoF Signature extraction: The extracted descriptions are quantized using hierarchical k-means clustering [Nister and Stewenius, 2006] and visual words are generated. The BoF signatures are constructed using binary index vector which represented by the appearance of visual words. The illustration is depicted in the Fig. 2.10.

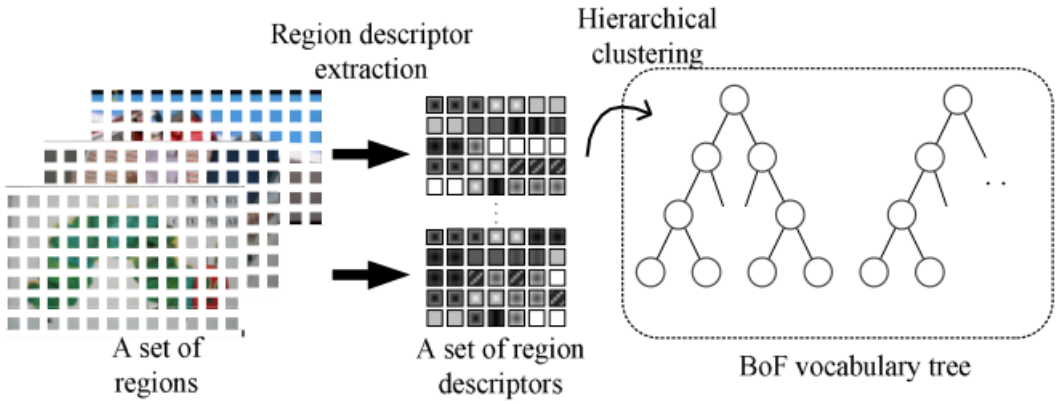


Figure 2.10: Feature extractiona and generation process of BoF vocabulary as proposed in [Cho et al., 2011].

3. BoF Signature Matching: Once the BoF signatures are generated, the linear combination of the distances are measured between training and test images.

In a similar fashion, in the work of [Botterill et al., 2008], BoF model is used for robot localization through scene recognition. the proposed approach [Botterill et al., 2008] uses a combination of shape descriptor, SURF [Bay et al., 2008] and hue histogram descriptor [Filliat, 2007] to described the features before using these features to build a visual dictionary.

Fusion of global and local features for classification: Not only for image retrieval, early fusion is also used for image classification tasks, such as in [Ji et al., 2013; Chow and Rahman, 2007]. In the work of [Chow and Rahman, 2007], global features and the local features are combined using tree-structured representation and then the combined representation is passed to the classification process. The overview of the proposed strategy is illustrated in the Fig. 2.11. Following steps are performed in this work.

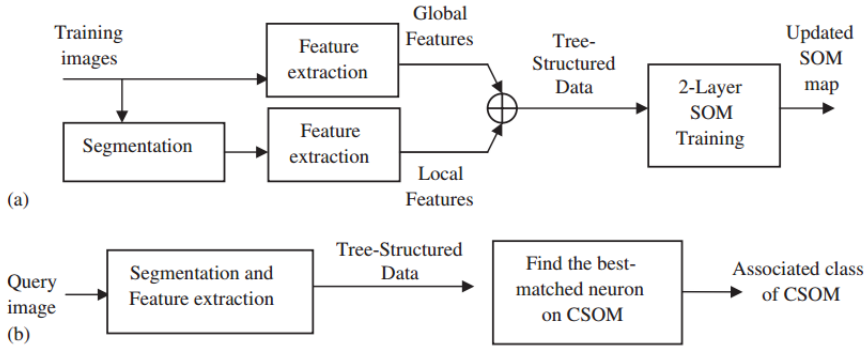


Figure 2.11: Overview of the classification system [Chow and Rahman, 2007] (a) training phase and (b) classification phase.

1. Image content representation: The image is segmented into several regions using JSEG segmentation method [Deng et al., 1999], where JSEG quantizes an image into several representative classes. Then the image is decomposed into several numbers of homogeneous regions. The regions are represented by local features, such as color moment, shape, texture features. A color histogram is used as a global feature, which used HSV color space. The features integration is achieved through a tree structure. In the tree structure, the each root node is assigned to a global feature and the child nodes are assigned to the local features. The integrated model is depicted in the Fig. 2.12.

2. SOM networks for classification: In this step, The combined tree-structured features are processed two-level self-organizing map (SOM). All the image regions, *i.e.,* child nodes in the tree structure, are processed by unsupervised SOM and then image regions are compressed by position vector in SOM map. Finally, a supervising concurrent SOM (CSOM) classifier [Neagoe and Ropot, 2002], which uses global representation, *i.e.,* root of the tree structure and the position vectors, is used for the overall classification task.

Fusion of object and background features for scene classification: For scene image classification, early fusion of object and background features is proposed in the work of [Ji et al., 2013]. In this proposal, the extracted local features are combined and embedded in BoF representation model before feeding them into SVM classifier for a final decision. The proposed framework is illustrated in the Fig. 2.13.

The steps involved in this approach are presented below.

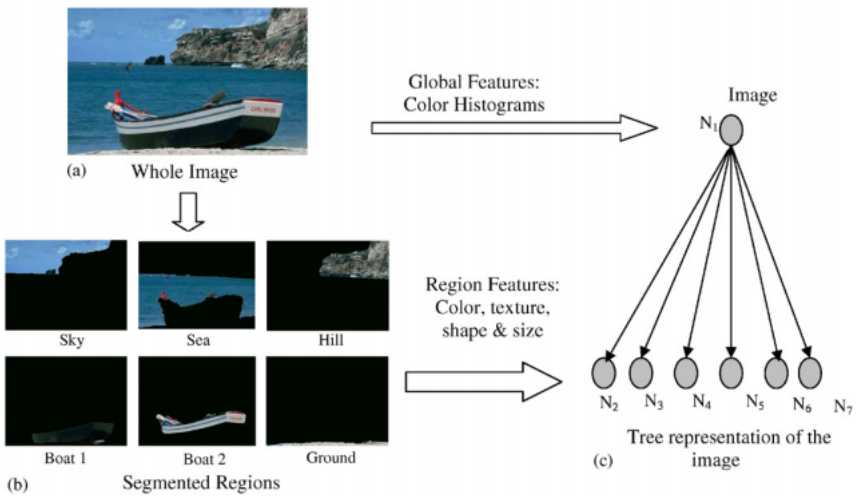


Figure 2.12: Image content representation by integrating global features and local features as proposed in [Chow and Rahman, 2007].

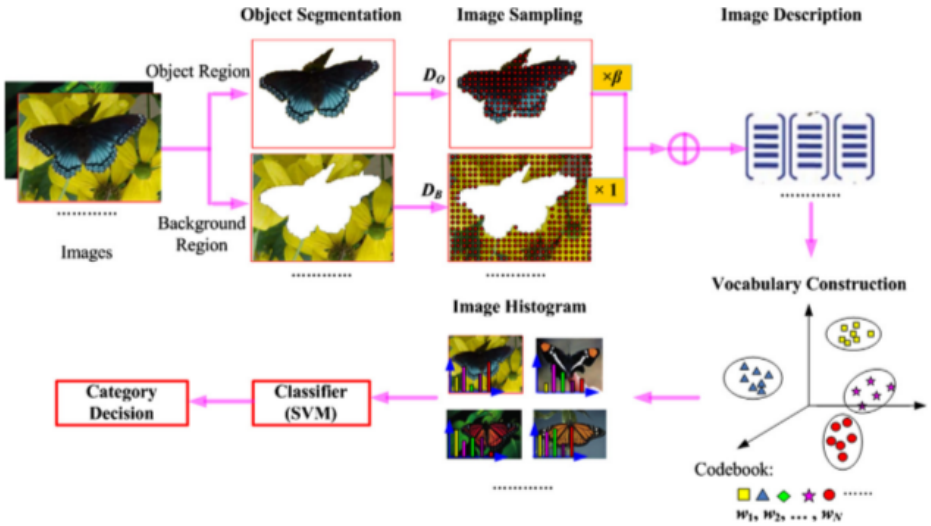


Figure 2.13: The framework for object and background feature fusion for scene image classification as proposed in [Ji et al., 2013].

1. Object region detection and segmentation: In this step, region-contrast (RC) method [Cheng et al., 2011], a bottom-up saliency detection method, is employed for object region detection and segmentation.

2. Patch sampling: The interest and distinguished image patches are identified using dense sampling method [Fei-Fei and Perona, 2005]. In the dense sampling method, an evenly sampled grid spaced at $10by10$ pixels is adopted. The patch size is randomly sampled between scale 10 and 30 pixels.

3. Object-enhanced patch description: In this step, the images are described by SIFT [Lowe, 2004] features. The SIFT descriptors in each image are divided into two parts, *i.e.,* object descriptors (D_O) and background descriptors (D_B). The descriptions of the object and

the background regions are combined using assigned weight on the part as presented in the Eq. 7.

$$D_I = \beta * D_O + D_B \tag{6}$$

where β is the assigned weight for the object descriptors. The object regions keep more important information than background regions. Therefore, more weight is assigned ($\beta > 1$) for object descriptors in the Eq. 7.

4. Vocabulary construction: The combined features are quantized into visual words using k-means clustering. The images are represented bu the histogram of the visual words.

5. Finally, non-linear SVM is used for image classification.

Fusion of CNN and SIFT features for image retrieval: The deep features or CNN features are quite effective to address different computer vision tasks, such as image classification, detection, recognition, retrieval. Although the deep features are very successful, it is still a debatable issue whether CNN features will always be able to produce commendable performance over hand-crafted features, such as SIFT. In the works of the [Chandrasekhar et al., 2015; Zheng et al., 2016], it is presented that the CNN features are not always able to outperform SIFT feature. Instead, the combination of hand-crafted features and CNN features could be effective. In the work of [Yan et al., 2016], CNN features and SIFT feature are combined to generate complementarity CNN and SIFT (CCS) for image retrieval. The CCS is a multi-level representation, *i.e.,* scene-level (CNN feature), object-level (CNN feature) and point-level (SIFT), of the images. The illustration of the representation is depicted in the Fig. 2.14.

Following steps are performed in this combination strategy [Yan et al., 2016]:

1. Scene-level representation: The global representation of the images is captured in scene-level representation (f_s), where CNN features are used. The CNN features are extracted from the pool5 layer in GoogLeNet [Szegedy et al., 2015].

2. Object-level representation: CNN features are used to describe the object-level representation. Initially, the candidate object regions are detected by egdebox [Zitnick and Dollar, 2014] strategy. Then the CNN features are extracted using pool5 layer in GoogLeNet from the candidate regions. In the next step, the extracted features are pooled to a fixed length vector (f_o) using max pooling, sum pooling and VLAD pooling methods.

3. Point-level representation: Hand-crafted feature, such as RootSIFT [Arandjelović and Zisserman, 2012], is used to describe the point-level representation. This representation is included mainly due to two reasons. First, it preserves the geometric invariance in image representation and second, the hand-crafted feature can produce stable performance without requiring a supervised learning process. Several post-processing steps, such as VLAD, $L2$ normalization and PCA, is applied to generate the final feature vector (f_p).

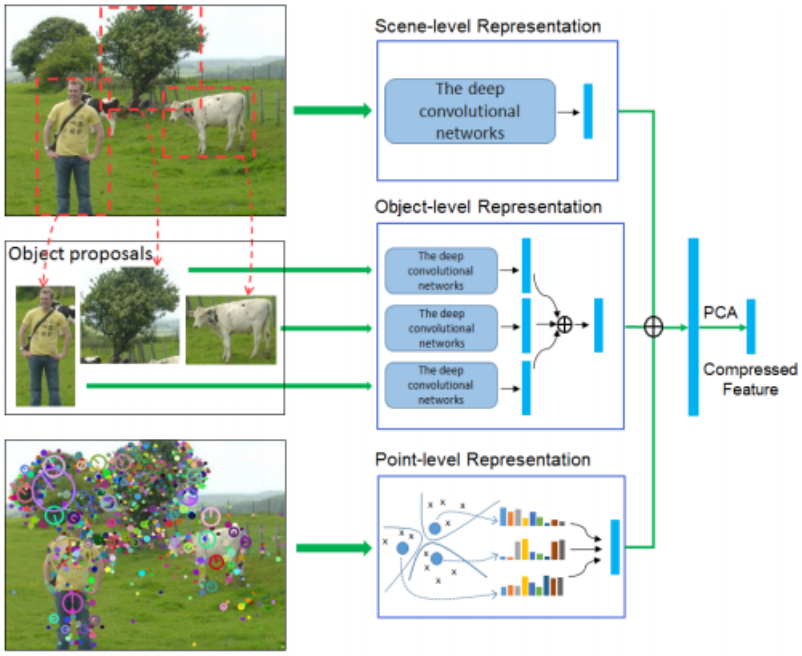


Figure 2.14: Three level representations and combination strategy as proposed in [Yan et al., 2016].

4. Fusion of the three-level representation: In this step, three representations are concatenated to generate the final representation (f).

$$f = [f_s, f_o, f_p] \tag{7}$$

The representation vector is very high in dimension. Thus, the PCA and $L2$ normalization are applied on f to generate compact final representation.

Several other early fusion strategies are proposed in the literature. For example, in the work of [Wang et al., 2009], HOG-LBP features are integrated and fed into SVM classifier to tackle the partial occlusion in human detection application. In the work of [Schwartz et al., 2009], descriptor combinations of edge-based features, texture and color information are used for human detection. As the combined descriptor is very high in dimension, partial least square is applied to obtain lower dimensional subspace. In this context, several other works are proposed [Tuzel et al., 2007; Chen and Chen, 2008; Wu and Nevatia, 2008] and they use different combinations of features, such as intensity, gradient, edges, covariance descriptors, etc. Another strategy, such as in [Madhusudhanarao et al., 2015], combines low-level and high-level features using KL divergence algorithm [Goldberger et al., 2003] for CBIR in medical imaging.

2.4.2 Late fusion strategies

In the late fusion category, several strategies are proposed in the literature. We present few selected of these strategies in this section.

Score-based late fusion of low level features for CBIR: Late fusion can be implemented on either score-based or ranked based approaches. A comparison between the most classical late fusion approaches is discussed in [Chatzichristofis et al., 2010b; Neshov, 2013] for image retrieval. In the work of [Neshov, 2013], score-based late fusion method are presented to combine multiple features. This strategy is presented below.

1. Let us consider, Im is an image in the image database of N images. The score, *i.e.,* the distance to the query, for Im in a list, L_j, is denoted by $(S_j(Im))$.

2. In the score-based method, the final score of the each retrieved images $(S_f(Im))$ in the arranged lists are computed and is arranged in ascending order. This generates a final list (L_f). This CombSUM method [Depeursinge and Müller, 2010], as presented in the Eq. 11, is used to compute the final score.

$$S_f(Im) = \sum_{j=1}^{N_j} S_j(Im) \tag{8}$$

In this method, the score in the each list (L_j) is not in same range. Therefore, each list may influence differently in the final combination. This issue is handled using normalization process such as CombSUM with Min-Max normalization, CombSUM with Z-Score normalization. Let us consider, $\overline{S_j(Im)}$ is the normalized score. The CombSUM with Min-Max normalization can be calculated as shown in the Eq. 9.

$$\overline{S_j(Im)} = \frac{S_j(Im) - S_j^{min}}{S_j^{max} - S_j^{min}} \tag{9}$$

Where S_j^{min} and S_j^{max} are the minimum and maximum score in L_j. The CombSUM with Z-Score normalization can be calculated using the Eq. 10.

$$\overline{S_j(Im)} = \frac{S_j(Im) - \mu}{\sigma} \tag{10}$$

Where μ is is the average of the un-normalized scores and σ is the standard deviation. Finally, using one of the Eq. 9 or Eq. 10, the final score can be calculated as follows:

$$S_f(Im) = \sum_{j=1}^{N_j} \overline{S_j(Im)} \tag{11}$$

Rank-based late fusion of low level features for CBIR: Rank-based strategy [Neshov, 2013] is an effective way to combine multiple features. The main steps of the rank-based strategy are presented below:

1. Let us consider, Im is an image in the image database of N images. The rank to the query, for Im in a list, L_j, is denoted by $R_j(Im)$.

2. The rank-based method can be computed using Borda count (BC) or Inverse Ranking Position (IRP). In Borda count, each image in the list, L_j, is assigned with Borda count points (BC). The most relevant image in the each list assigned with maximum Borda count points and subsequent images are assigned with less Borda count according to their relevancy with the query image. The Borda count is computed as presented in the Eq. 12.

$$BC_j(Im) = N - R_j(Im) \tag{12}$$

Here, N is total number of images in the dataset and value for $R_j(Im)$ is in between 0 to $N-1$. Final ranked list is produced by calculating total BC points.

$$BC(Im) = \sum_{j=1}^{N_j} BC_j(Im) \tag{13}$$

In IRP, the lists are merged by calculating IRP distance for each image as presented in the Eq. 14.

$$IRP(Im) = \frac{1}{\sum_{j=1}^{N_j} \frac{1}{R_j Im}} \tag{14}$$

The score-based and the rank-based methods are used in the work of [Chatzichristofis et al., 2010b] to combine the responses of the multiple descriptors. The descriptors used are color and edge directivity descriptor [Chatzichristofis and Boutalis, 2008a], fuzzy color and texture histogram [Chatzichristofis and Boutalis, 2008b], texture directionality histogram descriptor [Chatzichristofis and Boutalis, 2010] and spatial color distribution descriptor [Chatzichristofis et al., 2010a]. The proposed late fusion framework is illustrated in the Fig. 2.15.

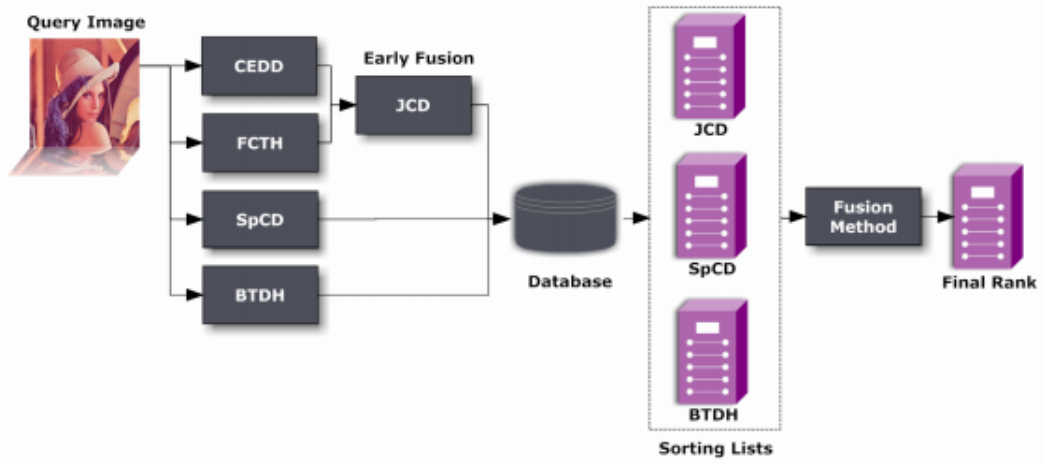


Figure 2.15: Late fusion implementation process as presented in [Chatzichristofis et al., 2010b].

Different late fusion methods, such as CombSUM, Borda count with CombSUM, IRP, Z-Score, which are already explained above, are applied on the ranking list of the each descriptor to generate the final response.

CBIR using weight based multi-feature fusion: Another way to do late fusion is using assigned weight to the each feature during similarity measurement between query and database. [Huang et al., 2010] proposed a weight based late fusion method where HSV color moment features [Brunelli and Mich, 2001] and Gabor filter texture descriptors [Yang et al., 2003] are combined for CBIR. The proposal is illustrated in the Fig. 2.16.

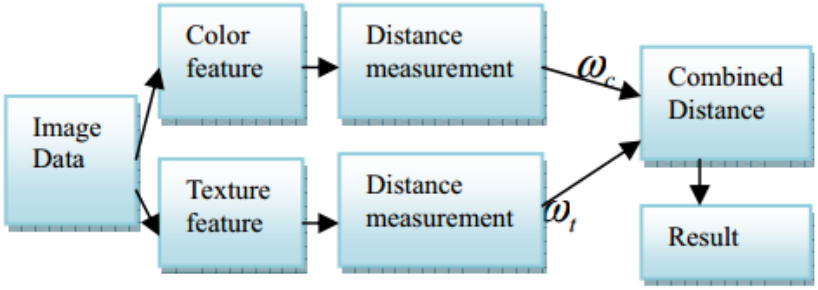


Figure 2.16: Fusion of color moment and texture features as proposed in [Huang et al., 2010].

Database images, as well as query images, are represented by color moment features and texture descriptors. Then, the similarity is measured between each query and database images separately for each feature types. In this work [Huang et al., 2010], Euclidean distance is used for similarity measurement as presented in the Eq. 15.

$$D(q,s) = \{\sum_{i=0}^{L-1}(q_i - s_i)^2\}^{1/2} \tag{15}$$

Where L is the feature vector dimension, $q = (q_0, q_1, ..., q_{l-1})$ is the query feature and $s = (s_0, s_1, ..., s_{l-1})$ is the database features.

The computed Euclidean distances are normalized using the Eq. 16.

$$D(q,s) = \frac{1}{L}\sum_{i=0}^{L-1}(1 - \frac{|q_i - s_i|}{max(q_i, s_i)}) \tag{16}$$

Let us consider, the distances computed for color moment and texture features are represented by $D_c(q,s)$ and $D_t(q,s)$. The global similarity measurement $D(q,s)$ is calculated by combinng $D_c(q,s)$ and $D_t(q,s)$ distances based on the assigned weight as presented in the Eq. 17.

$$D(q,s) = \frac{\omega_c.D_c(q,s) + \omega_t.D_c(q,s)}{\omega_c + \omega_t} \tag{17}$$

Where ω_c and ω_t are the color moment and texture feature weight respectively. The weight values, *i.e.,* $\omega_c = 3$ and $\omega_t = 1$ are selected to compute global similarity.

One of the drawbacks of this method is that the assigned weight is fixed for all the image type or image database. The weights cannot be assigned depending on the image content.

Another weight based multi-features late fusion is proposed by [Huang et al., 2015b]. Unlike in the work of [Huang et al., 2010], where weights are fixed, in the strategy of [Huang et al., 2015b], weights are assigned and re-adjusted depending on the number of features, its length and the relevant feedback of user's. At first, the similarity distances between query features and the database features are measured separately using three different type of features, such as color feature [Han and Ma, 2002], shape features [Ma et al., 2011] and texture feature [Jie and Li, 2008]. The similarity distance (D_i^*) is then normalized in between (0,1) using means (μ_{D_i}) and standard deviations (σ_{D_i}) of the distances.

$$D_i^* = (1 + ((D_i - \mu_{D_i})/3\sigma_{D_i}))/2 \tag{18}$$

Finally, the total similarity distance is computed as presented in the Eq. 19.

$$D = \omega_i D_i (i = 1, 2, 3) \tag{19}$$

Here, ω_1, ω_2 and ω_3 are the weights assigned to color, shape and texture features. The weights can further be readjusted on the basis of relevance feedback.

Quite similar late fusion strategy can be found in the work of [Dubey et al., 2010], *i.e.,* the similarity distance is computed individually for each feature. Then the distances are combined by taking the average of the calculated distances to produce the final response. Five different features, such as color moment [Stricker and Orengo, 1995], color histogram [Stricker and Orengo, 1995], texture [Hejazi and Ho, 2007] and edge histogram descriptor [Amato and Lecce, 2003] are used for image representation.

Late fusion in scene categorization: One of the problems in scene categorization is how to understand and describe an image semantic scene by making use of low-level features to represent high-level semantic meanings. To mitigate this issue, multiple features combination using late fusion are proposed in the literature. When considering scene categorization, late fusion is performed slightly differently. It usually involves a weighted voting strategy from the outputs of the classifiers associated with the individual descriptors, such as in [Zhang et al., 2011b; Zhou et al., 2013].

A scene categorization strategy using multiple low-level features in a BoF model is proposed by [Zhang et al., 2011b]. The main steps of this approach are explained below.

1. Image representation: To represent images using low-level image features, every image is segmented into regions using normalized cuts algorithm [Shi and Malik, 2000]. The normalized cut algorithm treats image segmentation as a graph partitioning problem. It measures both the total dissimilarity between the different groups as well as the total similarity within the groups. Low-level features are extracted from each of these regions. The feature set consists of 36 low-level features, *i.e.,* 18 color features, 12 texture features and 6 shape features. Then features are represented by BoF model. The entire process is illustrated in the Fig. 2.17.

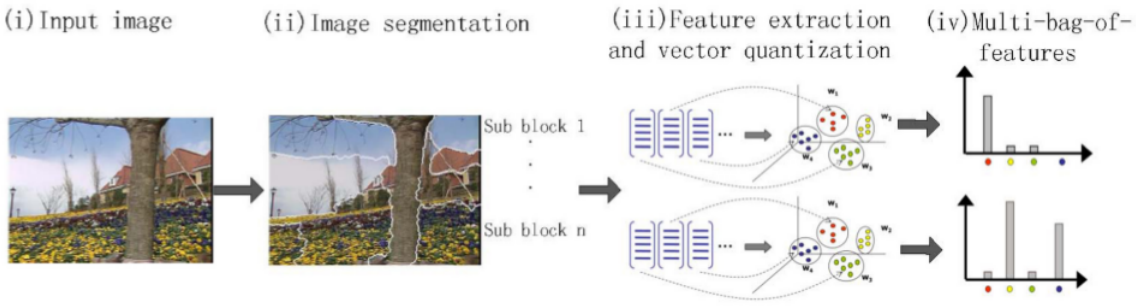


Figure 2.17: Image representation model as proposed in [Zhang et al., 2011b].

2. Scene categorization: In this step, the generated BoF for the training images are used to train SVM model. For 36 types of extracted low-level features, 36 SVM models are obtained where each SVM classifier is corresponding to one type of low-level feature. For a test image (I_q), 36 types of features are extracted and these features are used in multi-bag-of-features by vector quantization with the aid of pre-trained codebooks. The final categorization result is determined by a weighted voting from the output ($[r^1_{I_q}, ...r^{36}_{I_q}]$) of the each of 36 SVM classifier as presented in the Eq. 20.

$$C(I_q) = max[V(c_1)...V(c_k)] \qquad V(c_j) = \sum_{j=1}^{36} \omega_j \alpha_j \tag{20}$$

Where ω_j is the voting weight of the classifier and $\alpha_j = 1$ when $r^j_{I_q} = c_j$ and $\alpha_j = 0$ in other conditions.

In the scene classification context, [Zhou et al., 2013] put forward a strategy by combining local feature in a multi-resolution bag-of-features model. The overview of this approach is depicted in the Fig. 2.18.

In this strategy [Zhou et al., 2013], three resolution images are constructed by sub-sampling the input images. During the training process, the local features, such as SIFT, are extracted from all three resolution with dense regions. Then the features are quantized to form a visual codebook using the k-means clustering method. To incorporate spatial information, two modalities of horizontal and vertical partitions are adapted to partition all resolution images into sub-regions with different scales. Each sub-region is then represented as a histogram of codeword occurrences by mapping the local features to the codebook. The multiple category scenes are classified with SVM trained by the one-versus-all rule, *i.e.,* a classifier is learnt to separate each class from the rest. The scenes are classified using non-liner SVM with χ^2 kernel as presented in the Eq. 21.

$$K(V_i, V_j) = exp(-\frac{1}{\gamma} \sum_{ch=1}^{3} \beta_{ch} D^{ch}_{\chi^2}(V^{ch}_i, V^{ch}_j)) \tag{21}$$

Where ch denotes the three feature channels corresponds to three resolutions. V^{ch}_i and V^{ch}_j are

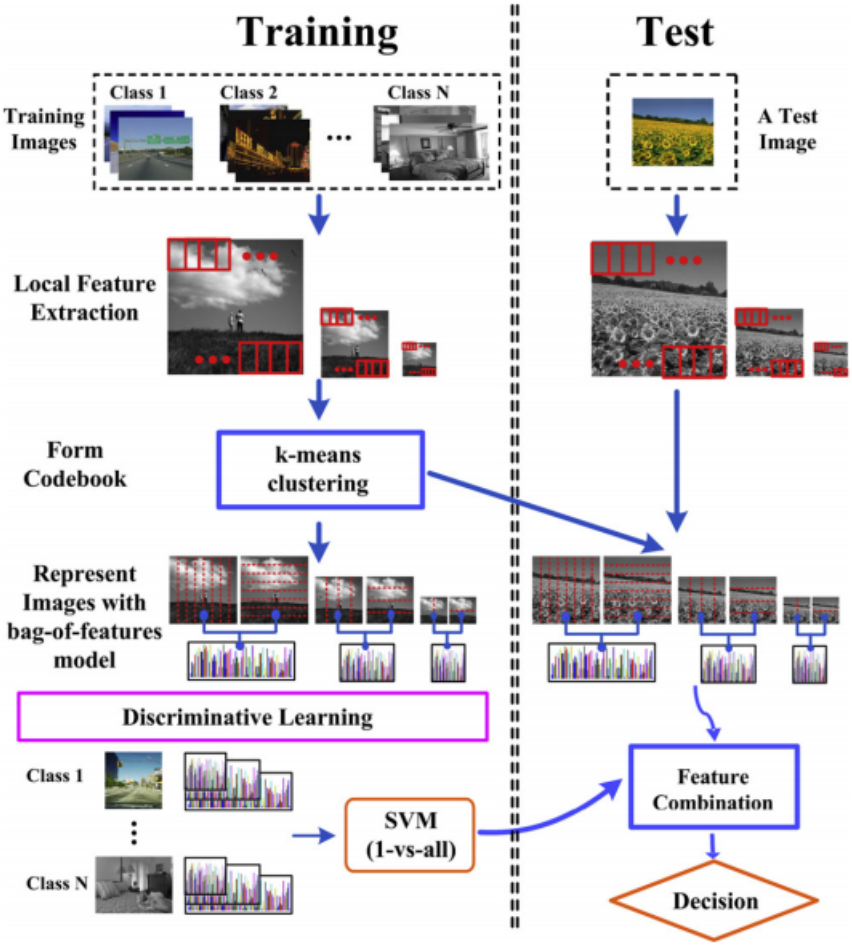


Figure 2.18: Overview of the multi-resolution bag-of-features based scene classification as proposed in [Zhou et al., 2013].

the i^{th} j^{th} representation vectors of the training images V_i and V_j. $D_{\chi^2}^{ch}$ is defined in the Eq. 22.

$$D_{\chi^2}^{ch} = \frac{1}{2}\sum_{l=1}^{M}\frac{(u_l - w_l)^2}{|u_l + w_l|} \tag{22}$$

Where M is the dimension of the image representation. γ can be defined as in the Eq. 23.

$$\gamma = (\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{ch=1}^{3}\beta_{ch}D_{\chi^2}^{ch}(V_i^{ch}, V_j^{ch}))/N^2 \tag{23}$$

Where N is the total number of training images. Finally, the representations of different resolution channels are combined to reach a final decision. The final decision function of image x is presented in the Eq. 24.

$$y(x) = \arg\max_{c=1,...,C}(K(x)^T\alpha_c + b_c) \tag{24}$$

Where y is the class label of test image x, $K(x) = (K(V_1, V_x), ..., K(V_N, V_x))^T$, b is the threshold paremeter for each class and α is the weight parameter.

Selective weighted late fusion for visual concept recognition: In the work of [Liu et al., 2014], visual concept recognition problem is addressed by late fusion of visual and textual features. The visual and textual features are combined with a multimodal approach which predicts the visual concept based on selectively weighted late fusion (SWLF) approach, where score level fusion is used. The proposed strategy is illustrated in the Fig. 2.19.

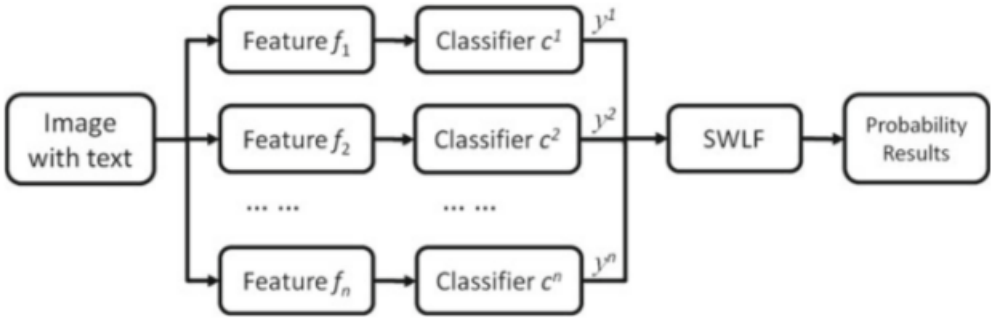


Figure 2.19: A framework for the SWLF scheme as proposed in [Liu et al., 2014].

Following steps are involved in this approach.

1. The SWLF requires a training phase to select the best experts or classifiers and corresponding weights for each visual concept.

2. The dataset is divided into two parts, *i.e.,* training and validation set. Each visual concept is trained with a binary classifier (expert) for each visual feature. Thus, there are multiple experts are generated as each concept is described by multiple features.

3. The quality of the expert is evaluated by the interpolated Average Precision (iAP) metric (in the range of 0 to 1) which is computed from validation set. Higher the value of iAP, the more weight is assigned to the expert during late fusion. For a given visual concept (k), the computed iAPs are normalized into w_k^i. The sum of weighted late fusion is computed as presented in the Eq. 25.

$$z_k = \sum_{i=1}^{k} (w_k^i * y_k^i) \tag{25}$$

   Where y_k^i is the score of the i^{th} expert of concept k.

4. The visual concepts are described by several global features, such as color information in the HSV space, color histograms, color moments, texture [Ojala et al., 1996], color LBP [Zhu et al., 2010], and local features, such as C-SIFT, RGB-SIFT, HSV-SIFT [Sande et al., 2010], and DAISY [Tola et al., 2010].

In addition, several other late fusion strategies can be found in the literature. For example, a graph-based query specific fusion approach is proposed in the work of [Zhang et al., 2012b], where several retrieval sets of images are merged and re-ranked by creating link analysis on a fused graph. [Raina et al., 2014] proposed a feature combination strategy using fuzzy heuristics rule-set technique, where color histogram, grey-level co-occurrence matrix and texture features are fused and the user can select relevancy of each feature. For face classification, LBP, Gabor and HOG features are combined in the work of [Maatta et al., 2012], where SVM is applied to each feature for classification and score-based fusion is used to decide the final output. In other strategies, such as in [Gehler and Nowozin, 2009; Risojevic and Babic, 2013], boosting can learn individual classifier and the weights of the each classifier for the combination.

2.4.3 Other fusion strategies

Although most of the fusion strategies are accommodated either in early or in late fusion category, some fusion strategies can be termed as intermediate or sequential or progressive fusion. In the below, we present some of the strategies in this category.

Image matching using multiple local features: One of the strategies in sequential fusion is that one descriptor is considered as a filter before using another one on the remaining subset of images or regions in the images. For example, in the work of [Cao et al., 2010] such a strategy is proposed for image retrieval task by fusion of multiple local features, such as Affine-SIFT and color moments. The steps performed in this proposal is presented below:

1. To begin with, MSER [Matas et al., 2002] is used to detect the interest points in the images. The MSER points are described by 18-dimension color moment invariants descriptor.

2. For a given query image, each MSER point which is described by color moment is matched with database images using Mahalanobis distance below a predefined threshold.

3. Then, Affine-SIFT [Morel and Yu, 2009] features are computed only inside the matched MSER points between the query image and the database images using the nearest neighbor distance ratio method. If there is no Affine-SIFT feature, then the corresponding MSER region will not be considered. The final match is identified if the Eq. 26 satisfies.

   $$dist(A, B)/dist(A, C) < r \tag{26}$$

   Where B and C are the first nearest neighbor (1-NN) and the second nearest neighbor (2-NN) of the interest point A in the query, and r is the distance ratio threshold.

Combining complementary kernels in complex visual categorization tasks: For visual categorization task, several works, such as in [Vedaldi et al., 2009; Gehler and Nowozin, 2009; Yan et al., 2009], combine multiple descriptors. One of the likeable explanation for feature combination is the use of multiple kernels learning (MKL). The advantage of MKL is the possibility to jointly learn the weighting of the different channels and classification functions. In the work of [Picard et al., 2010], MKL is used to combine multiple descriptors for the categorization task. The fusion takes place at the intermediate stage of the whole process. The proposed strategy is illustrated in the Fig. 2.20.

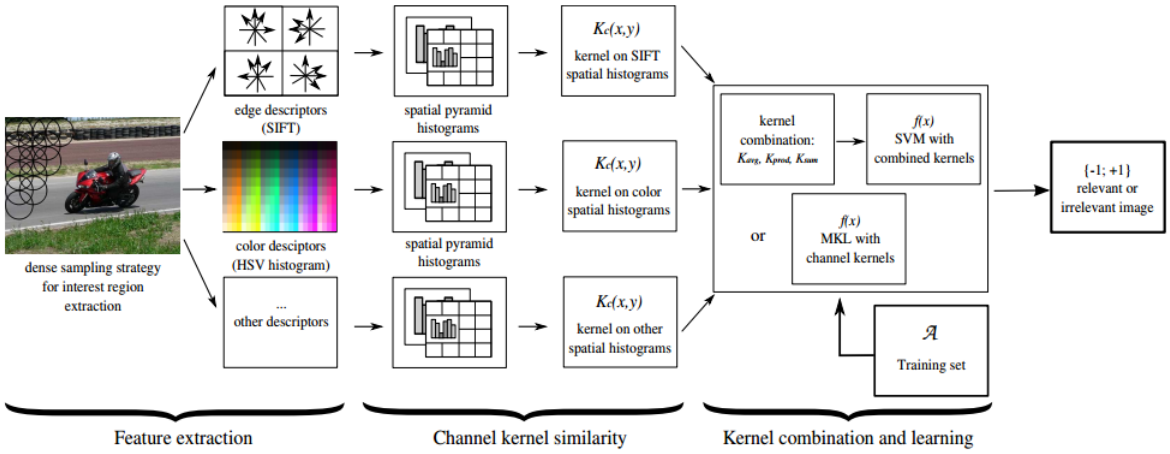


Figure 2.20: Overview of Fusion of complementary kernels as proposed in [Picard et al., 2010].

The following steps are executed in this approach.

1. Extraction of local descriptors: A set of descritors, such as edge and color descriptor, describes the images using dense sampling strategy. The descriptors are computed over a local neighborhood at each patch with associated scale. Opponent color SIFT (oc-SIFTs) [van de Sande et al., 2010] is used as a local edge descriptor. It is a concatenation of three one dimensional SIFT histogram based on the channels of the opponent color space. The histogram dimension of os-SIFTs is 384 ($= 128 * 3$). HSV color space is used to extract local color descriptor. At each grid position, color histogram is computed over the region by quantizing HSV space, which leads to 144 descriptor dimension ($= H * S * V = 8 * 6 * 3$).

2. Channel kernel similarity: Spatial pyramid matching (SPM) [Lazebnik et al., 2006] is used to incorporate the spatial information for each extracted image descriptors. At each scale in SPM, the image is divided into regular grid and then a histogram of visual word is computed for each region. In this strategy [Picard et al., 2010], the two scale is used, *i.e.,* scale 0 for global histogram and scale 1 is decomposed into $3by3$ overlapping windows. For each region (r) and each descriptor channel (c), the Gaussian kernel similarity is defined between two images (x_i and x_j) is defined as in the Eq. 27.

$$k_{c,r}(x_i, x_j) = e^{-\gamma_c d_{\chi^2}(x_i^{(c,r)}; x_j^{(c,r)})} \tag{27}$$

Where $x^{c,r}$ is the histogram of visual words for x associated with descriptor channel c, region r and $d_{\chi^2}(.,.)$ is the χ^2 distance, and γ is Gaussian kernel parameter.

3. Kernels combinations: The kernels can be combined by using early fusion (*i.e.,* product kernerl) or weighted sum fusion. In this work, a non-sparse combination via l_1 MKL [Rakotomamonjy et al., 2008], which does not ignore informative image modalities is proposed. In order to find a optimal classification function ($f(x)$), two steps are performed. First, the optimal γ is determined by cross-validation and second, the kernels are combined for the optimal γ value. The γ and the kernel combination coefficient (β_m) are learned simultaneously. For each c, a set of M kernels ($K_{c,\gamma}$) is formed. The optimal function is given in the Eq. 28.

$$f(x) = \sum_{i=1}^{N_e} \alpha_i y_i \sum_{c=1}^{N_c} \sum_{\gamma=\gamma 1}^{\gamma M} \beta_{c,\gamma} kc, \gamma(x, x_i) - b \tag{28}$$

where the joint optimization is performed on α_i and $\beta_{c,\gamma}$.

Local and global feature combination for object class recognition: A combination of global features and local features are used for object class recognition in the work of [Lisin et al., 2005], where global features are first used for coarse classification, before exploiting more expensive local features in order to refine classification. The features are combined using a two-tier hierarchy of classifiers. The following steps are followed to combine the features.

1. At the first tier, the classification is performed using global features, such as texture [Ojala et al., 2002] and shape features [Ravela, 2003]. The classes, which are not separable in the global features space, are combined into super classes.

2. In the second tier, the local feature classifier, such as SIFT [Lowe, 2004], is trained to classify the classes present in the superclass.

3. For a given query, if the query is classified into a super class by a global classifier, then the query is passed to local feature classifier for more accurate classification.

4. The two-tier strategy accelerate the overall process. Support vector machine (SVM) is used as global features classifier and non-parametric density is used for local features classifier.

Local feature matching using global information: The features matching task is addressed by using local features and global features in the proposal presented in the work of [Li et al., 2012a]. As the local features only use neighborhood information, thus it has a deficit of global concept during feature point matching. This issue can be tackled by using the global representation of the images. In the work of [Li et al., 2012a], the matched local features between the

images are considered as a filter to generate the global features. The proposed approach is two stages process.

1. In the first stage, the initial matched feature points (IMFPs) between two images are obtained by the local feature. The IMFPs create a new coordinate system which consists of spatial locations of the matched features.

2. In the second stage, the global features are used to filter out the mismatches in the previous step. The global features are generated only for the IMFPs using the coordinate system created in the previous step. The points which are not matched in the first step are not included to generate global feature.

3. SIFT [Lowe, 2004], SURF [Bay et al., 2008], PCA-SIFT [Ke and Sukthankar, 2004] are used as the local features in this work.

Additionally, in some other works [Azad et al., 2009; Tao et al., 2013; Elnemr, 2016], different local features such as Harris, MSER detectors are used as a filter to detect interest points. These points are described by local descriptors such as SIFT, SURF, for image retrieval tasks. The progressive fusion is performed in the work of [Li et al., 2002], where color histogram acts as a first level filter followed by texture and wavelet descriptor as a secondary filter.

2.5 Conclusions

In this chapter, we studied, reviewed and presented inclusive works on CBIR system. We begin with the general concept of CBIR system. Gradually, the presentation moved to the important components of the CBIR system with the preeminent attention on the feature fusion strategies in CBIR. As feature extraction is one of the initial and pertinent steps, different feature types, such as conventional hand-crafted features and recently developed deep features, are discussed and presented. We observed that the conventional feature detectors are characteristically diverse. Hence, the spatial complementarity between the detectors could be beneficial in image retrieval. Then we presented several representation models of the feature in CBIR, before turning our focus on the feature combination strategies.

The fusion strategies are categorized into three types, *i.e.,* early, late and other (intermediate or sequential) fusion, based on the approaches proposed in the literature. Fusion strategies are investigated depending on the position in the whole process chain and the application targeted. The detailed discussion related to the fusion strategies are covered in the Sec. 2.4. We observed that certain approaches, such as concatenation based, rank-based, score-based, are quite popular and adopted and proposed for different computer vision related tasks. Although, they may or may not always be effective. A global overview of the fusion approaches proposed in the literature are summarized in the Tables 2.1, 2.2, and 2.3.

Fusion strategy			Description	Application targeted	Related work
Early	Late	Intermediate			
✓			SIFT-LBP & SIFT-HOG feature concatenation	Image retrieval	[Yu et al., 2013]
✓			Color moment & LBP feature concatenation	Image retrieval	[Choudhary et al., 2014]
✓			Color & texture features fusion based on weight	Image retrieval	[Yue et al., 2011]
✓			Shape descriptors combination by GP framework	Image retrieval	[da S. Torres et al., 2009]
✓			Region descriptors fusion by BoF model	Object detection	[Cho et al., 2011]
✓			Global & local feature combination using tree-structured representation	Image classification	[Chow and Rahman, 2007]
✓			Object & background features combination by assigned weight	Scene classification	[Ji et al., 2013]
✓			CNN & SIFT features concatenation	Image retrieval	[Yan et al., 2016]
✓			Combination of HOG & cell-structured LBP augmented feature	Human detection	[Wang et al., 2009]
✓			Edge, texture & color features concatenation	Human detection	[Schwartz et al., 2009]
✓			Combination of covariance descriptors	Classification	[Tuzel et al., 2007]
✓			Intensity-based & gradient-based features combination	Human detection	[Chen and Chen, 2008]
✓			Edgelet, HOG & covariance descriptor fusion in cascade structure	Object detection	[Wu and Nevatia, 2008]
✓			Fusion low-level & high-level features by KL divergence algorithm	Image retrieval	[Madhusudhanarao et al., 2015]

Table 2.1: Summary of the different fusion strategies in CBIR: Part 1.

Fusion strategy			Description	Application targeted	Related work
Early	Late	Intermediate			
	✓		Score-based fusion of low-level features	Image retrieval	[Neshov, 2013]
	✓		Rank-based fusion of low-level features	Image retrieval	[Neshov, 2013]
	✓		Score-based & rank-based fusion of multiple descritprs	Image retrieval	[Chatzichristofis et al., 2010b]
	✓		Color moment & texture features weight-based fusion during similarity measurement	Image retrieval	[Huang et al., 2010]
	✓		Color, shape & texture feature fusion based on adjusted weight	Image retrieval	[Huang et al., 2015b]
	✓		Multi features fusion by averaging similarity distance	Image retrieval	[Dubey et al., 2010]
	✓		Color, shape & texture features fusion by weighted voting	Scene categorization	[Zhang et al., 2011b]
	✓		Multiple features combination by selective weighted late fusion	Visual concept recognition	[Liu et al., 2014]
	✓		Fused graph based fusion by re-ranking retrieval images	Image retrieval	[Zhang et al., 2012b]
	✓		Multi features fusion using fuzzy heuristics technique	Image retrieval	[Raina et al., 2014]
	✓		Score-based fusion of LBP, Gabor & HOG features	Face classification	[Maatta et al., 2012]
	✓		Gabor & SIFT fusins by hierarchical stacking	Image classification	[Risojevic and Babic, 2013]

Table 2.2: Summary of the different fusion strategies in CBIR: Part 2.

Fusion strategy			Description	Application targeted	Related work
Early	Late	Intermediate			
		✓	Sequential fusion of Affine-SIFT & color moments	Image retrieval	[Cao et al., 2010]
		✓	Global & local features combination by two-tier hierarchy of classifiers	Object class recognition	[Lisin et al., 2005]
		✓	Progressive fusion of local & global features	Image retrieval	[Li et al., 2012a]
		✓	Fusion of color histogram & wavelet features in two-level filter strategy	Image retrieval	[Li et al., 2002]
		✓	Fusion of SURF, color correlograms & improved color coherence vector	Image retrieval	[Elnemr, 2016]
		✓	Color SIFT & color histogram fusion in multiple kernel combination	Visual categorization	[Picard et al., 2010]
		✓	Fusion of multiple descriptors by browsing tree approach	Image retrieval	[Landré et al., 2001]

Table 2.3: Summary of the different fusion strategies in CBIR: Part 3.

Chapter 3

Query-by-example image retrieval by multi-descriptor fusion

3.1 Introduction

Query-by-example (QbE) is the most popular paradigm in CBIR system. The core of the QbE consists in the extraction of distinguished features from a dataset and then measurement of the resemblance between them. Hence, the key emphasis is on describing suitable image characteristics, which should coincide with the user's vision and perception of similarity of the images. The gap between the high-level perception of human and the low-level feature description is known as semantic gap. Thus, the pivotal focus is on to reduce the differences/gap between high-level semantic concepts and machine-defined image characteristics. The large diversity in the image feature descriptors, *i.e.,* local features, global features, are available for various CBIR related tasks. Thus, an image can be described using several feature descriptors. Instead of using a single type of feature descriptor to represent an image content, combined or fused use of multiple descriptors is propitious to better describe the image content. The primary focus in this chapter is on the proposal of the core of the thesis, *i.e.,* the proposal of a model for combining low-level and generic descriptors in order to obtain a descriptor of higher representativeness adapted to a given use case, while maintaining genericity in order to be able to index different types of visual contents. Therefore, we concentrate on designing a complete image retrieval search engine, named Fusion of Inverted Indices (FII) image search engine, for CBIR using multiple local image features. The heart of the FII search engine lies on the novel proposal of a fusion strategy for the multi-dimensional local features. The fusion strategy is developed on the inverted multi-indices data structure. The scope of this part of work, presented in this chapter, is highlighted in the Fig. 3.1.

The considered application being query-by-example, another major difficulty will be the complexity of the proposal, which will have to meet with reduced retrieval times, even with large

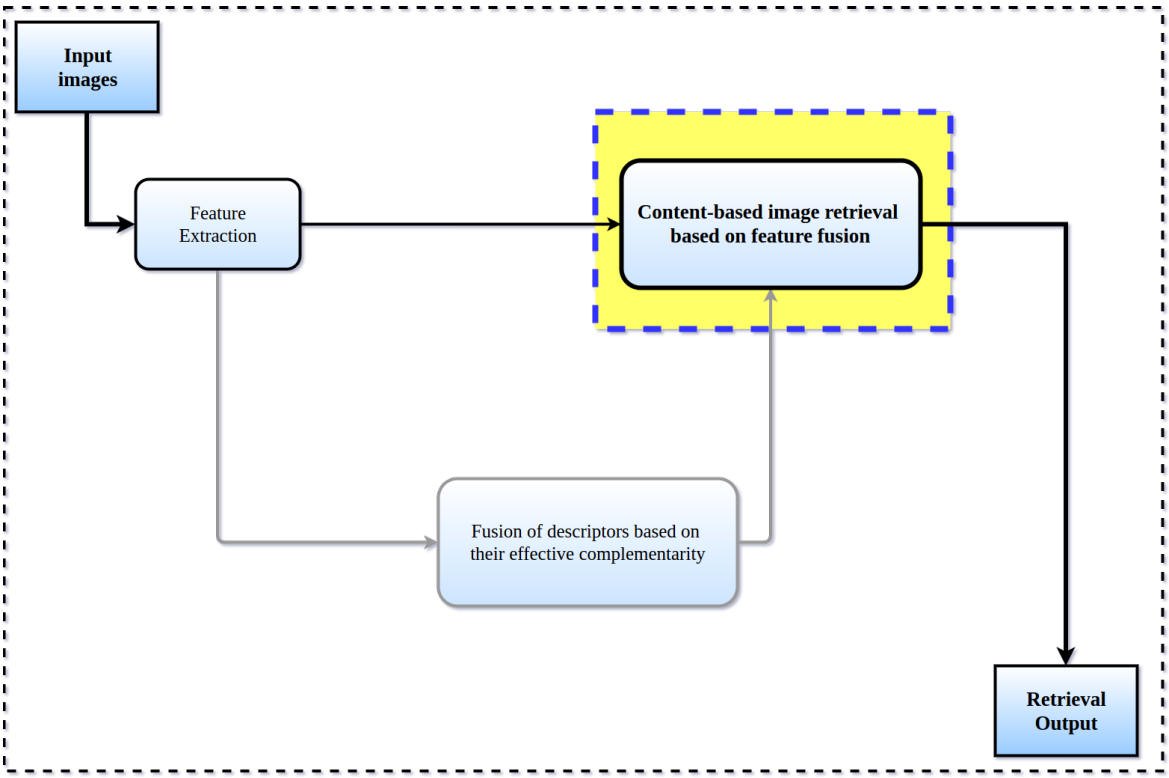


Figure 3.1: Overview of the thesis proposal. We discuss the highlighted step of the proposal in this chapter.

datasets. The relevance of the proposal will also depend on the effectiveness of the associated access method. Therefore, this chapter discusses the proposal of a query-by-example image search strategy using multiple descriptors fusion.

Finding the appropriate features to define the image content holds one of the basic keys for an effective CBIR system. Nowadays, the image content, image types or the image styles are very diverse such as natural scenery, architectures, buildings, facade, portraits, paintings, remote sensing images, satellite images, street view images, old photographs, etc. At the same time, literature on image descriptors is very rich [Datta et al., 2008; Tuytelaars and Mikolajczyk, 2008; Sande et al., 2010], providing several families to describe different image characteristics for different targets. Due to the large diversity of existing feature descriptors in CBIR, the image contents can be better represented by the combined use of several descriptors in order to explore their potentially diverse characteristics.

In the literature of CBIR, several types of image features are proposed to describe different types of image contents depending on the applications. In-depth discussion of image features is presented in the Sec. 2.2 of Chapter 2. In this proposal, we are interested in using local image features to describe image contents. The local features or low level features are normally a distinct image patch which is different from its immediate neighbhorhood. These distinct patterns, such as shape, color, texture, etc. are described by local image descriptors. In our proposal of image retrieval search engine, the local image descriptors are used jointly.

Several fusion approaches have been proposed in the literature, which we presented in Sec. 2.4 of Chapter 2. Whether it is an early fusion or a late fusion strategy, there are certain limitations involved. For example, in early fusion strategy, a single representation of the concatenated multiple features generates a very high dimensional feature vector which is not desirable during image retrieval. Thus, it is pertinent to investigate the best and most generic strategy to combine image characteristics. In this work, we propose a novel fusion method for efficiently combining multiple descriptors for QbE image retrieval, called Fusion of Inverted Indices (FII) image search engine. Fusion of inverted indices strategy is developed on inverted multi-index [Babenko and Lempitsky, 2012]. Inverted multi-index is a data structure which decomposes the high dimensional image descriptor space into n desired subspaces. Then, the best responses to a query in each subspace are retrieved and combined into one response that ensures better result than the traditional approaches based on classical inverted indices. The FII search engine allows combining any number of multidimensional image descriptors by integrating their responses to a query in finer subdivisions. The chapter is organized as follows: Sec. 3.2 is dedicated to the discussion of inverted indexing structures, the overview of the proposed fusion strategy is presented in the Sec. 3.3, Sec. 3.4 describes the detail of the proposed fusion methodology, followed by the evaluation in the Sec. 3.5, before concluding the chapter in Sec. 3.6.

3.2 Inverted indexing structures in image retrieval

Inverted indices, also known as inverted file indices, is a data structure to store the mapping of each content to its position in the database. It is first used in text-based search engine indexing algorithm where the goal is to reduce the querying time by finding relevant documents in which query word presents.

In computer vision, the use of inverted indices structure is first proposed in [Sivic and Zisserman, 2003] for object retrieval and similarity search. The inverted indexing structure is built around a visual vocabulary or codebook. The visual of descriptions of the image dataset, *i.e.,* feature description vectors, are quantized into several clusters to generate the codebook, which we presented in the Sec. 2.3 of Chapter 2. Inverted indexing stores the information of the list of feature descriptors which are used to generate codewords or the feature descriptors lie within the proximity of each codeword. During the similarity search of a given query, a set of closest codewords to the query are computed. Each of the closest codewords has a list of associated several feature descriptors in the database. Thus, these retrieved lists are the similar description to the query. An example is of classical inverted indexing is depicted in the Fig. 3.2.

As shown in the Fig. 3.2, the query (presented with the red color star) belongs to one of the clusters. Hence, the inverted indices structure returns all the feature descriptions, shown in the red color circle, associated with this cluster. There are certain advantages of using inverted indices structures:

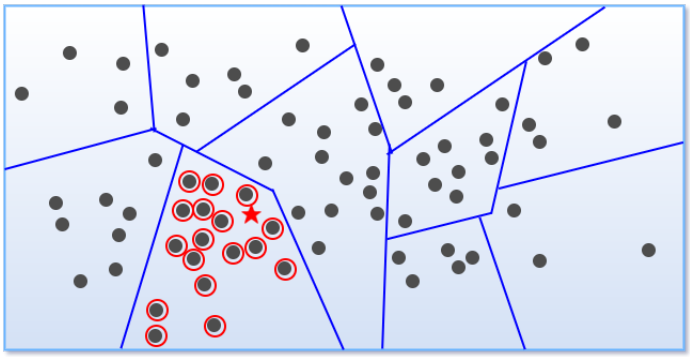


Figure 3.2: Similarity search using inverted indexing structure.

First, the exhaustive search process between the query and each description in the dataset is avoided and the similar descriptions are retrieved efficiently during the search process. Hence, the introduction of the inverted indexing structure accelerates the image retrieval process.

Second, the inverted indexing structure does not store original feature description vectors. Normally, it contains image identifiers of the image database. Thus, the memory usage during similarity search can be reduced conspicuously.

Although the inverted indexing structure is very efficient for image retrieval, the problem arises while tackling large image dataset consists of million of feature descriptions. For a very large volume of descriptions, a finer division of the search space is required to avoid the overpopulating of the inverted list with the several features descriptions. If the number of codewords is restricted to a small number, the return lists to a given query will be very large. This might include irrelevant descriptions in the returned similarity list. One of the ways to achieve finer search space is to increase the number of codewords. However, increasing number of codewords reduces the efficiency of the search process as it increases the time to build inverted indexing structure and increases the search process.

To overcome these limitations, the inverted multi-indices data structure is proposed in [Babenko and Lempitsky, 2012]. Inverted multi-indices is a data structure for efficient similarity search in a large dataset of high dimensional vector. It opens higher sparse subdivision of the search space without affecting overall processing time compared to classical inverted indexing [Sivic and Zisserman, 2003]. The vector quantization in inverted indices is replaced by product quantization (PQ) while building inverted multi-indices.

Product quantization (PQ) [Gray and Neuhoff, 1998] was proposed to improve the approximate nearest neighbor search [Jegou et al., 2011]. Higher dimensional vectors are split into low dimensional subspaces of a Cartesian product. These subspaces are quantized independently. The Euclidean distance between two vectors, which are epitomized by a subspace quantization index, is computed through quantized codes. The overall process enhances the search quality by limiting the quantization noise. PQ is integrated with inverted indexing in order to avoid exhaustive search, hence it boots searching speed.

The idea of PQ is used in the inverted multi-indices structure. In the inverted multi-indices, one high dimensional vector is decomposed into n smaller dimension sub-spaces. Considering the low indexing construction time and low query time, the high dimensional vectors are split into two parts. Then n PQ codebooks are computed by clustering each of the n sub-spaces separately. It is constructed as a multi-dimensional table which contains n lists of ordered codewords from the n corresponding codebooks. A given query is split into the same sub-spaces and k nearest neighbors (k nearest codewords), from the corresponding codebook, are computed and stored in a list. These n lists of k-NN consist of codewords with associated inverted indices. They are merged using MultiSequence algorithm [Babenko and Lempitsky, 2012] to generate final list, which comprises the vector similar to the query.

Let us consider, a high dimensional vector, $P_i \in R^M$, from the dataset, and split into two parts.

$$P_i = [p_1^i \quad p_2^i], \quad where \quad p_1^i \in R^{M/2}, p_2^i \in R^{M/2} \tag{1}$$

The two PQ codebooks consist of L codewords, denoted by U and V, for the inverted multi-indices are generated using k-means applying on each part of the vector.

$$U = \{u_1, u_2, u_3, ..., u_L\} \quad and \quad V = \{v_1, v_2, v_3, ..., v_L\} \tag{2}$$

For a given query, Q, is split into two parts, $Q = [q_1 \quad q_2]$. Now, for the each part of the query vector, q_1 and q_2, k nearest codewords, from the corresponding codebook U and V, are computed.

$$q^1 \in U = \{u_{\alpha(1)}, u_{\alpha(2)}, ..., u_{\alpha(k)}\} \quad and \quad q^2 \in V = \{v_{\beta(1)}, v_{\beta(2)}, ..., v_{\beta(k)}\} \tag{3}$$

Where $\alpha(k)$ and $\beta(k)$ are the k^{th} nearest codewords of q^1 and q^2 in codebook U and V correspondingly. The distances between q^1 and $u_{\alpha(k)}$ is denoted by $r(k)$ and q^2 and $u_{\beta(k)}$ is denoted by $s(k)$.

$$r(k) = d_1(q^1, u_{\alpha(k)}) \quad and \quad s(k) = d_2(q^2, v_{\beta(k)}) \tag{4}$$

The generated two distance sequences, $r(1), r(2)..., r(k)$ and $s(1), s(2)..., s(k)$, from q^1 and q^2 are paired using a MultiSequence algorithm to generate the final response to the query. The distances in each sequence is arranged in increasing order. In other words, $r(1)$ distance is lesser than $r(2)$ and so on. The MultiSequence algorithm is developed on priority queue of index pair of distances. While combining these two sequences, the sum of the distances of the codeword pairs are considered.

$$r(i) + s(j) = d(q, [u_{\alpha(i)} v_{\beta(j)}]) \tag{5}$$

Where, $i, j = 1, 2, ..., k$.

The lowest sum of the distances of the codeword pairs from the query is computed in the each subsequent step of the MultiSequence algorithm. Each codeword is associated with a list of

images. Then, these associated lists are added to the output list, which is the answer to the query (Q). Therefore, the inverted multi-indices data structure can achieve accurate and faster retrieval of the nearest neighbors for a given query.

Our proposal of fusion of inverted indices image search engine is developed on the inverted multi-indices concept. The proposal of the image search engine is presented in the upcoming sections.

3.3 Fusion of inverted indices to combine multiple descriptors

We propose a novel fusion method, called the fusion of inverted indices (FII), for efficiently combining multiple image descriptors for content-based image retrieval. The proposal is developed based on the concept of the inverted multi-indices approach, but amended in several ways. In the inverted multi-indices data structure as revisited in the Sec. 3.2, the vectors are decomposed into two equal sub-vectors. However, in the fusion of inverted indices, the query image is represented with several multi-dimensional image descriptors. This approach allows combining multiple image descriptors by integrating their responses to a query in finer subdivisions. The performed steps are:

1. A query image is represented by m image descriptors, leading to m descriptions of the content. The extracted descriptions are quantized separetly to generate a codebook for each descriptor. At the same time, corresponding inverted unique indices are genereted.

2. Then, m candidate lists of responses are built, where each list contains the k nearest codewords to the respective query, their respective distances and the set of associated images. The repeated image identifiers are not considered in the inverted indices structure. Thus, we call it inverted unique indices.

3. Since the distances from different lists are related to their descriptor space and characteristics, standard normalization is applied by using the maximum and minimum distances of the respective descriptor to them.

4. The candidate lists are combined through the multisequence algorithm, as proposed in [Babenko and Lempitsky, 2012], which returns final lists that consist of codewords and associated image ids sorted by their increasing distances from the query.

5. A voting algorithm is proposed to compute a frequency list that consists of image ids and associated frequencies according to their occurrences. All the frequency lists are summed up to generate a final frequency list that consists of the most similar images retrieved for the given query image.

The implementation details of the fusion of inverted indices (FII) image search engine are presented in the upcoming sections of this chapter. Before that, in a nutshell, we present the three different approaches to the similarity search, *i.e.,* inverted indices, inverted multi-indices and fusion of inverted indices, are illustrated in the Fig. 3.3.

As shown in the Fig. 3.3, for each approach, three multi-dimensional words (colored circles) are distributed in the descriptor space. An image content, rich in interest points, can overpopulate some clusters, therefore, those clusters (represented by green numbered squares) have a strong impact on that image representation. In order to perform the task of finding the 3-NN for a given query point (yellow star $\star$), the three strategies proceed as follows.

- Inverted files: Classical inverted file identifies the cluster to which the query belongs and retrieves all its associated descriptions inside (see Fig. 3.3(a)).
- Inverted multi-index: Inverted multi-indices subdivides the descriptor space into n subspaces ($n = 2$ here). Then the multi-sequence algorithm combines the nearest centroid to the query in each of the n subspaces, selecting the descriptions related to all the best combinations of subspace centroids. However, overpopulating descriptions from one image decreases the possibility of retrieving descriptions from other images with lower amounts of descriptions (see Fig. 3.3(b)).
- Fusion of inverted indices: With this approach, fusion of inverted indices, n descriptors are used to find images that match the query with more similar characteristics (here a 2D descriptor A and a 3D descriptor B). Each cluster in a descriptor space represents only the nearest descriptions from each matched image (descriptions in hooped dotted circles). Furthermore, as n subspace responses are combined, we are able to obtain a direct rank of the images that better match the query image. This rank represents the images that are similar in all or most of all descriptor characteristics (see Fig. 3.3(c)).

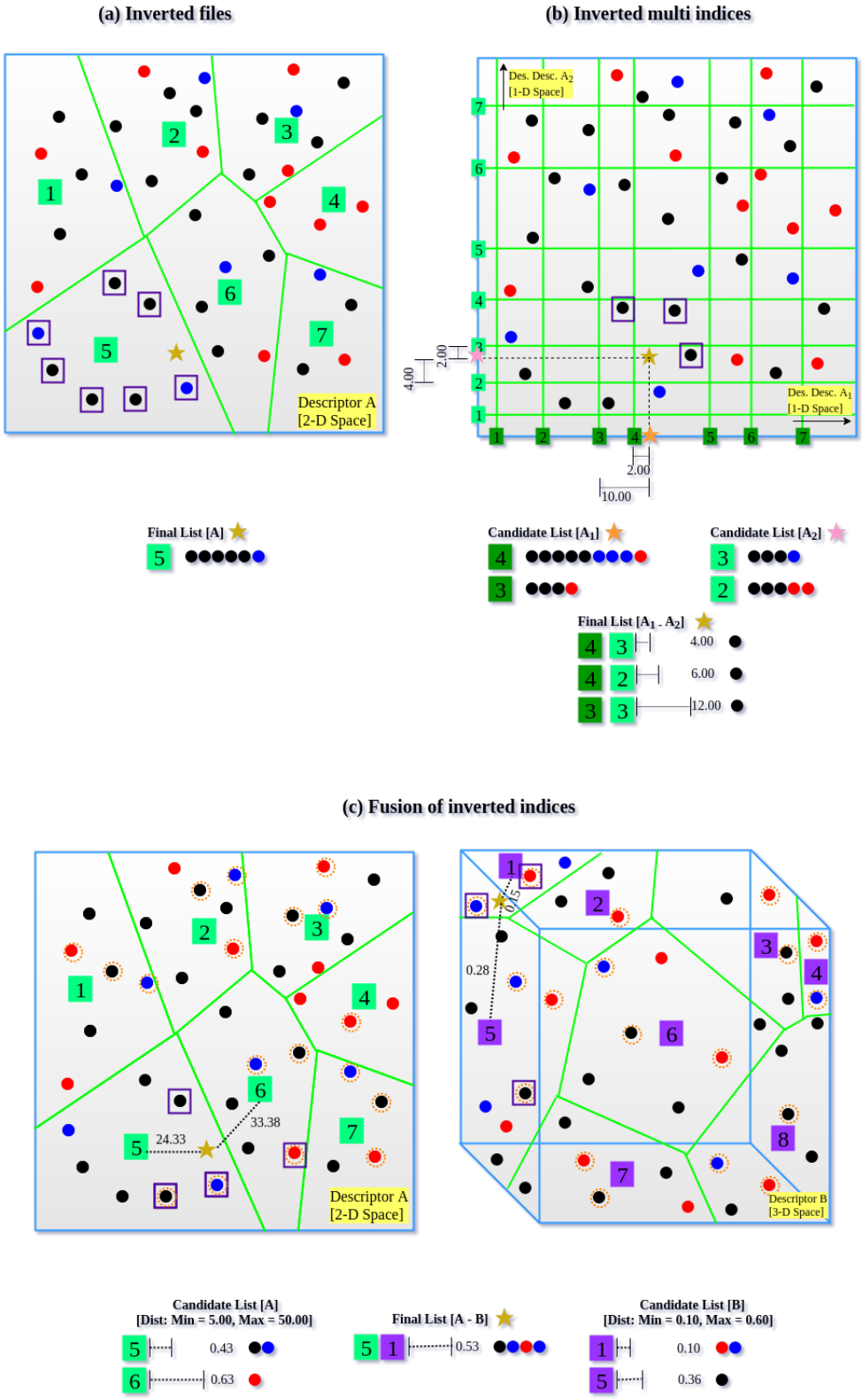


Figure 3.3: Illustration of three different strategies for similarity search: (a) Inverted files (b) Inverted multi indices (c) Fusion of inverted indices.

3.4 Fusion of inverted indices image search engine overview

The fusion of inverted indices (FII) image search engine is proposed for CBIR by fusing multple inverted indices as presented in the previsous Sec. 3.3. The overview of the FII image search engine is depicted in the Fig. 3.4.

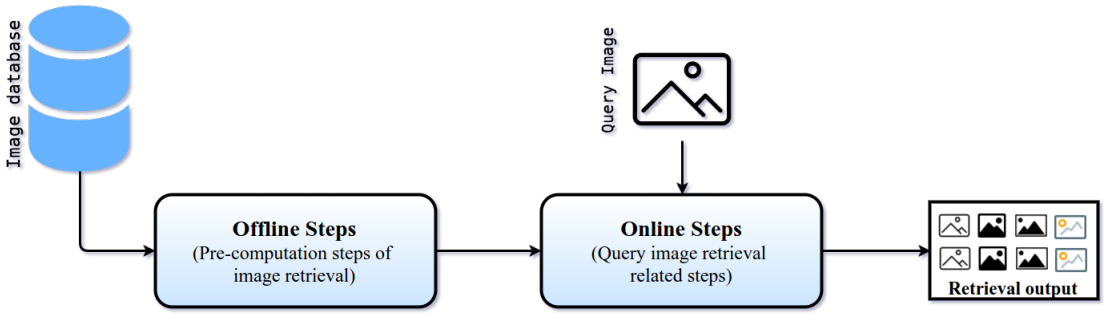


Figure 3.4: Overview of the proposed fusion of inverted index search engine for image retrieval.

FII search engine is decomposed into two main stages:

1. Offline stage and
2. Online stage

The offline stage consists of several steps in which the image dataset is processed and subsequent files are generated.

1. Let us consider m image descriptors are used to describe the image dataset.
2. All extracted descriptions from the image dataset and followed by an image identifier are combined together into a labelled data, $LD_1, \ldots, LD_m$, from their respective descriptors.
3. Therefore, m separate codebooks, $CB_1, \ldots, CB_m$ and corresponding inverted unique indices, $IUI_1, \ldots, IUI_m$, are generated from their respective descriptions using labelled data.
4. The generated codebooks and inverted unique indices in Step 2 are used in the Online stage, during image retrieval.

More details about the offline stage are explained in the Sec. 3.4.1.

The online stage deals with the retrieving of similar images to a query image (Q). The performed steps of the online stage are:

1. A query image is represented by m different descriptors, $Q_1 = \{q_1^1, q_2^2, \ldots, q_1^n\}, \ldots, Q_m = \{q_m^1, q_m^2, \ldots, q_m^n\}$.

2. k nearest codewords (from the previously generated codebooks, $CB_1, \ldots, CB_m$, in the offline stage) for each query description in $Q_1, \ldots, Q_m$ is obtained, leading to m k-NN lists ($kNNL$).

3. Then, m candidate lists, $CList_1 = \{cl_1^1, cl_1^2, \ldots, cl_1^n\}, \ldots, CList_m = \{cl_m^1, cl_m^2, \ldots, cl_m^n\}$, of responses are built, where each list contains the k nearest codewords to the respective query, their respective distances and the set of associated images (using previously generated inverted unique indices, $IUI_1, \ldots, IUI_m$, in the offline stage). The candidate list does not contain several references to the one image. In the building process, the first reference to an image (as it is associated to one of the k nearest codewords in the k-NN list) is considered.

4. Since the distances from different lists are related to their descriptor space and characteristics, standard normalization is applied by using the maximum and minimum distances from the dataset of the respective descriptor in the dataset to them.

5. The candidate lists are combined through the MultiSequence algorithm, which returns final lists, $FL = \{fl_1, \ldots, fl_n\}$, that consist of codewords and associated image ids sorted by their increasing distances from the query.

6. A voting algorithm is proposed to compute the frequency of the retrieved images. The voting algorithm generates weight list ($WL = \{wl_1, \ldots, wl_n\}$) that consists of image ids and weights based on the associated distances. The weight of the image is determined by subtracting the associated distance from 1. Finally, all the frequency lists are summed up to generate a final frequency list (FqL), depends on the corresponding weights and the frequency of the occurrences of the images, that consists of the most similar images retrieve for the query image.

The proposed FII approach can be categorized as intermediate fusion because of the candidate lists, related to closest words for each descriptor, are merged (and not the candidate lists of images, as with late fusion). More details of the online stage are illustrated in the Sec. 3.4.2.

3.4.1 Offline stage

The offline stage of FII search engine performs several steps related to precomputation of the image dataset and subsequent files generation such as feature extraction, codebook generation, etc. For sake of clarity, the presentation is restricted to two descriptors fusion, but can easily be generalized to any number of descriptor combination.

The overview of the FII offline stage is illustrated in the Fig. 3.5. Each step is detailed in the Secs. 3.4.1.1, 3.4.1.2 and 3.4.1.3.

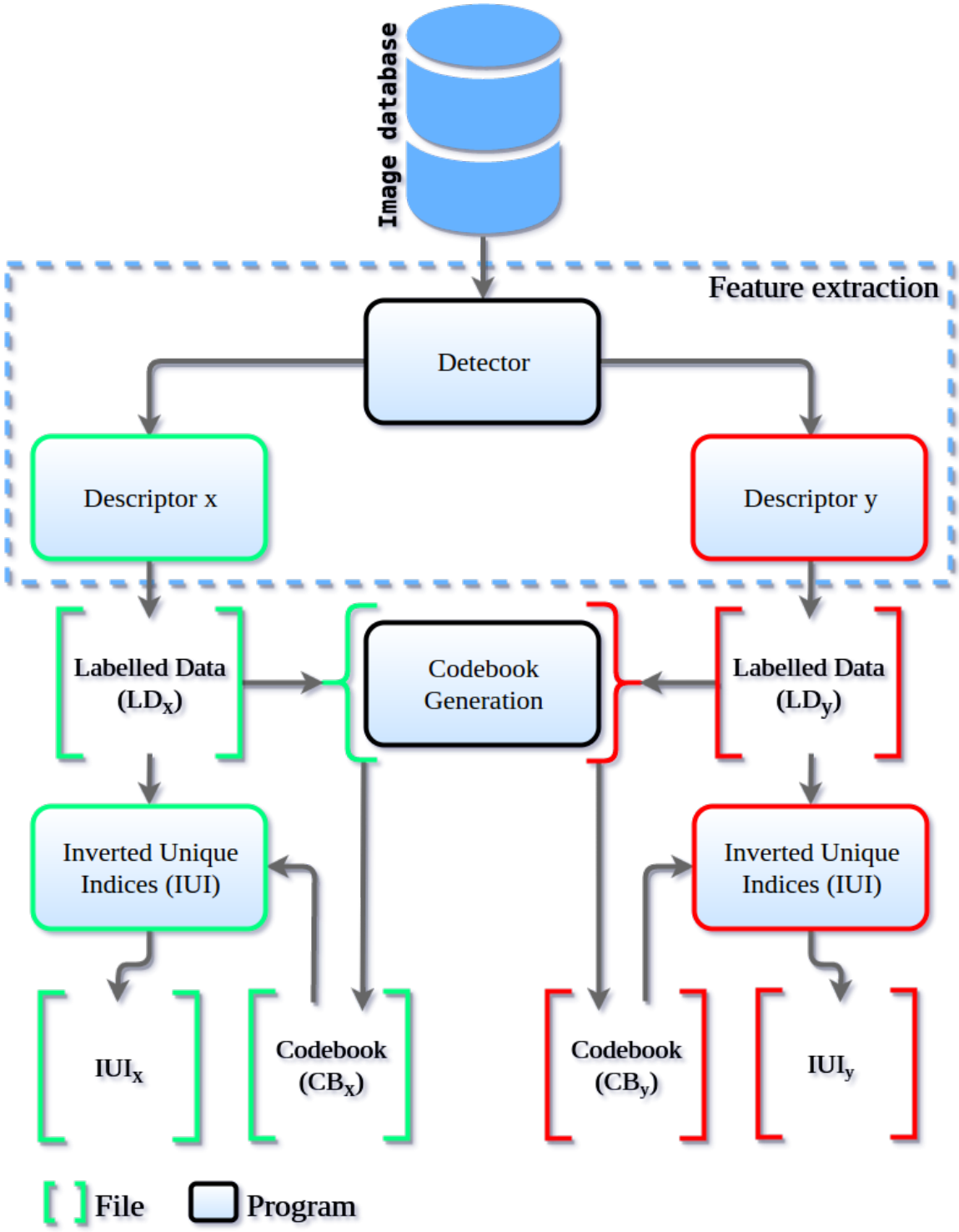


Figure 3.5: Offline steps of FII search engine.

3.4.1.1 Feature extraction

Feature extraction is the first step of the offline and as well as online stage in FII search engine. In this step, distinguished features are extracted from the database images.

The feature extraction step is divided into two sub-steps, *i.e.,* keypoint detection and description.

In the FII search engine, we mainly focus on the local feature detectors and descriptors to extract image features. The local descriptors first usually require defining a local support of extraction of the information in the images. Therefore, first, we detect the interest keypoints from the images using local detector. Then the extracted keypoints are described using several local descriptors.

Let us consider descriptor x and y are used for feature description. All the image description vectors are combined together to generate labelled data, LD_x and LD_y. The labelled data consists of all descriptions of a dataset. The each description is associated with an image identifier which is also included in the labelled data.

3.4.1.2 Codebook generation

The extracted visual feature vectors which are combined in a labelled data (LD_x and LD_y) from the image dataset are clustered into a vocabulary of visual words or codewords using clustering algorithm such as hierarchical k-means. A hierarchical clustering produces a sequence of clusters in which each cluster is nested into the next clustering in the sequence. It aims to find the best step at each clustering fusion in a greedy algorithm. Furthermore, hierarchical clustering is flexible enough in terms of applicability. Depending on the input data, any distance metric, such as Levenshtein, Jaccard, Euclidean, etc. can be used in hierarchical clustering. We used a fast implementation of hierarchical clustering. The codewords, which are the centres of the clusters, are the representation of the several similar image patches. Therefore, codebook (CB) size is the number of the clusters. In the codebook generation step, codebook CB_x and CB_y are generated from their respective descriptors. The file structures to generate codebook from the extracted features are presented below.

Name	`[I/O]`	Description
Labelled data (LD)	`[I]`	Store a matrix of size $n \times (d+1)$, where n is the number of the descriptors that are going to be used to train the codebook, d is the descriptor's dimension and additional image identifier of the descriptor to each row.
Codebook (CB)	`[O]`	Store a matrix of size $kc \times d$, where kc is the desired number of codewords (cluster centre) and d is the codeword's dimension.

3.4.1.3 Inverted unique indices

An inverted index is indexing structure to map the content such as words, to its location in a dataset in order to fast search of texts or images. When considering image features, inverted indexing is a mapping of codewords of their location in an image dataset. The proposed in-

verted unique indices ('IUI') is similar to a classical inverted index file, but instead of having a sequence of image ids associated to each codeword it has a set of image ids. Instead of having repeated image ids associated with each codeword, IUI considered only set of image ids and keeps repeated image ids only one time. The proposed inverted unique indices is compact as it represents the image ids related to a codeword avoiding redundancies.

In the step of inverted unique indices, IUI_x and IUI_y are generated from respective descriptor and codebook. The file structures and the algorithm to generate IUI are presented below.

Name	[I/O]	Description
Labelled data (LD)	[I]	Shown in Sec. 3.4.1.2.
Codebook (CB)	[I]	Shown in Sec. 3.4.1.2.
Inverted Unique Indices (IUI)	[O]	Store one row per codeword with its related set of image ids and associated distances.

Algorithms

Algorithm 2 – INVERTED UNIQUE INDICES

INPUT: Labelled Data (LD); Codebook (CB)
OUTPUT: Inverted Unique Indices (IUI)

1. *Declare* $InvertedUniqueIndices{:}IUI$
2. *Declare* $NN - Matrix \leftarrow NearestNeighbor\_Search(LB, CB)$
3. *For each* $q, cw, dist(q, cw)$ *in* $NN - Matrix$
4. $\quad$ *If* $q.imageId \notin IUI[cw]$
5. $\quad\quad$ *Push* $IUI[cw] \leftarrow q.imageId$
6. *Return* IUI

The inverted unique indices algorithm, shown above, is explained in the following description:

1. Line 2: The $NearestNeighbor\_Search(LD, CB)$ will find the nearest neigbor, from CB, for each descriptor, q, in LD. The information will be stored in the $NN - Matrix$.
2. Line 3: For each row in the $NN - Matrix$ we obtain the current descriptor id, and the related nearest codeword id and distance in between them. This is done n times to create the inverted unique indices for each codeword.
3. Lines 4-5: If the descriptor image id is not in the set of its corresponding codeword, image id is pushed in.
4. Line 6: Return the *Inverted Unique Indices*.

To illustrate the inverted unique indices generation steps, An example is given in Appendix B.1.

3.4.2 Online stage

The online stage of FII search engine deals with the retrieving of similar images to a query image (Q). The overview of the online stage is depicted in the Fig. 3.6. For the explanation purpose, the image description is restricted to two descriptors. Let us consider a query image space (Q) is represented by the descriptors, x and y, as used in the Sec. 3.4.1.1. The proposed FII merges Q_x and Q_y image description by combinining their responses. This is achieved by performing several steps which are explained in the following Secs. 3.4.2.1, 3.4.2.2 and 3.4.2.3 and 3.4.2.4.

3.4.2.1 Search k-Nearest neighbor

The nearest neighbors search concerns the retrieval of the k similar codewords in CB_x and CB_y to the each query point in Q_x and Q_y respectively. It generates k-NN lists, $kNNL_x$ and $kNNL_y$, consisting k nearest codewords and associated distances. Since the distances from different lists are related to their descriptor spaces, standard normalization is applied by using the maximum and minimum distances of the respective descriptor to them as presented in the Eq. 6.

$$dist(q, cw) = \frac{dist(q, cw) - dist_{min}}{dist_{max} - dist_{min}} \tag{6}$$

The k-NN search algorithm is presented below.

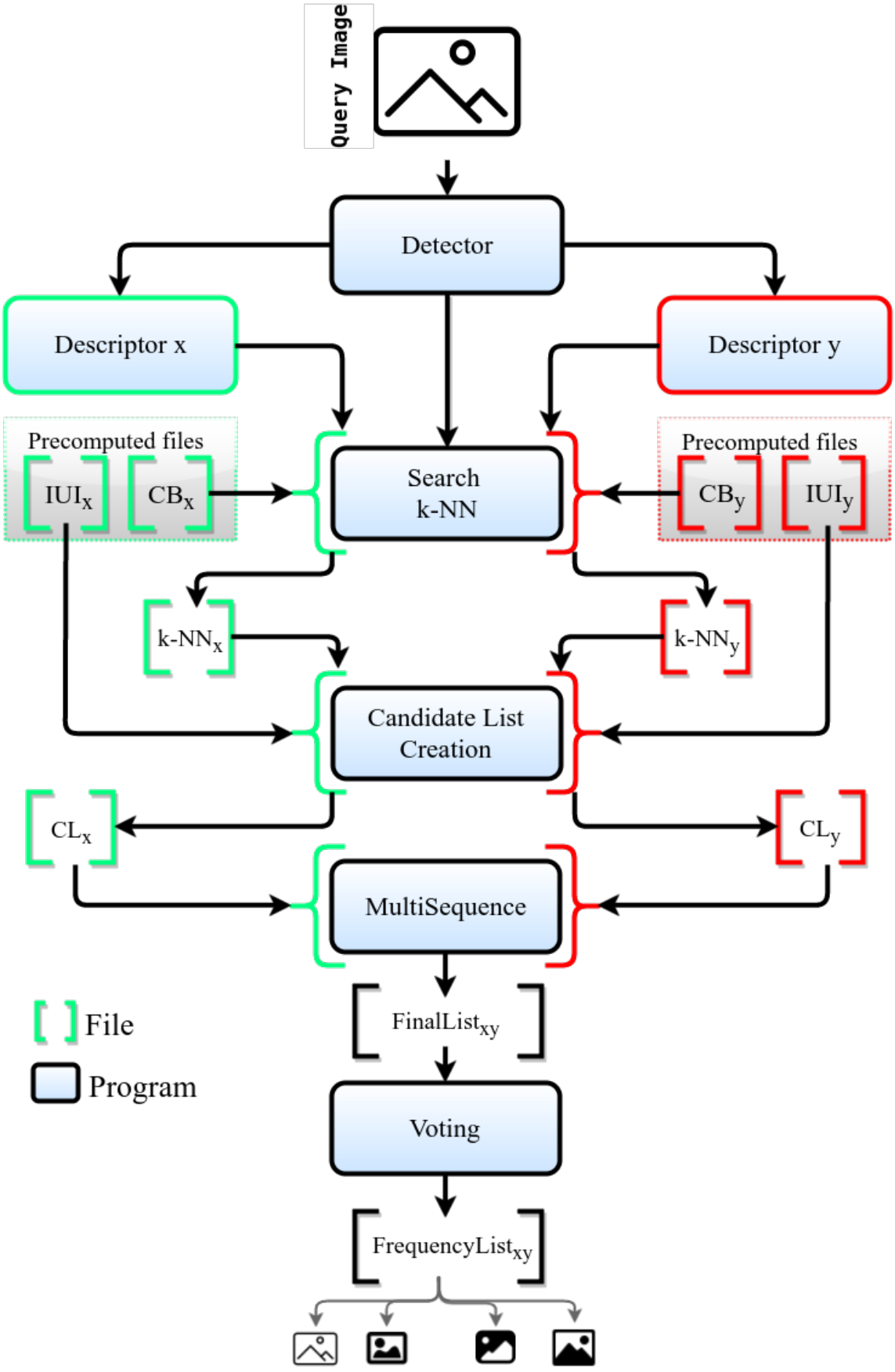


Figure 3.6: Online steps of fusion of inverted indices (FII) search engine.

Name	[I/O]	Description
Query point (q)	[I]	Query point of the descriptors.
Codebook (CB)	[I]	Explained in Sec. 3.4.1.2
k-NN List (kNNL)	[O]	Stores k references to the codeword and the distance achieve between it and the q.

Algorithms

Algorithm 3 – SEARCH k-NN

INPUT: Query point (q); Codebook (CB); DistMax ($dist_{max}$); Number of nearest codewords (k)
OUTPUT: KNN lists($kNNL$)

1. *Declare* $KNN - Lists{:}kNNL$
2. $kNNL \leftarrow KNN - SEARCH(q, CB)$
3. *For each* $dist$ *in* $kNNL$
4. $\quad\llcorner\ dist \leftarrow dist/dist_{max}$
5. *Return* $kNNL$

The steps of search k-NN algorithm are explained below.

1. Line 2: Computes the k nearest codewords for each query point, q, and stores them with their distances to q.
2. Lines 3 to 4: Normalizing the distances for every *kNNL* that belongs to each query.
3. **Line 5**: Return the *kNNL*.

To illustrate more about the k nearest neighbor search process, an example is given in the Appendix B.2.

3.4.2.2 Candidate list creation

In the candidate list creation step, n candidate lists ($CList$) of responses of descriptors x and y are built from k-NN list and corresponding inverted unique indices (see Sec. 3.4.1.3). Each $CList$ contains k nearest codewords to the respective query, their respective distances and the set of associated T images from inverted unique indices, IUI_x and IUI_y. The repeated image ids are not included in the candidate list. During the candidate list creation process, it only includes the first reference to an image and it consists of an approximate T number of image ids.

The details of the candidate list creation algorithm are explained below.

Name	[I/O]	Description
Inverted Unique Indices (IUI)	[I]	Explained in Sec. 3.4.1.3.
Knn List (kNNL)	[I]	Explained in Sec. 3.4.2.1.
Candidate List (CList)	[O]	Stores the merge information of a k-NN list and the respecivive inverted unique indices to its codeword, without repeating image ids.

Algorithms

Algorithm 4 – CANDIDATE LIST CREATION

INPUT: Knn List ($kNNL$); Inverted Unique Indices (IUI); maximum number of images (T)
OUTPUT: Candidate list ($CList$)

1. *Declare* $CandidateList{:}CList$
2. *Declare* $set{:}imageIds$
3. *For each* $i, cw, dist$ *in* $kNNL$
4. $\quad CList[i].id \leftarrow cw$
5. $\quad CList[i].dist \leftarrow dist$
6. $\quad cw\_imageIds \leftarrow getImgIds(IUI, cw)$
7. $\quad$ *For each* $imageId$ *in* $cw\_imageIds$
8. $\quad\quad$ *If* $imageId \notin imageIds$
9. $\quad\quad\quad$ *Push in* $CList[i].imageIds \leftarrow cw\_imageIds$
10. $\quad$ *If* $size(imageIds) \geqslant T$
11. $\quad\quad$ *Break*
12. *Return* $CList$

Following steps are performed during candidate list creation:

1. Lines 1 to 2: Declaring candidate list and a set image ids. This set, guarantees that the candidate list will not introduce repeated image ids.

2. Line 3: Obtains the current index, i, the corresponding codeword, cw, and distance, $dist$, from the k-NN list. This is done, for lines 4-11, until the quantity of unique image ids are T or a little greater.

3. Lines 4 to 5: The i-th element in the candidate list will assume the information of the current codeword.

4. Lines 7 to 9: We analize if the inverted unique indices corresponding to the current cw are already in the set, if they are then they will not be added image ids of the current element in $CList$.

5. Lines 10 to 11: If the number of image ids in the set is equal or greater than T, the loop of lines 4 to 11 is interrupted.

6. Line 11: Returning the Candidate List.

In the Appendix B.3, an example of candidate list creation is given.

3.4.2.3 MultiSequence algorithm

In the MultiSequence step, multiple responses of image descriptions, x and y, are combined together. The candidate lists, which is explained in Sec. 3.4.2.2, are joined through the multiSequence algorithm. It generates n final lists, FL. Each FL consists of codeword pair and associated images sorted by their increasing distances from the query. MultiSequence step consists of two algorithms depending on the number of candidate lists to be joined. MultiSequence pair algorithm is applied if only two candidate lists need to be joined. Otherwise, when the number of lists is more than two, we introduce divide and conquer strategy to split the group into two halves and so on. These two algorithms are explained below.

Name	`[I/O]`	Description
Candidate List (CList)	`[I]`	Explained in Sec. 3.4.2.2.
Final List (FL)	`[O]`	Stores combined codeword pair, associated image ids and distances.

Algorithms

Algorithm 5 – MULTISEQUENCE

INPUT: Vector of candidate lists ($CLists$); number of images to be retrieved (T); first ($first$) and last ($last$) positions in $CLists$ to analize

OUTPUT: Final list (FL)

1. *If* $last - first > 1$, *then*
2. $\quad half \leftarrow \lfloor \frac{first+last}{2} \rfloor$
3. $\quad CList_u \leftarrow MultiSequence(CLists, first, half, T)$
4. $\quad CList_v \leftarrow MultiSequence(CLists, half + 1, last, T)$
5. $\quad$ *Return* $MultiSequence(CList_u, CList_v, T)$
6. *Else if* $last - first = 1$, *then*
7. $\quad$ *Return* $MultiSequencePair(CLists[first], CLists[last], T)$
8. *Else*

9. └ *Return* $CLists[first]$

The MultiSequence algorithm, shown above, selects the best sequence between two or more candidate lists, by performing the following steps:

1. Lines 1 to 5: If there are more than two candidate list to join, the group will be splitted in two new lists where both lists will have the same size or the second one will be greater by at most one element.
2. Lines 6 to 7: If there are two lists to join the MultiSequence pair algorithm is called, in order to join them.
3. Lines 8 to 9: If there is just one list, it will be returned (in case of having an odd number candidate lists).

Algorithm 6 – MULTI SEQUENCE PAIR

INPUT: First candidate list ($CList_u$); second candidate list ($CList_v$); and, number of images to be retrieved (T)
OUTPUT: Final list (FL)

1. *Declare the* $finallist$*:* $FList$
2. *Declare a* $priority\ queue$ *of distances:* $Dists$
3. *Declare an empty set of images id's:* $ImgIds$
4. *Declare a vector of indices accesed* $LastIdx\_v$
5. *Push in* $Dists \leftarrow$ *Pair*$(1, 1, CList_u[1] + CList_v[1])$
6. *While (Size of* $ImgIds < T$*) and (*$Dists$ *is not* $\varnothing$*)*
7. $\quad$ $Pair(u, v) \leftarrow$ *Pop from* $Dists$
8. $\quad$ $LastIdx\_v[u] \leftarrow v$
9. $\quad$ $FList \leftarrow Element(CList_u[u], CList_v[v])$
10. $\quad$ $ImgIds \leftarrow getImgIds(Element)$
11. $\quad$ *If (*$u \leqslant$ *Size of* $CList_u$*) and (*$v = 1$ *or* $LastIdx\_v[u+1] = v - 1$*)*
12. $\quad\quad$ *Push in* $Dists \leftarrow$ *Pair*$(u + 1, v, CList_u[u].dist + CList_v[v].dist)$
13. $\quad$ *If (*$v \leqslant$ *Size of* $CList_v$*) and (*$u = 1$ *or* $LastIdx\_v[u-1] \geqslant v + 1$*)*
14. $\quad\quad$ *Push in* $Dists \leftarrow$ *Pair*$(u, v + 1, CList_u[u].dist + CList_v[v].dist)$
15. *Return* FL

The MultiSequence Pair algorithm explained above, selects the best sequence between two candidate lists, by performing the following steps. In order to understand the algorithm, we must think about a virtual matrix where columns and rows indices are indicated by u and v respectively.

1. Lines 1 to 4: Declaring structures. *LastIdx_v* must have a number of slots equal to *CList*$_u$ size and every slot must be initialized with 0.
2. Lines 5: Push into *Dists* a pair indicating the first element of each list, in order to start the multi sequence search.
3. Lines 6: Execute lines 7 to 13 until T is reached, or there are no more pairs to get more images ids.
4. Lines 7: Pop from *Dists* the pair of elements which sum the shortest distance.
5. Lines 8: Both coordinates of the last pair selected are updated in *LastIdx_v*, by indicating that for the u-th column we have accesed the cell thqt also belongs to the v-th row.
6. Lines 9: Insert the new element to the *Final List*.
7. Lines 10: Insert to the *ImgIds* set the new element list of image ids.
8. Lines 11: Evaluates if the pair in the next column of the current pair column can be consider as a candidate.
   - First, there must exist a next column to consider the execution of line 12.
   - Second, if the current v value is not referencing to the first row, we must be sure that the pair in $(u,v-1)$ has already been considered in the *Final List* as it has a shorter distance.
9. **Lines 12**: Push to *Dists* the pair of elements in the next column of the virtual combination pairs matrix.
10. **Lines 13**: Evaluates if the pair in the next increasing row of the current pair row can be consider as a candidate.
    - First, there must exist a next row to consider the execution of line 14.
    - Second, if the current u value is not referencing the first column, we must be sure that the pair in $(u-1,v+1)$ has already been considered in the *Final List* as it has a shorter distance.[1]
11. **Lines 14**: Push in *Dists* the pair of elements in the next row of the virtual combination pairs matrix.
12. **Lines 15**: Returns the *Final List*.

To give more insight about MultiSequence pair algorithm, an illustration is given in Appendix B.4.

[1]Note that line 11 and 13 are just evaluating that to consider a pair (u,v) as candidate, the precedent pairs $(u-1,v)$ and $(u,v-1)$ must have already been inserted into the *Final List*, as they have shorter distances.

3.4.2.4 Voting algorithm

A voting algorithm is proposed to compute the frequency of the retrieved images from the final lists generated by the MultiSequence steps in the Sec. 3.4.2.4. The voting algorithm generates a frequency list that consists of image ids and associated frequencies according to their occurrences in the final lists (FL). The frequency of an image depends on the associated image weight. The image weight is determined on the distance associated with the each image FL. All the frequency lists are summed up to generate a final frequency list (FqL) depends on the associated weight and the frequency of the occurrences of the images. The FqL consists of the most similar images ids retrieved in the sorted sequence of decreasing similarities to the query image.

The proposed voting strategy is explained below.

Name	`[I/O]`	Description
Final List (FL)	`[I]`	Explained in Sec. 3.4.2.3.
Frequency List (FqL)	`[O]`	Stores list of image ids and the corresponding weight values to the image query.

Algorithms

Algorithm 7 – VOTING

INPUT: Final list (FL)
OUTPUT: Frequency List (FqL)

1. *Declare a vector of* $weightlist$*:* WL
2. *For each* fl *in* FL
3. *Declare* $weightlist$*:* wl
4. *For each* e_{fl} *in* fl
5. *Declare* $element$*:*e_{wl}
6. $e_{wl}.id \leftarrow e_{fl}.id$
7. $e_{wl}.imgids \leftarrow getImgIds(e_{fl})$
8. $e_{wl}.weight \leftarrow 1 - e_{fl}.dist$
9. *Push in* $wl \leftarrow e_{wl}$
10. *Push in* $WL \leftarrow wl$
11. *Declare* $frequencylist$*:* FqL
12. *For each* wl *in* WL
13. *For each* $imgId$ *in* $getImgIds(wl.e)$
14. $FqL[imgId] \leftarrow FqL[imgId] + wl.e.weight$
15. *For each* $imgId$ *in* FqL
16. $FqL[imgId] \leftarrow FqL[imgId]/n$
17. *Return sorted* FqL

The steps of the voting algorithm are presented in the following:

1. Line 1: Declaring a vector of weight lists.
2. Lines 2-10: Fill the vector of weight lists, with the computed weight lists. Where one element's weight is just 1 minus its own distance.
3. Lines 11: Declare a hash table to store the frequency list, this will acelerate the access for updating each image frequency (in function of the sum of the weights).
4. Lines 12-14: Pop from *Dists* the pair of elements which sum the shortest distance.
5. Lines 15-16: Normalizing the frequencies.
6. Lines 17: Return the sorted frequency list (from the most to the least similar image to the query).

A voting algorithm example is given in the Appendix B.5.

3.5 Experiments and evaluations

This section presents and discusses the experiments conducted to evaluate our contributions. It consists of several subsections by starting with image dataset descriptions and images retrieval performance measurement criteria in the Sec. 3.5.1. Section 3.5.2 presents the parameter configurations used in the experiments followed by the evaluations of CBIR using two state of the art image retrieval strategies. In the Sec. 3.5.3, the proposed FII image retrieval strategy is evaluated against two public benchmarks with different parameter configurations and also it is compared with other two state of the art CBIR strategies. The complexity of the proposed FII image retrieval strategy with other image retrieval strategies is presented in the Sec. 3.5.4. Dimension reduction of the image descriptors is useful for image retrieval. Therefore, the next sets of experiments are conducted with the reduced dimensions of the descriptors using different dimension reduction strategies in the Sec. 3.5.5. The same Sec. 3.5.5 also explains the advantages of using reduced dimensions in the image retrieval. After completion of these initial sets of experiments, we expand our experiments using other parameter configurations, such as different combinations of descriptors and varying k nearest neighbor, in the Sec. 3.5.6. We further evaluate the FII image retrieval strategy against two additional public benchmarks in the Sec. 3.5.7, followed by image retrieval examples from different datasets in the Sec. 3.5.8.

3.5.1 Framework of evaluation

The initial set of experiments are conducted on two image datasets with different sizes and contents:

- COIL_DB: this dataset contains 600 synthesized images containing 100 objects with different orientations and viewpoints, from the well-known benchmark *COIL-100*[2], synthetically inserted on photographs as background (images with heterogeneous and complex contents). Examples are shown in first row of Fig. 3.7.
- Paris_DB: it is a public benchmark[3] consisting of 6412 images collected from Flickr by searching for 12 particular Paris landmarks; see examples in second row of Fig. 3.7.

Figure 3.7: Samples from the benchmarks used in our experiments: 1^{st} row for COIL_DB and 2^{nd} row for Paris_DB.

The image descriptors employed in our experiments are local descriptors, suitable to retrieve objects in such datasets with cluttered contents. The interest keypoints are extracted using Hesaff Affine (hesaff) [Mikolajczyk and Schmid, 2004] detector which provides better performance compared to others. Therefore it is used by default in the rest of the experiments in this chapter. However, several other detectors are also considered for the image retrieval experiments and those experiments are presented in the later chapters. The extracted keypoints are described by several local descriptors. Among the descriptors tested, we concentrate mainly on Scale-invariant feature transform (SIFT) [Lowe, 2004], Speeded up robust features (SURF) [Bay et al., 2008], and Shape Contexts (SC) [Belongie et al., 2002], which performed better individually for these datasets. Therefore, we use these three descriptors and their combinations for the experiments. Later in this section, we consider other descriptors for image retrieval. The comparison results between the different combinations of the descriptors justify the selection of the descriptor combination.

Performances are presented with mean Average Precision (mAP) and precision-recall curve. In the context of information retrieval, precision and recall are defined in terms of a set of retrieved images and a set of relevant images. Precision is a measure of relevant images in the retrieved images, while recall measures how truly relevant results are returned.

The precision (Pr) and recall (Re) can be measured as:

$$Pr = \frac{T_p}{T_p + F_p} \tag{7}$$

[2]http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php.
[3]http://www.robots.ox.ac.uk/∼vgg/data/parisbuildings/index.html.

$$Re = \frac{T_p}{T_p + F_n} \tag{8}$$

where T_p is true positives, F_p is false positives and F_n is false negatives. A high area under the precision-recall curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate.

The mAP is a summarized measure of quality across all the queries by averaging average precision, in the range of 0 to 1. The mAP is can compute as presented in the Eq. 9.

$$mAP = \frac{1}{Q} \sum_{q_i \in Q} AP(q_i) = \frac{1}{Q} \sum_{q_i \in Q} (\frac{1}{n_r} \sum_{n=1}^{n_r} Pr_{q_i}(Re_n)) \tag{9}$$

where $AP(q_i)$, *i.e.,* average precision, measures the mean over the precision (Pr_{q_i}) after each relevant retrieval at n^{th} recall (Re_n), and n_r is the number of relevant retrieval for i^{th} query.

3.5.2 Parameter settings and baseline

The two main parameters of the proposed fusion of inverted index approach are the codebook size and parameter k of the individual k nearest neighbor search. The codebook size is varied in the range of {25000,125000} for COIL_DB and in range {500000,2000000} for Paris_DB. The best accuracy is obtained with 50000 words for COIL_DB and 1500000 words for Paris_DB, for the three used descriptors. Similarly, parameter k was varied from 1-NN to 5-NN, where 2-NN achieved the best results for all the descriptors and datasets. These parameters are used by default in the rest of the experiments.

To give the first insight into the proposal, we begin by evaluating the original version of the inverted multi-index [Babenko and Lempitsky, 2012] completed with our strategy of image voting (see Sec. 3.4.2.4) on a single descriptor, facing the classical approach based on Bag-of-Features (BoF) with tf-idf scores [Sivic and Zisserman, 2003]. Table 3.1 shows the mAP achieved on the three descriptors used individually for both datasets.

Dataset	Descriptor	BoF with tf-idf [Sivic and Zisserman, 2003]	Inverted multi-index [Babenko and Lempitsky, 2012]
COIL_DB	SIFT	0.103	0.552
	SURF	0.098	0.446
	SC	0.102	0.435
Paris_DB	SIFT	0.095	0.437
	SURF	0.084	0.481
	SC	0.082	0.394

Table 3.1: mAP results for individual descriptors.

An approach based on BoF with tf-idf achieves a lower precision since this representation relies on a global description of the image based on the frequency of presence of words in the image

associated with a global similarity measure (usually the Euclidean distance), less robust to the complex scenes present in the two datasets. Not surprisingly, the voting strategy with the inverted multi-index approach performs better since it searches more locally for similar areas in different images by comparing groups of words.

3.5.3 Fusion of different descriptions with FII

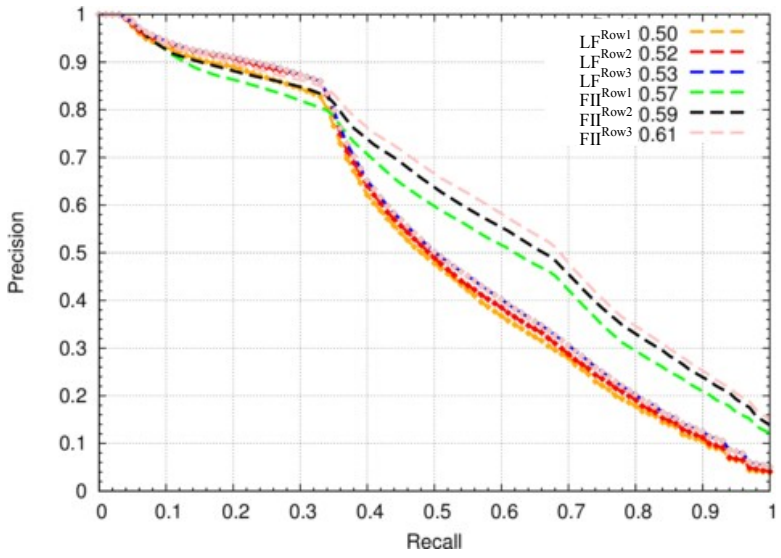
This section presents the results of the fusion obtained with joint use of different descriptions, in reference to Sec. 3.4. Different combinations of SIFT, SURF and SC descriptors are evaluated using the proposed FII image search engine. We also compare FII proposal to two other state-of-the-art descriptor fusion approaches: feature concatenation (CBoF) [Yu et al., 2013] based on BoF with tf-idf and the best late fusion technique (LF) [Neshov, 2013] based on combining multiple ranked outputs to produce a final ranked list. The detailed of these two fusion strategies are presented in the Sec. 2.4.1 and Sec. 2.4.2 of Chapter 2. Table 3.2 shows the performance obtained for the two datasets. The highlighted table cells represent the best achieved results.

Dataset	Descriptors combination	CBoF [Yu et al., 2013]	LF [Neshov, 2013]	FII
COIL_DB	SIFT-SURF	0.103	0.505	0.566
	SIFT-SC	0.104	0.524	0.594
	SIFT-SURF-SC	0.114	0.537	0.612
Paris_DB	SIFT-SURF	0.102	0.531	0.545
	SURF-SC	0.098	0.495	0.529
	SIFT-SURF-SC	0.112	0.532	0.546

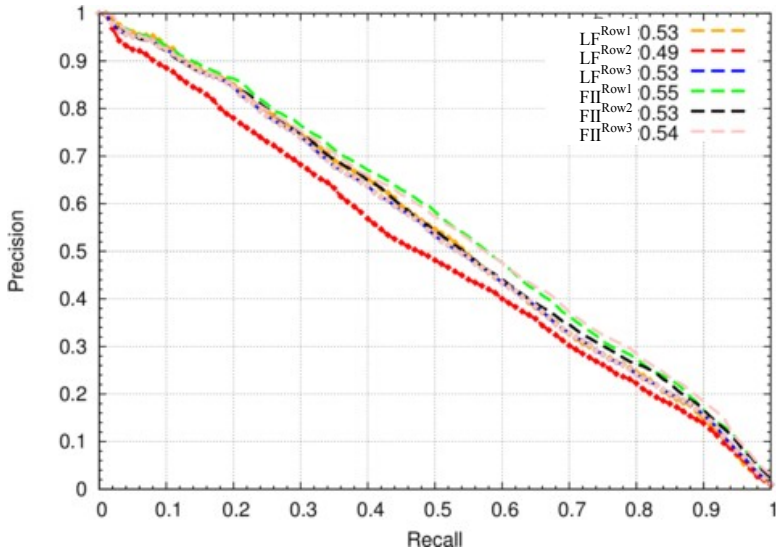
Table 3.2: Comparison of mAP obtained with the fusion of descriptors using different descriptor fusion strategies.

Due to the poor scores obtained with the classical approach BoF with tf-idf on individual descriptors (see Table 3.1), the results obtained here with approach CBoF are low again, even if the fusion improves them slightly. The LF method performs better for the two datasets, but it is not able to outperform the FII method, whatever the combination. This is due to the fact that the former method only considers the associated neighbors to the nearest word to the query, while the FII considers several combinations of word neighbors. Also that the fused descriptors represent two complementarity subspaces which estimate better the approximation of nearest neighbors. We also observe that the best configurations of descriptors are not exactly the same for the two datasets: it is SIFT-SURF-SC for COIL_DB and SIFT-SURF and SIFT-SURF-SC performed equally well for Paris_DB.

The precision-recall curves of the LF and FII approaches are presented in the Fig. 3.8 for COIL_DB and Paris_DB respectively. In the Fig. 3.8(a), the LF^{Row1} indicates the precision-



(a)



(b)

Figure 3.8: Precision-recall curves for different descriptor fusion: (a) COIL_DB (b) Paris_DB.

recall obtained with SIFT-SURF combination for COIL_DB as presented in the Table 3.2. Similarly, legends with FII^{Row} represent the precision-recall achieved with FII approach.

3.5.4 Image retrieval complexity

In this section, the relative efficiency of the different descriptor fusion strategies, *i.e.,* feature concatenation (CBoF) [Yu et al., 2013], late fusion technique (LF) [Neshov, 2013] and our proposed FII method, are presented. Let us consider, images are described by three different descriptors of dimensions n_1, n_2 and n_3. The same codebook size ($= L$) is used.

The CBoF early fusion builds a high dimensional space by concatenating several descriptors. The concatenated vector dimension is $(n_1 + n_2 + n_3)$. Due to the vector concatenation, the

memory footprint is quite high. However, for a given query, the matching step is performed only once. Therefore, the overall image retrieval time is less compared other two strategies. LF strategy [Neshov, 2013], to match a query with the codebooks, performs three times considering three descriptors to combine. Therefore, LF computes L (L = size of codebook) distances between n_1, n_2 and n_3 dimensional vectors separately. This generates three separate response lists to the query. Finally, these lists are combined based on score or ranking of the retrieved images to produce the final retrieved list. For FII strategy, the number of operations is same as LF strategy during matching a query with the codebook. However, the FII strategy incurs additional computation cost during MultiSequence steps. This additional computational time is related to the number of fused descriptors, as the complexity of the MultiSequence algorithm is $n \log n$, where n denotes the number of descriptors.

Table 3.3 presents the average image retrieval time obtained for the different combination of descriptors with three fusion strategies.

Dataset	Descriptors combination	Averaged time (Sec)		
		CBoF [Yu et al., 2013]	LF [Neshov, 2013]	FII
COIL_DB	SIFT-SURF	0.031	0.043	0.052
	SIFT-SC	0.024	0.035	0.041
	SIFT-SURF-SC	0.066	0.081	0.094
Paris_DB	SIFT-SURF	0.825	1.024	1.219
	SURF-SC	0.731	0.882	0.905
	SIFT-SURF-SC	1.016	1.985	3.738

Table 3.3: Comparison of image retrieval time (in second) obtained with the fusion of descriptors using different descriptors fusion strategies.

The average retrieval time for CBoF is less compared to other two strategies, but the retrieval accuracy (mAP) is not high (see Table 3.2 in Sec. 3.5.3). Although the LF strategy has better retrieval time compared to FII approach, the image retrieval accuracy can not beat our FII strategy. The FII approach produces better retrieval accuracy but with extra computational expenses compared to LF fusion.

3.5.5 Feature dimensionality reduction and their fusion

In this section, we explore the potential of the reduction of the dimensions of the descriptors in the proposed image retrieval strategy. The joint use of different descriptors naturally leads to increase the volume of manipulated features, and then to slow down the computational processes. This drawback can be addressed by exploiting dimensionality reduction techniques, which decrease each multidimensional description dimension to its half or fourth part while

maintaining a good degree of accuracy, sometimes similar even higher to the one of the original description. In addition, each dimensionality reduction technique may bring some particular advantage: principal component analysis (PCA) [Valenzuela et al., 2012] is able to remove noise from the descriptions, while partial least squares (PLS) [Rosipal and Krämer, 2006] can add distinctiveness as it takes into account class correlation. In addition, used in the FII framework, it can improve the results, similarly as approach [Babenko and Lempitsky, 2012] which is based on a simple decomposition into subspaces.

Consequently, we propose to use them to decompose the multidimensional description into smaller subspaces, instead of simply splitting it into several parts as in [Babenko and Lempitsky, 2012]. Indeed, we think that this alternative may conduce to establish finer subdivisions and then to determine nearest neighbors better, in addition, to reduce the amount of features to manipulate.

Dataset	Descriptor	Reduction		Fusion
COIL_DB	$SIFT^{128}$	$SIFT^{PCA32}$	$SIFT^{PLS32}$	$SIFT^{PCA32}_{PLS32}$
	0.552	0.466	0.450	0.582
		$SIFT^{PCA64}$	$SIFT^{PLS64}$	$SIFT^{PCA64}_{PLS64}$
		0.475	0.459	0.589
	$SURF^{64}$	$SURF^{PCA20}$	$SURF^{PLS20}$	$SURF^{PCA20}_{PLS20}$
	0.443	0.355	0.331	0.432
		$SURF^{PCA32}$	$SURF^{PLS32}$	$SURF^{PCA32}_{PLS32}$
		0.369	0.362	0.452
	SC^{36}	SC^{PCA12}	SC^{PLS12}	SC^{PCA12}_{PLS12}
	0.531	0.403	0.378	0.501
		SC^{PCA18}	SC^{PLS18}	SC^{PCA18}_{PLS18}
		0.442	0.420	0.539

Table 3.4: mAP results with reduced descriptions and their fusion for COIL_DB.

In the first set of experiments, the descriptors are used individually and each of them is decomposed with PCA and PLS dimensionality reduction techniques. On the two datasets, Tables 3.4 and 3.5 show the mAP obtained (i) before any reduction, with the simple splitting strategy of inverted multi-indices of [Babenko and Lempitsky, 2012] (column 'Descriptor'), (ii) after reduction, again with the simple splitting strategy (column 'Reduction') and (iii) with the fusion of two reduced descriptions (column 'Fusion'). The sub-index and super-index texts next to each descriptor indicate their original dimension or their reduced dimension preceded by the technique used. For example, $SURF^{PLS32}$ indicates that description SURF was reduced with PLS down to 32 dimensions, and SC^{PCA12}_{PLS12} that description SC was reduced both with PCA and PLS down to 12 dimensions, before fusing them as two separate descriptors in inverted multi-index. For each descriptor, in the column 'Fusion', we experiment a combination associated to a dimensionality lower than the original one (*i.e.,* $SIFT^{PCA32}_{PLS32}$) and a combination with equal

Dataset	Descriptor	Reduction		Fusion
Paris_DB	$SIFT^{128}$	$SIFT^{PCA32}$	$SIFT^{PLS32}$	$SIFT^{PCA32}_{PLS32}$
	0.431	0.446	0.481	0.488
		$SIFT^{PCA64}$	$SIFT^{PLS64}$	$SIFT^{PCA64}_{PLS64}$
		0.481	0.424	0.485
	$SURF^{64}$	$SURF^{PCA20}$	$SURF^{PLS20}$	$SURF^{PCA20}_{PLS20}$
	0.483	0.453	0.443	0.494
		$SURF^{PCA32}$	$SURF^{PLS32}$	$SURF^{PCA32}_{PLS32}$
		0.486	0.469	0.517
	SC^{36}	SC^{PCA12}	SC^{PLS12}	SC^{PCA12}_{PLS12}
	0.392	0.312	0.277	0.334
		SC^{PCA18}	SC^{PLS18}	SC^{PCA18}_{PLS18}
		0.386	0.370	0.418

Table 3.5: mAP results with reduced descriptions and their fusion for Paris_DB.

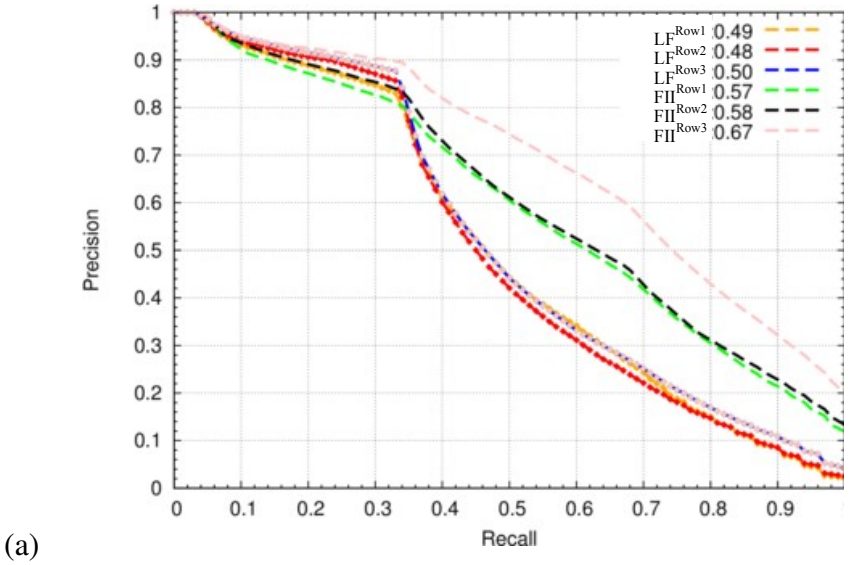
dimensionality (*i.e.,* $SIFT^{PCA64}_{PLS64}$), knowing that the dimensionality of SIFT is 128.

Irrespective of the dataset, the loss in precision is not proportional to the percentage of dimension reduction applied for single reduction; it achieves slightly lower or similar precision than with the original description. The column 'Fusion' shows that the fused reduced descriptions of the original descriptor are able to achieve better results than the individual parent descriptions. This is due to the fact that the fused descriptors represent two complementarity subspaces which estimate better the approximation of nearest neighbors. In addition, the largest dimension of the manipulated features is the same as the one of its original description (*e.g.,* 32+32=64 for SURF).

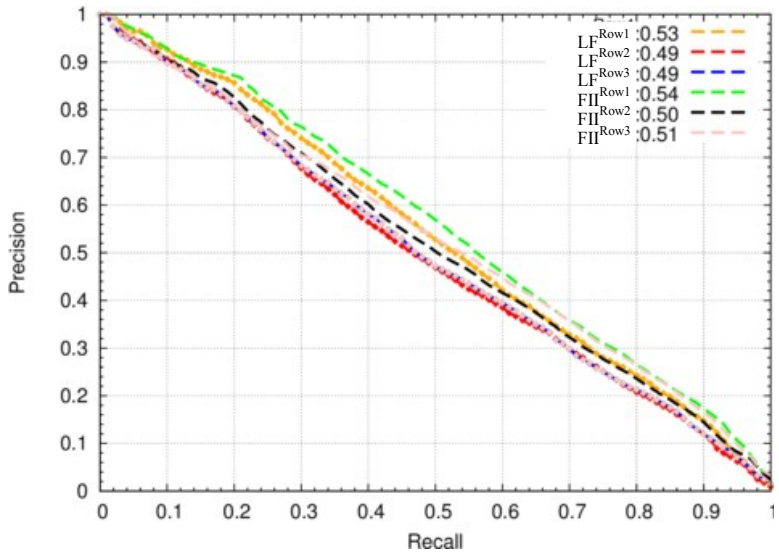
Dataset	Descriptors	LF	Time (S)	FII	Time (S)
COIL_DB	$SIFT^{PCA64}$ $SURF^{PCA32}$	0.496	0.031	0.572	0.033
	$SIFT^{PCA32}$ $SURF^{PCA20}$ SC^{PCA12}	0.487	0.043	0.589	0.045
	$SIFT^{PCA32}_{PLS32}$ $SURF^{PCA20}_{PLS20}$ SC^{PCA12}_{PLS12}	0.504	0.079	0.670	0.083
Paris_DB	$SIFT^{PCA64}$ $SURF^{PCA32}$	0.536	0.524	0.542	0.609
	$SIFT^{PCA32}$ $SURF^{PCA20}$ SC^{PCA12}	0.495	0.738	0.503	0.928
	$SIFT^{PCA32}_{PLS32}$ $SURF^{PCA20}_{PLS20}$ SC^{PCA12}_{PLS12}	0.492	1.480	0.517	3.680

Table 3.6: mAP and average retrieval time obtained with the fusion of different reduced dimensional descriptors.

In the next set of experiments, we evaluate the step of fusion of several descriptors with their reduced version. Table 3.6 shows the mAP obtained for COIL_DB and Paris_DB. The precision-recall curves of the LF and FII approaches are presented in the Fig. 3.9 corresponding to the Table 3.6.



(a)



(b)

Figure 3.9: Precision recall curves for different reduced dimensional descriptors fusion: (a) COIL_DB (b) Paris_DB.

For example, 'Row3' in the Fig. 3.9 is related to $\text{SIFT}^{\text{PCA32}}_{\text{PLS32}}$ $\text{SURF}^{\text{PCA20}}_{\text{PLS20}}$ $\text{SC}^{\text{PCA12}}_{\text{PLS12}}$, which involves 6 combinations of descriptors. Although the highest retrieval accuracy is achieved for COIL_DB is 0.67, it comes with extra computational cost due to the combination of six descriptors. For Paris_DB, the best achieved mAP is 0.54 with the reduced descriptor fusion. However, the best retrieval accuracy is achieved with the combination of without dimension reduction of SIFT-SURF-SC as presented in the Table 3.2. We observe that for FII, the best mAPs are similar to or even better than the ones obtained by fusing the original descriptions (Table 3.2), and always better than the ones of LF strategy. Note that the total amount of dimensions for the fused reduced descriptions never exceed 128, which is the dimension of the largest description used alone (SIFT). In the 'Time' column of Table 3.6, the retrieval times obtained with two fusion strategies corresponding to each combination are presented. The computational

time for LF is slightly faster since our algorithm performs several combinations to find the best nearest neighbors. The largest gap concerns Paris_DB with six combined descriptions.

3.5.6 Additional experiments on Paris_DB and COIL_DB by varying k-NN

In this section, additional image retrieval experimental results are presented using FII search engine with different combinations of descriptors and varying k-NN values ($k = 2$ and $k = 5$). We also include another image descriptor called SPIN [Lazebnik et al., 2003] to combine with other descriptors.

The retrieval results for COIL_DB are presented in the Table 3.7, where the highlighted table cell represents the best achieved result. The used codebook size is the same as used in the earlier experiments, *i.e.,* 50000. We observed that retrieval with $k = 2$ performs better compared to $k = 5$ for the majority of the combinations. The best mAP is achieved with SIFT-SURF-SC with $k = 2$. Although, SIFT-SURF-SC-SPIN combination achieved the second best mAP, but the computation cost is high due to the fusion of four descriptors.

Dataset	Descriptor combination	mAP	
		$k = 2$	$k = 5$
COIL_DB	SIFT-SURF	0.566	0.574
	SIFT-SC	0.594	0.542
	SIFT-SURF-SC	0.612	0.567
	SIFT-SPIN	0.528	0.473
	SIFT-SURF-SC-SPIN	0.606	0.559

Table 3.7: mAP obtained with different descriptors fusion and varying k-NN for COIL_DB.

Similarly, the mAPs with varying k-NN values are presented in the Table 3.8 for Paris_DB. The codebook size used is 1500000. We observed the similar trend in retrieval for Paris_DB, *i.e.,* the $k = 2$ performs better compared to $k = 5$ in most of the combinations.

Dataset	Descriptor combination	mAP	
		$k = 2$	$k = 5$
Paris_DB	SIFT-SURF	0.545	0.523
	SURF-SC	0.529	0.534
	SIFT-SURF-SC	0.546	0.528
	SURF-SPIN	0.456	0.413
	SIFT-SURF-SC-SPIN	0.525	0.505

Table 3.8: mAP obtained with different descriptors fusion and varying k-NN for Paris_DB.

3.5.7 Additional image retrieval experiments on other image datasets

In this section, two new image datasets, Oxford_DB and Holiday_DB, are used for image retrieval using FII search engine:

1. Oxford_DB: this public benchmark[4] consists of 5062 images collected from Flickr by searching for particular 11 Oxford landmarks (see first row of Fig. 3.10).

2. Holiday_DB: this dataset is a public benchmark[5] consisting of 1491 images includes a large variety of scene types. Examples are shown in second row of Fig. 3.10.

Figure 3.10: Samples from the two new datasets used in the experiments: 1^{st} row for Oxford_DB and 2^{nd} row for Holiday_DB.

The codebook size is used for Oxford_DB is 1500000 and 850000 for Holiday_DB. We observed in the Sec. 3.5.6 that better retrieval accuracy is achieved with SIFT, SURF, SC and their combinations with $k = 2$. Therefore, same descriptors and their combinations and $k = 2$ are used for Oxford_DB and Holiday_DB image retrieval.

The mAPs are presented in the Table 3.9 for Oxford_DB. The best retrieval accuracy, which is highlighted in the table, is obtained with the combination of three descriptors, SIFT-SURF-SC and with $k = 2$ for Oxford_DB. Other combinations of the descriptors also performed well, but are not able to outperform the combination of these descriptors.

Dataset	Descriptor combination	mAP
		$k = 2$
Oxford_DB	SIFT-SURF	0.481
	SIFT-SC	0.478
	SURF-SC	0.494
	SIFT-SURF-SC	0.498

Table 3.9: mAP obtained with different descriptors fusion and $k = 2$ for Oxford_DB.

[4] http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/
[5] https://lear.inrialpes.fr/~jegou/data.php

A similar set of experiments is conducted on the Holiday_DB as presented in the Table 3.10. As expected, the SIFT-SURF-SC combination obtained the highest accuracy, followed by SIFT-SURF combination.

Dataset	Descriptor combination	mAP
		$k = 2$
Holiday_DB	SIFT-SURF	0.640
	SIFT-SC	0.596
	SURF-SC	0.633
	SIFT-SURF-SC	0.646

Table 3.10: mAP obtained with different descriptors fusion and $k = 2$ for Holiday_DB.

3.5.8 Image retrieval examples by FII search engine

In this section, several examples of image retrieval by FII search engine on different datasets are presented. The tested queries are executed with different descriptor combinations and varying k values of the nearest neighbors.

An image retrieval example of Paris_DB is depicted in the Fig. 3.11. The query is executed with different combinations of descriptors. The best performing combination is SIFT-SURF-SC for Paris_DB (see Table 3.8), followed by SIFT-SURF combination. In the Fig. 3.11(a), the first 10 retrieved images are correctly retrieved with 'SIFT-SURF-SC' for the Grande Arche query. 9 out of 10 first images are correctly retrieved for SIFT-SURF combination, followed by SIFT-SC combination. In the retrieval results with the SIFT-SURF combination, shown in the Fig. 3.11(b), the 5^th retrieved image is marked as correct retrieval. Although, this image may not look similar to the 'Grande Arche' query, but the image is part of the Grande Arche architecture and belongs to the same image class. The combination of SIFT-SURF-SC represents complementarity subspaces which estimate a very precise approximation of the nearest neighbors compared to SIFT-SC.

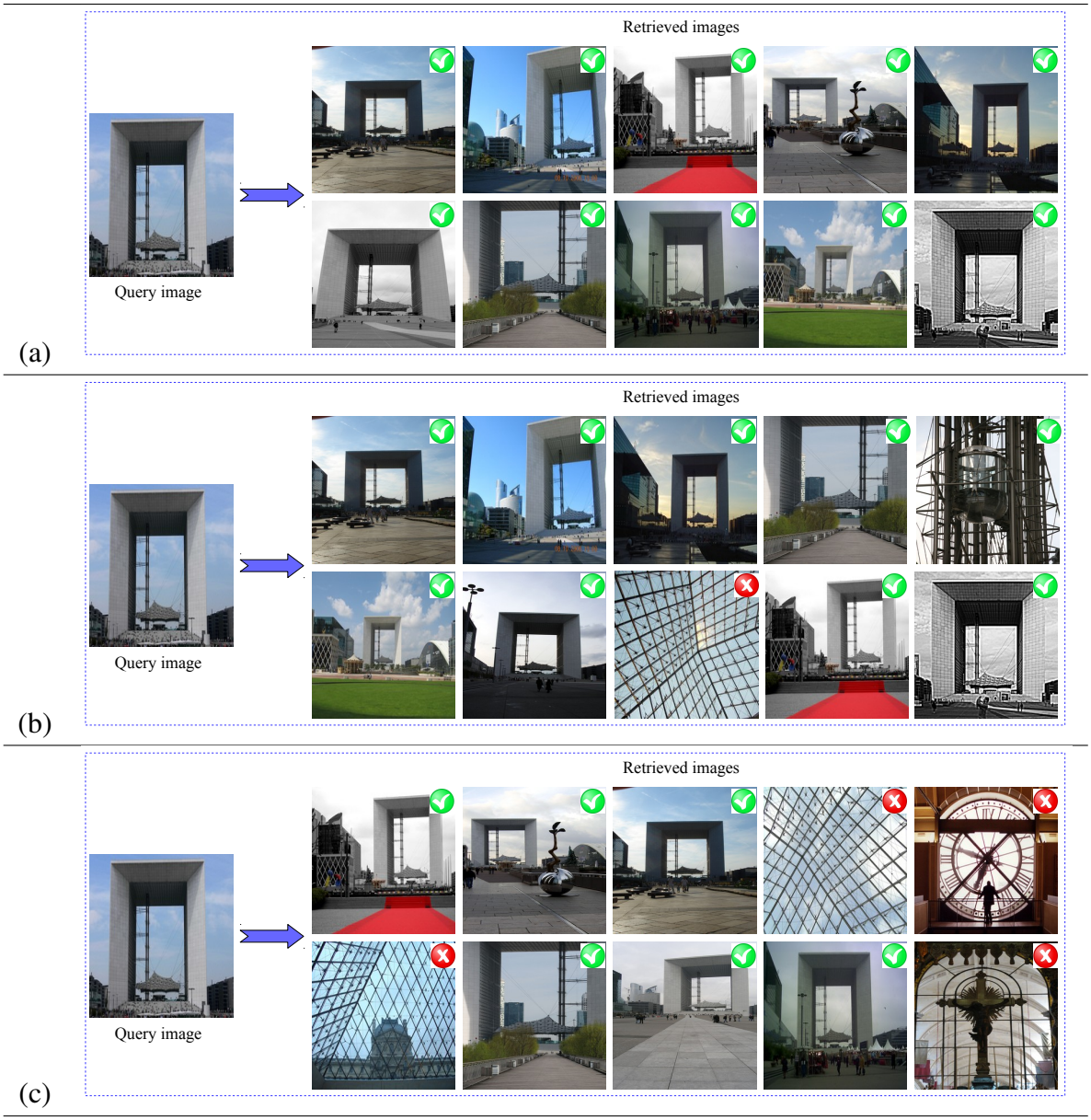


Figure 3.11: The 10 first retrieved images by decreasing order of similarity, from left to right and top to bottom with the FII search engine with $k = 2$ for Paris_DB: (a) SIFT-SURF-SC (b) SIFT-SURF (c) SURF-SC.

In the example shown in the Fig. 3.12, the query is executed with varying k ($k = 2$ and $k = 5$) while the descriptor combination is same. Globally, the $k = 2$ performs better compared to higher value k-NN for all the datasets. Similar drift is observed in Fig. 3.12 examples. Execution with $k = 2$ retrieved 9 veracious images out of 10 while only 6 images were accurately retrieved with $k = 5$.

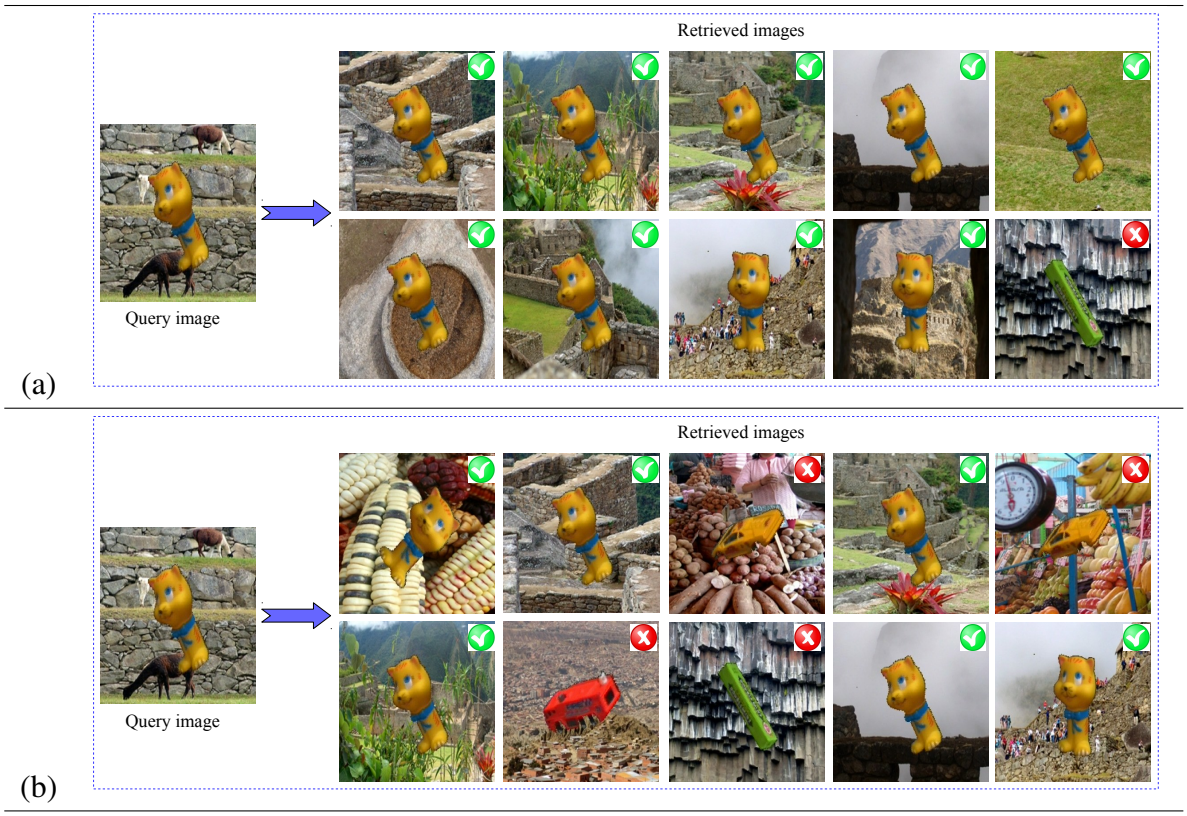


Figure 3.12: The 10 first retrieved images by decreasing order of similarity, from left to right and top to bottom with the FII search engine with SIFT-SURF-SC descriptor combination for COIL_DB: (a) $k = 2$ (b) $k = 5$.

The example in Fig. 3.13, both the SIFT-SURF-SC and SIFT-SC descriptor combinations retrieved ten correct images in the first ten retrievals.

3.6 Conclusions

In this chapter, we have proposed a strategy for the fusion of multiple image descriptors based on an improved inverted multi-index structure. Our primary goal was to develop an efficient image retrieval algorithm which is generic so that it can use several descriptors. Therefore we developed a very generic fusion strategy which is capable of combining multiple multi-dimensional image descriptors. With the experiments performed for image retrieval on different datasets, we have shown that inverted multi-index approach for single descriptor improves the image retrieval accuracy and efficiency compares to traditional inverted indexing method. Also, the experiments performed for similarity search on different image datasets have demonstrated the relevance of their combination through multi-index structure: the combination of different image characteristics clearly improves the content representation, and the strategy of fusion brings distinctiveness during the nearest neighbor search.

Our fusion strategy can be categorized as intermediate fusion due to the candidate lists, related

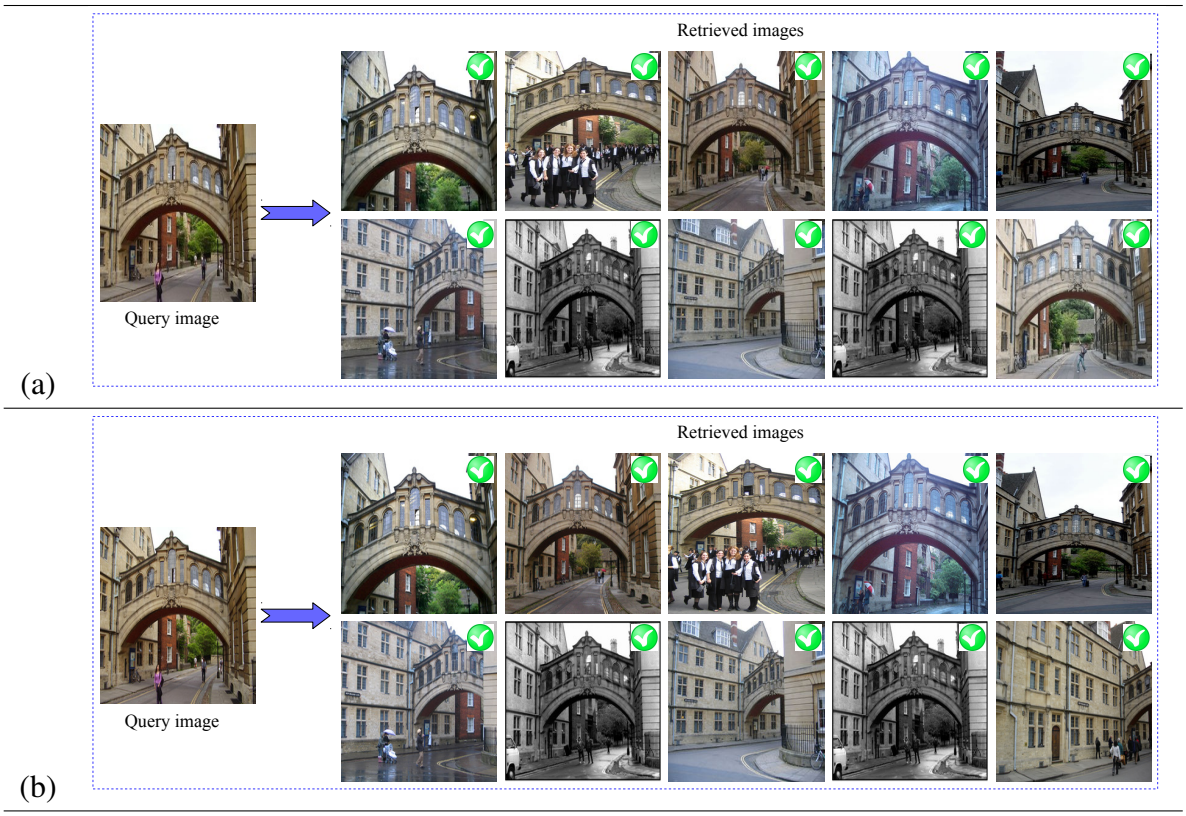


Figure 3.13: The 10 first retrieved images by decreasing order of similarity, from left to right and top to bottom with the FII search engine with with $k = 2$ for Oxford_DB: (a) SIFT-SURF-SC (b) SIFT-SC.

to closest words for each descriptor, that are merged (and not the candidate lists of images, as with late fusion). The proposal has demonstrated its superiority facing two state-of-the-art fusion approaches, such as early fusion [Yu et al., 2013] and a late fusion strategy [Neshov, 2013]. In addition, an insight of effects of using dimension reduced descriptors for image retrieval was presented in the Sec. 3.5.5. The use of complementary techniques of dimension reduction as description decomposition, PCA and PLS, contributes to improving distinctiveness during similarity search, while potentially reducing the volume of manipulated features, and then limiting the computational complexity despite the multiple descriptions involved.

In this chapter, the FII search engine is used only to combine multiple image descriptors. However, FII strategy is also capable of combining multiple image detectors. The complementarity characteristics of the image detectors and their combinations for QbE image retrieval using FII search engine are discussed in the Chapter 4. We also explore the possibilities to use FII search engine for different applications such as image retrieval and content arrangement for museum image collections, image localization, cross-domain image retrieval, etc. These applications are explored in the upcoming Chapter 5.

Chapter 4

Fusion of descriptors based on their effective spatial complementarity

4.1 Introduction

With a large number of local feature detectors and descriptors in the literature of query-by-example image retrieval, in this chapter, we propose a solution to predict the optimal combinations of local features, for enhancing the image retrieval performances. In the context of image retrieval, we have put forward a proposal of image search engine, fusion of inverted indices (FII), in the Chapter 3. FII search engine is capable of combining several multi-dimensional local image descriptors and it improves the image retrieval accuracy. In this chapter, we concentrate on the selection of the local features and their combinations so that it can enhance the content representation for image retrieval in order to further improve the retrieval accuracy. Therefore, several spatial complementarity criteria of local feature detectors are analyzed and then engaged in a regression based prediction model. The proposal can improve retrieval performance even more by selecting optimal combination for each image and also for globally for a given dataset, as well as be being profitable in the optimal fitting of some parameters of the image search engine. The experimental results highlight the importance of spatial complementarity of the features to improve retrieval and prove the advantage of using this model to optimally adapt detectors combination and some parameters. The scope of this work, presented in this chapter, in the entire thesis is highlighted in the Fig. 4.1.

This work concerns the evaluation of the complementarity of existing local features by proposing statistical criteria of analysis of their spatial distribution in the image.This work should allow highlighting a synergy between some of these descriptions when judged sufficiently complementary. All the different descriptors involved may not have the same relevance, and in addition, their distinctiveness may be different from one content to another. We think that it is important to appraise the complementarity between such local features. Due to the very rich

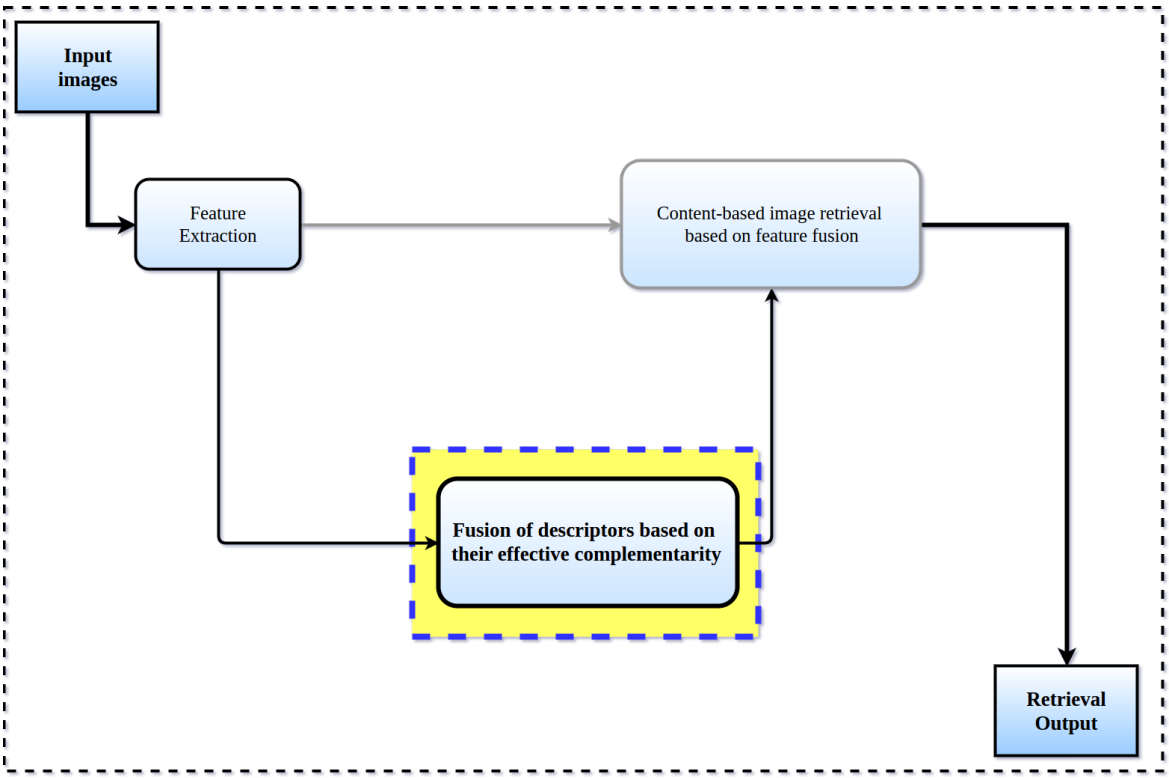


Figure 4.1: Overview of the thesis proposal. We discuss the highlighted step in this chapter.

literature on feature point detectors, which is discussed in the Sec. 2.2.1.2 of Chapter 2, that highlight detectors of different genres, such as blob, corner, symmetry, etc., we have chosen to focus on the complementarity of the detected points in the image.

Let us consider an example shown in the Fig. 4.2. Five detectors, Hessian affine (Hesaff) [Mikolajczyk and Schmid, 2004], Maximally stable extremal region (MSER) [Matas et al., 2002], Star [Agrawal et al., 2008], binary robust invariant scalable keypoints (brisk) [Leutenegger et al., 2011] and oriented and rotated BRIEF (orb) [Rublee et al., 2011], detect different interest points on an image. The spatial distribution of the points over the image are depicted in the Fig. 4.2. Hesaff detector is invariant to affine transformation and it estimates affine shape of points on the image. The detected points by Hesaff detector are represented by blue circles as depicted in the Fig. 4.2. MSER detector extracts numbers of covariant regions which are from the same image and these detected regions are represented by yellow circles as shown in the Fig. 4.2. Similarly, other detectors detect different sets of points which are spatially distributed over the image. We observe these detectors extract many distinct points or regions on the image. Therfore, we observed, the set of points detected by the combination detectors contains more information compared to the set of points detected by any of the single detectors. Thus, it is possible that these two sets, or some of them, are spatially complementary to each other.

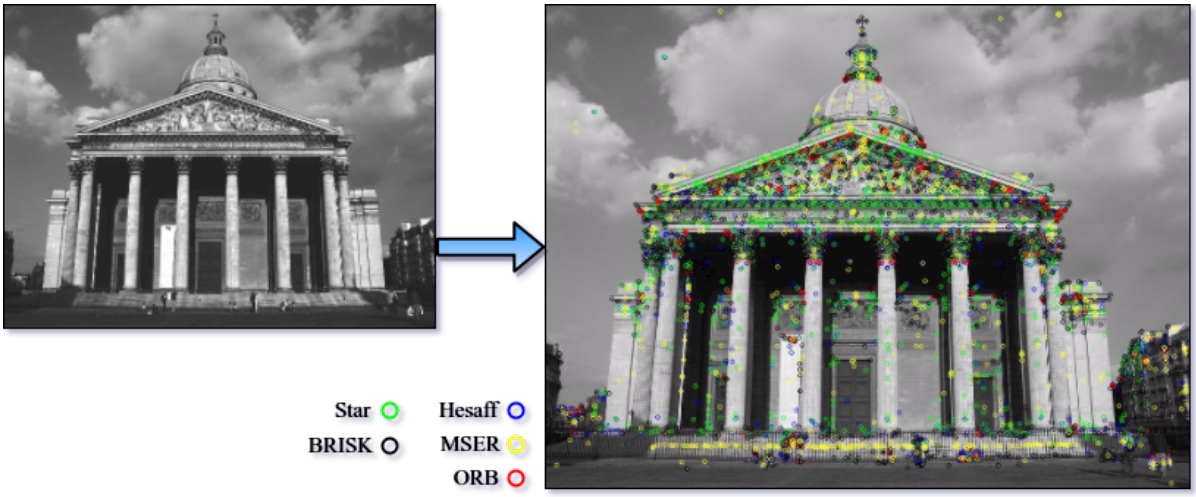


Figure 4.2: Distribution of the interest points by five different detectors over an image.

Depending on the image content, different genres of detectors can detect different sets of points which can be held more information to describe the image content. Two more examples are presented in the Fig. 4.3 and Fig. 4.4.

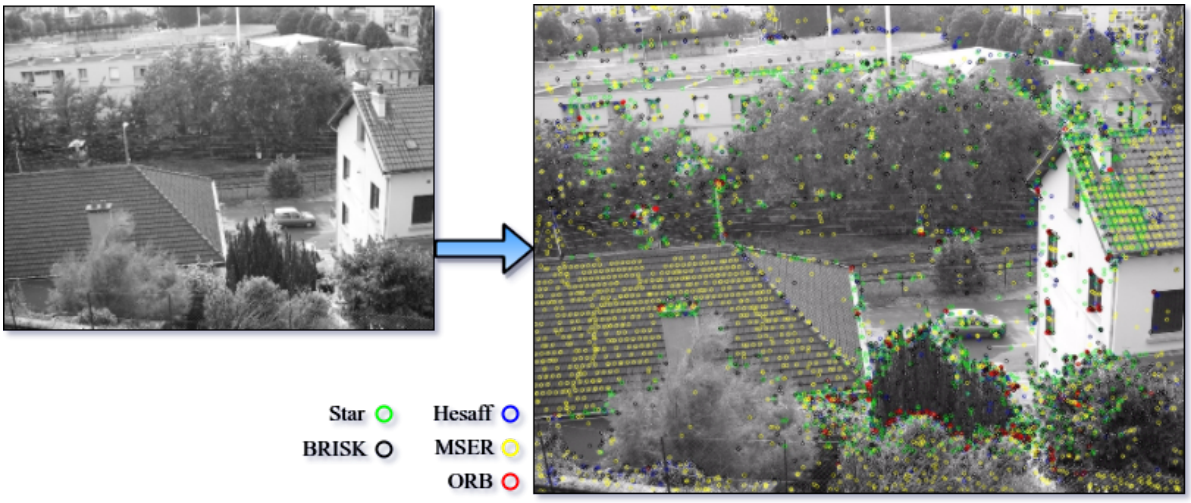


Figure 4.3: Distribution of the interest points by five different detectors over an image consists of nature and buildings.

Thus, we exploit the statistical criteria of spatial analysis, in order to give the possibility to combine several detectors and then to better describe the image content. Additionally, our ambition is to propose a solution for an optimal combination of the features for each image and not only globally for the whole dataset.

The substantial number of availability of the local feature detectors in the literature makes it arduous to determine the most relevant detector combinations for a given image content or for a given dataset. So, when we go through the literature, we have encountered several evaluation criteria to determine the effectiveness of the feature detectors.

Most of the evaluation criteria are proposed and developed depending on the target applications.

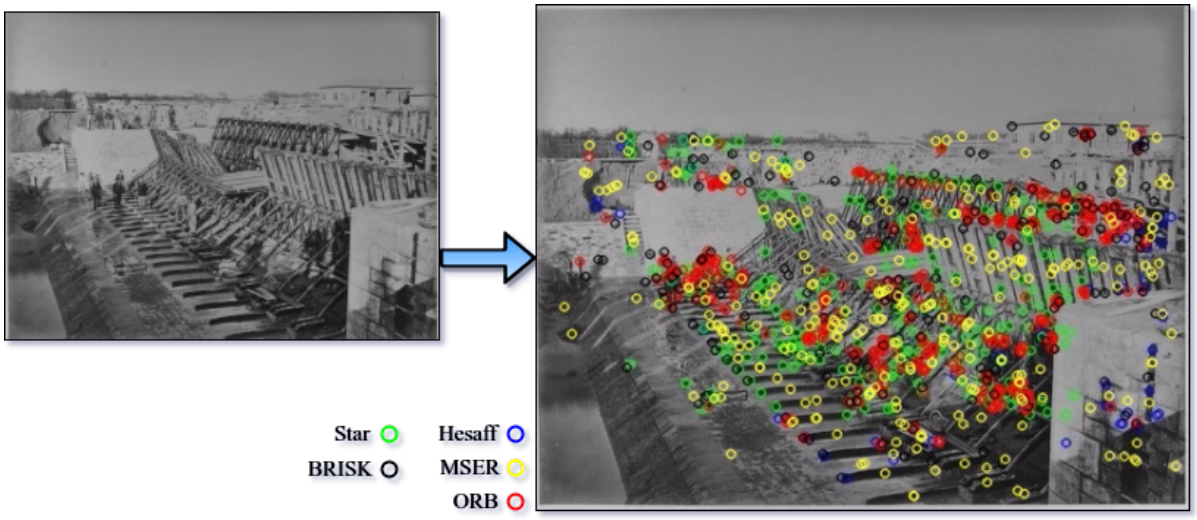


Figure 4.4: Distribution of the interest points by five different detectors over an old image consists of bridge construction.

One of the most commonly used criteria in the literature to evaluate detectors performance is repeatability [Schmid et al., 2000; Mikolajczyk and Schmid, 2004; Ehsan et al., 2010; Gales et al., 2010]. It measures the capability of the detector to identify the similar points in the images, which are submitted under different changes, such as illumination, scale view points, blur etc. In the work of [Schmid et al., 2000], the repeatability rate is defined as the total observed points detected in the both images by the same detector. Although it is popular measurement criteria, but it does not guarantee high performance of detectors. There are certain limitations of this criterion, such as neither it always determines the effect of the specific transformation on the number of corresponding detected points, nor always reflects the effect of the transformation on the matched points, and the reference images are not always fixed during the evaluation of detectors for a given image dataset. These drawbacks are addressed in the work of [Ehsan et al., 2010], where the sequence of images are considered to determine the effects of various transformations and the reference image is always the first image in the sequence. Similar measurement criteria are used in the work of [Moreels and Perona, 2007] for feature detectors and descriptors evaluation based on 3D objects. This work uses epipolar geometry to evaluate the effectiveness of the feature detectors under illumination and viewpoints changes. Repeatability gain and contribution measures criteria are proposed in the work of [Gales et al., 2010] to evaluate the complementarity between detectors. Contribution evaluation measures the volume of dissimilar points detected by the two detectors. Another measurement criteria, *i.e.,* information content is proposed in the work of [Schmid et al., 2000]. It measures the distinctiveness of the detected interest points, which are described by the local grayvalue shape descritor, by measuing the entropy of the detected points. Spatial clustering based evaluation is proposed in the work of [Mikolajczyk et al., 2005]. It determines how differenet sets of interest points extract similar local structure in a cluster.

In the work of [Haja et al., 2008; Zeisl et al., 2009], localization accuracy is used to evaluate

the effectiveness of the detectors. The localization accuracy is computed by the position and region based measurement of the detected points in the work of [Haja et al., 2008], where a set of 6 sequences of images is used as test images. [Zeisl et al., 2009] proposed a framework to evaluate the localization accuracy for the feature points, which are detected at the different image scales. To evaluate the complementarity of the local features, entropy coding scheme based method is proposed by [Dickscheid et al., 2011]. In this work, the density of the local features is computed by measuring the distance to entropy density of the local image patch. In the work of [Dickscheid and Förstner, 2009], convex hull approach is proposed, where the spatial distribution of the feature points is measured. However, to evaluate these approaches needs benchmark with the ground truth, may not always be available for many datasets.

Although certain criteria are proposed in the literature, the use of these criteria in the image retrieval context are not widely considered. In the Sec. 2.4 of Chapter 2, we have discussed several approaches of features combination in the context of image retrieval. Although several fusion strategies are proposed in the literature, we notice that the most of the strategies do not consider complementarity characteristics of the features while combining them. The features combined are chosen a priori, according to their presupposed complementarity.

In this context, it is important to know that the efficacy of the feature fusion strategies may not only rely on the fusion strategy but also on the selection of the features. For example, a hybrid method [Rashedi et al., 2013] is proposed for simultaneous feature adaptation and feature selection, for a given dataset. In this approach, the parameter optimization during feature extraction and feature selection are carried out on a subset of dataset images by employing mixed gravitational search algorithm. In the work of [Zhou et al., 2015], a rank based graph fusion technique is proposed by combining deep features, global and local features. The best feature combination is selected globally for a dataset based on the retrieval performance. [Sun, 2014] proposed a method for local selection of image features for similarity search and similarity graph construction. This is achieved by computing local laplacian score and feature sparsification and considering the importance of the local neighborhood of each image point with respect to the image. Note that in all these approaches, the optimal combination of features is carried out globally for a whole dataset and not locally for each image. We focus on the fusion of features for a given dataset and as well as each query image based on their effective spatial complementarity.

Our main contribution pivots on the proposal of a regression model that involves several complementarity statistical criteria of spatial analysis of feature detectors. Mean average precision (mAP), as evaluation measure of the quality of the content description, is incorporated to train the model and assists users to anticipate the proper feature combinations for each query image of the dataset. In the Chapter 3, a query-by-example image search engine, *i.e.,* fusion of inverted indices (FII) was proposed. The FII is employed for mAP computation in the proposed regression model. Additionally, we demonstrate that this proposal allows to optimally fit some other parameters of the FII search engine, such as the best k during the k-nearest neighbor search.

This chapter is organized as follows: Sec. 4.2 is dedicated to the discussion of several spatial complementarity criteria existing for local detectors. In the Sec. 4.3, we present the details about linear regression model, and then the proposal of image retrieval based on prediction model is presented in the Sec. 4.4. The experiments and evaluation of our proposed method are presented in the Sec. 4.5, followed by conclusion in Sec. 4.6.

4.2 Evaluation of the spatial complementarity

Literature on local feature detectors (see Sec. 2.2.1.2 of Chapter 2) is rich. Due to the different properties of the local detectors, it is possible that the interest points detected by the different detectors are not the same. The interest point detected by the different detectors may not have the same importance and also the distinctiveness of the points may differ from one image to another. In the Sec. 4.1, we presented the existing critera to determine the potency of a feature detector. We revisit here three criteria to evaluate the spatial complementarity of between detectors, which will be exploited in our prediction based image retrieval model. The presentation is restricted to pairs of detectors, but can easily be generalized to the complementarity of sets of detectors.

Our hypothesis is that better the detections are spread in the image, better the content is described, first because the detections would have more chance to describe the many areas of the image, and second because distant detections should statistically increase the variety of the associated descriptions, making the whole content description more distinctive. A similar idea is recognized in the work of [Sattler et al., 2016] on geometric burstiness problem in location recognition. The geometric burstiness can be seen as a problem of co-occurrence of visually similar features closely within the image and as well within multiple images of the database. The spatial distribution of the features in the images (database and query) are largely ignored in most of the conventional image retrieval approaches. The common assumption in image retrieval is that if the features are matched between the images, then not many inliers will appear in the unrelated images. However, this assumption does not satisfy as geometric burst likely to occur with the large database, if it contains several visually similar contents. As a consequence, unrelated images will contain a high number of inliers and will affect the retrieval performance. In the work of [Sattler et al., 2016], the burstiness problem is tackled by down grading the weight of the visually similar features in the images.

In our work, we focus on the larger spatial distribution of the features. To achieve this goal, we exploit several detectors of various natures, such as detectors of corners, of salient interest points, of local symmetries, of blobs, etc., with the ambition of maximizing the spatial distribution of these detections in the image. To give more insight before jumping to complementarity evaluations criteria, we consider two combinations of two detectors, 'hesaff-mser' and 'har-colsym'. Hesaff-mser has been evaluated more efficient compared to 'har-colsym' combination

on the considered dataset Paris_DB (see Sec. 3.5.1 in Chapter 3) in terms of spatial complementarity. For each complementarity score and each query image, we plot the difference of scores between the two combinations vs. the corresponding difference in the mAP the Fig. 4.5. We observe that globally, complementarity scores values increase with mAP values (most of the points are in the area related to positive axes).

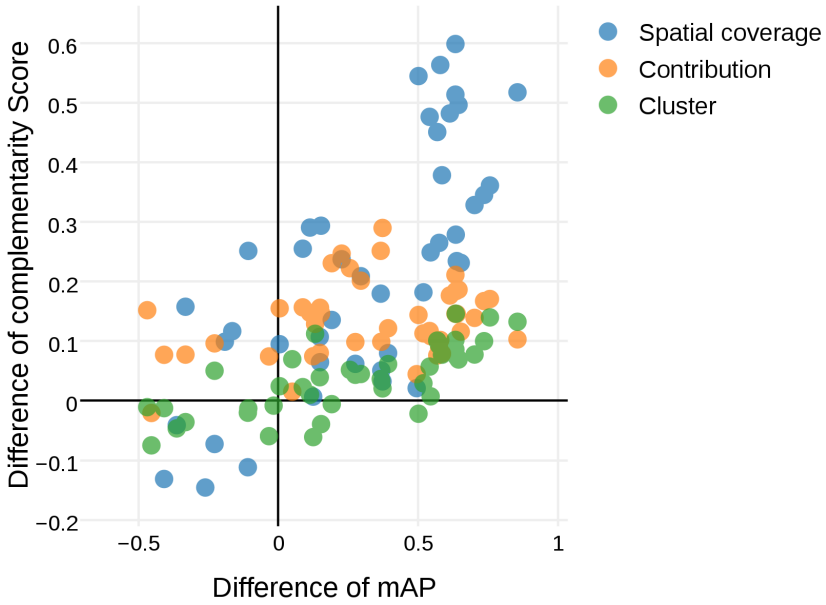


Figure 4.5: Relationship between complementarity scores differences and mAP differences, for each query image and two combinations of two detectors.

Therefore, to measure spatial complementarity information, we have chosen to explore criteria that evaluate the spatial complementarity of two detectors, which will be exploited in our prediction model. We revisit here three criteria in Sec. 4.2.1, 4.2.2 and 4.2.3. For sake of clarity, the presentation is restricted to pairs of detectors.

Let us consider the sets of keypoints extracted from an image (Im) by two detectors, D_a and D_b, are

Detector	Points
D_a	$d_a^1(x_a^1, y_a^1), d_a^2(x_a^2, y_a^2), \ldots, d_a^n(x_a^n, y_a^n) \mid\mid D_a = n$
D_b	$d_b^1(x_b^1, y_b^1), d_b^2(x_b^2, y_b^2), \ldots, d_b^m(x_b^m, y_b^m) \mid\mid D_b = m$

4.2.1 Analysis of the spatial coverage

In general, several factors are considered to determine the effectiveness of a feature point detector. One of the key factors is the distribution of interest points in an image detected by detectors. In this measurement criteria [Ehsan et al., 2013] we compute how well the sets of keypoints (detected by different detectors) are distributed over an image. In other sense, it measures the average coverage of the points in an image. Initially, the distance between keypoints

is computed. If the coverage of a keypoint set is larger than the others, that means the points are well distributed in an image. When we calculate the distribution of the points from two detectors, it is expected to gain large coverage of the distribution if the keypoints from two detectors are distinct. That also implies the better complementarity of two detectors.

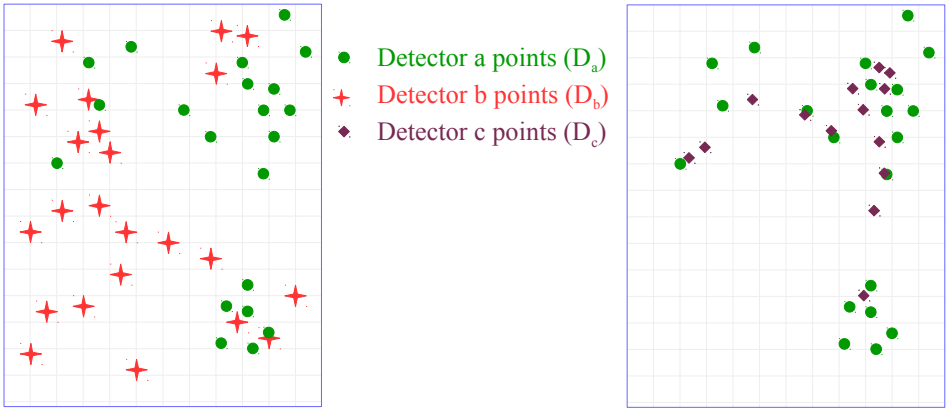


Figure 4.6: Illustration of the distribution of points by two different detectors over an image: (a) Keypoints from detectors D_a and D_b are distinct (b) Keypoints from detectors D_a and D_c represent similar regions of the image.

In the Fig. 4.6, we depicted two scenarios by using two detector combinations. In the first scenario, *i.e.,* 4.6(a), the extracted keypoints by the two detectors (D_a and D_b) are distinct and well distributed on the image. On the contrary, in Fig. 4.6(b), the detected keypoints are quite close spatially. Therefore, it is expected that the the D_a and D_b combination will have larger spatial coverage compared to D_a and D_c.

In order to compute spatial coverage, first, a keypoint, *e.g.,* $d_a^i(x_a^i, y_a^i)$, is considered as a reference point and Euclidean distances (ED_j^i) are calculated with other $(n + m - 1)$ points of $D_a \cup D_b$.

$$ED_j^i = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{1}$$

$$where \quad j = 1, \cdots, (n + m - 1) \ \ \& \ \ i \neq j$$

If two points detected by the two detectors are the same, there is no effect on the overall distribution. In order to neutralize the effect of the extreme outliers on the overall spatial distribution of $D_a \cup D_b$, the coverage measure is based on the harmonic mean. The mean of the distances is computed as:

$$EDMean_{nm}^i = \frac{n + m - 1}{\sum_{j=1, j\neq i}^{n+m-1} (1/ED_j^i)} \tag{2}$$

This step is reiterated for each keypoint of D_a and D_b considering each keypoint as a reference. The distribution complementarity score (Di_{cs}) is computed as:

$$Di_{cs} = \frac{n+m}{\sum_{i=1}^{n+m}(1/EDMean_{nm}^{i})} \tag{3}$$

A low score implies the detected points are similar, hence they are less complementary to each other. On the other hand, a higher complementarity score indicates the greater distribution of the keypoints all over the image, which implies better complementarity between the detectors.

The algorithm to compute spatial coverage is given below:

Algorithm 8 – DISTRIBUTION COMPLEMENTARITY SCORE

INPUT: Detector a of n keypoints (D_aKp_n); Detector b of m keypoints (D_bKp_m)
OUTPUT: Distribution complementarity score(Di_{cs})

1. *Declare* $Distributioncomplementarityscore{:}Di_{cs}$
2. *Declare vector of* $EucleadeanDistance : ED_j^i$
3. *Declare vector of* $EucleadeanDistanceMean : EDMean_{nm}^i$
4. *For each* $n, coordinate$ *in* D_aKp
5. $\quad$ *For each* $m, coordinate$ *in* D_bKp
6. $\quad\quad ED_j^i \leftarrow ComputeEucleadeanDistance(D_aKp_n, D_bKp_m)$
7. $\quad\quad EDMean_{nm}^i \leftarrow ComputeEucleadeanDistanceMean(ED_j^i)$
8. $Di_{cs} \leftarrow ComputeDistribution(EDMean_{nm}^i)$
9. *Return* Di_{cs}

We present an example with different scenarios to compute spatial coverage between the detectors in Appendix C.1.

4.2.2 Complementarity by contribution measure

The contribution criterion [Gales et al., 2010] is a measure of the amount of dissimilar points detected by two detectors. In this method, first, the total number of detected keypoints are calculated. Then the points detected in common between the detectors are determined.

It is possible that two detectors extract a certain number of same keypoints (p) for an image. The same detected points reduce the contribution measure of D_b over D_a and vice versa.

As depicted in the Fig. 4.7, most of the detected keypoints by the D_a and D_b are distinct. This implies a better complementarity between these two detectors.

The contribution of D_b over D_a ($Cn_{D_b|D_a}$) is computed as:

$$Cn_{D_b|D_a} = \frac{n-p}{n} \tag{4}$$

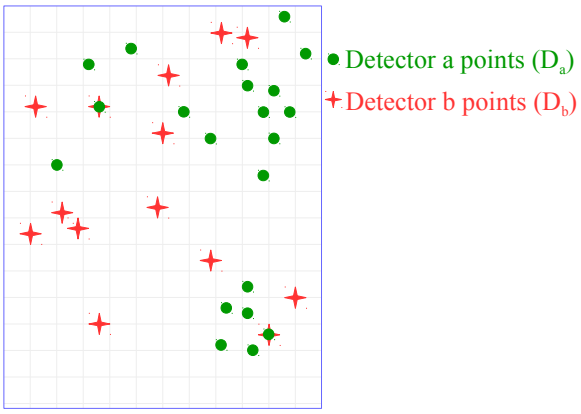


Figure 4.7: Illustration of the contribution measure for two detectors.

The overall complementarity between D_a and D_b is measured by:

$$Cn_{cs} = min(Cn_{D_b|D_a}, Cn_{D_a|D_b}) \tag{5}$$

If the detected keypoints between the detectors are different, the score goes high. The maximum score can be achieved is 1, when the detected keypoints are completely different. Increasing number of same keypoints will reduce the complementarity score.

Examples of the contribution measure between the detectors are presented in Appendix C.2.

4.2.3 Cluster-based measurement of complementarity

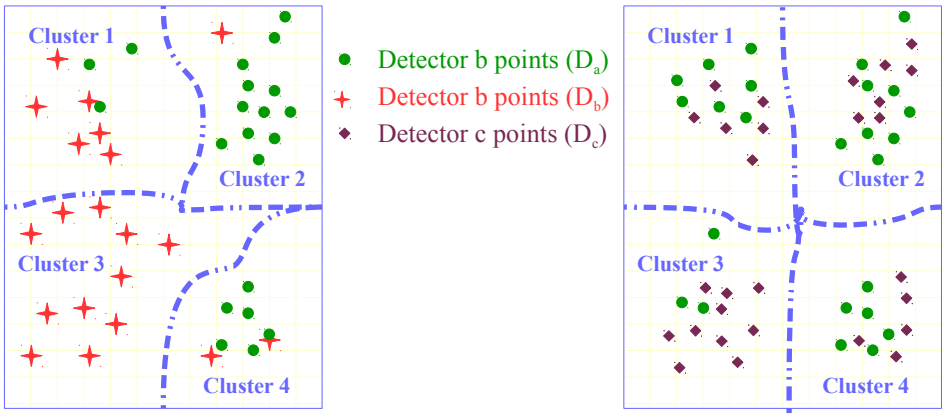
Based on spatial clustering, this measure [Mikolajczyk et al., 2005] determines how the different detectors extract similar local structures in a cluster. As depicted in the Fig. 4.8(a) and 4.8(b), the clusters are generated in the image space from extracted points of D_a-D_b and D_a-D_c, using a clustering algorithm (*e.g.,* k-means). In the first scenario, the clusters are mostly represented either by the points from D_a or D_b. However, in the second scenario, each cluster is represented by the points from the both the detectors. Therefore, D_a and D_b have a better complementarity to each other compared to D_a and D_c.

Now, each cluster $(c_j, j = 1 \ldots k)$ may contain points from D_a and/or D_b. Points from D_a and D_b in cluster c_j, *i.e.,* respectively. F_{jD_a} and F_{jD_b}, contribute to the total number of points (F_j) present in c_j. The frequency of the points from D_a and D_b in c_j is computed as:

$$p_{jD_a} = \frac{|F_{D_a}|}{|F_j|} \quad \& \quad p_{jD_b} = \frac{|F_{D_b}|}{|F_j|} \tag{6}$$

The whole complementarity score (Cl_{cs}) can be computed as:

$$Cl_{cs} = 1 - 2.\frac{1}{k}\sum_{j=1}^{k} min(p_{jD_a}, p_{jD_b}) \tag{7}$$



(a) (b)

Figure 4.8: Explanation of the cluster-based measurement: (a) Clusters are represented by either D_a or D_b (b) Clusters are euqally shared by D_a and D_c

When p_{jD_a} and p_{jD_b} are both close to 0.5, the score is close to 0, which indicates a small complementarity between two detectors. When one probability is close to 0 and the other is close to 1, the score is close to 1; it indicates a better complementarity of the detectors.

Algorithm 9 – CLUSTER COMPLEMENTARITY SCORE

INPUT: Description a files (Dsc_{a_n}); Description b files (Dsc_{b_m}); Cluster (Cl)
OUTPUT: Cluster complementarity score(Cl_{cs})

1. *Declare Cluster complementarity score:* Cl_{cs}
2. *Declare count Description a points in cl:* D_acl
3. *Declare Count Description b Points in cl:* D_bcl
4. *Declare Probability Description a points in cl:* PD_acl
5. *Declare Probability Description b points in cl:* PD_bcl
6. *For each cluster l in cl*
7. $\quad D_acl[l] \leftarrow ComputePointsCluster(Dsc_{a_n}, cl[l])$
8. $\quad D_bcl[l] \leftarrow ComputePointsCluster(Dsc_{b_m}, cl[l])$
9. $\quad PD_acl \leftarrow D_acl[l]/D_acl[l] + D_bcl[l]$
10. $\quad PD_acl \leftarrow D_bcl[l]/D_acl[l] + D_bcl[l]$
11. $\quad Cl_{sc} \leftarrow ClusteCompute(PD_acl, PD_bcl)$
12. *Return* Cl_{sc}

The examples of the cluster-based complementarity between the detectors are explained in Appendix C.3.

4.3 Learning the complementarity of detectors with a regression model

Regression analysis is a statistical process for estimating the relationship among variables. It includes many techniques for modelling and analyzing several variables when the focus is on the relationship between a dependent variable and one or more independent variables. More specifically, regression analysis helps to understand how the typical value of the dependent variable changes when any one of the independent variables is varied, while the other independent variables are held fixed. Most commonly, regression analysis estimates the conditional expectation of the dependent variable given the independent variables, *i.e.,* the average value of the dependent variable when the independent variables are fixed. It is widely used for prediction and forecasting, where its used has substantial overlap with the field of machine learning. It is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships.

In our complementarity evaluations, three spatial complementarity scores (presented in the Sec. 4.2) and the number of extracted keypoints per image between different detector combinations have been calculated for each query image. The evaluation of the retrieval result, *i.e.,* mean Average Precision (mAP) is also computed for the training image dataset which is assumed as the baseline or used to train the regression model. Therefore, the dependent variable is mAP and the four independent variables are three spatial complementarity scores and the number of keypoints. The motivation is to establish whether there is any relation between these dependent and independent variables by regression analysis. The idea is to establish a suitable regression model and use the same model to predict best appropriate detector combinations for other datasets. In this context, we are interested in linear regression model.

4.3.1 Linear regression model

Linear regression model [Montgomery et al., 2012] defines the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. If the number of independent variables used in the model is one, the model is called simple linear regression. When more than one independent variables are used in the model, the model is known as multiple linear regression model. Linear predictor functions are used to train the regression model and the dependent variable is estimated from observed data. Most commonly, linear regression refers to a model in which the conditional mean of dependent variable given the values of independent variables is an affine function.

Let us consider, for a given observational dataset, y is the dependent variable and $x_1, x_2...x_p$ are the dependent variables. There are i observational data where $i = 1, ..., n$. The linear regression

equation is given by:

$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} \tag{8}$$

$$y_i = \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta} \tag{9}$$

Here T is the transpose matrix, β is the regression coefficient matrix. The term $x_i^T \beta$ is the inner product between x_i and β.

Therefore these n equations can be written as below

$$y = \mathbf{X}\boldsymbol{\beta} \tag{10}$$

$where$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \qquad X = \begin{pmatrix} \mathbf{x}_1^{\mathrm{T}} \\ \mathbf{x}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{x}_n^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ x_{21} & \cdots & x_{2p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}$$

$$\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}$$

Several estimation methods, such as least square, maximum likelihood, etc., were developed to estimate the parameters in the linear regression. One of the estimator technique is ordinary least square estimation (OLS) to estimate unknown parameters in linear regression model. OLS is used for both observational and experimental data. It reduces the difference between observed data for training dataset and predicts the responses for experimental data by linear approximation of the data. The OLS method reduces the sum of squared residuals, and computes an expression for the estimated value of the unknown parameter β as in equation below:

$$\hat{\boldsymbol{\beta}} = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}y \tag{11}$$

$$= \big(\sum x_i x_i^{\mathrm{T}}\big)^{-1}\big(\sum x_i y_i\big) \tag{12}$$

The estimator is unbiased and consistent if the errors have finite variance and are uncorrelated with the regressors. The OLS estimator is used because it is consistent when there is multicollinearity and the model parameters are exogenous.

Once the regression model is trained, it is important to analyze whether the model is properly fit for the observed data or not. To determine the regression model fitness, the coefficient of determination (R^2) is widely used. It indicates whether the observed data is fitted with the statistical model. R^2 gives a quantification of how well observed predictions are replicated

by the regression model. The proportion of total variation in outcomes is explained by the regression model. The mathematical definition to compute R^2 is:

$$R^2 = 1 - \frac{SSE}{SST} \tag{13}$$

$where$

$$SSE = \sum_{i=0}^{n} (y_i - \widehat{y_i})$$

$$SST = \sum_{i=0}^{n} (y_i - \overline{y_i})^2$$

$y_i = Observed\ response$

$\widehat{y_i} = Predicted\ response$

$\overline{y_i} = Mean\ of\ observed\ response$

$SSE = \sum\ of\ squared\ error$

$SST = Total \sum\ of\ squares$

A Higher value of R^2 indicates better regression model fitness. The problem with R^2 value occurs when multiple predictors are used in the model. R^2 increases with the addition of a number of predictors in the model. Consequently, a model with more terms may appear to have a better fit simply because it has more terms. If a model has too many predictors and higher order polynomials, it begins to model the random noise in the data. This condition is known as over-fitting the model and it produces misleadingly high R^2 values and increases the possibilities to make the wrong prediction. In this context, adjusted R^2 is introduced. It takes into account the explanatory power of regression models that contain several predictors. If the newly added predictor improves the model more than would be expected by chance, the value of adjusted R^2 increases and vice versa. Adjusted R^2 can be computed as:

$$AdjustedR^2 = R^2 - (1 - R^2)\frac{n-1}{n-p-1} \tag{14}$$

where n is a total number of observations, p is the number of predictors. Model fitness increases with the increasing value of adjusted R^2. In this work, we consider adjusted R^2 to evaluate the model fitness.

4.4 Image retrieval based on a regression model and complementarity measures

To perform image retrieval, we propose to learn a regression model based on the complementarity criteria revisited in Sec. 4.2, on a number of detected interest keypoints per image (Kp)

and on mean Average Precision (mAP) as image retrieval system output. The choice of these parameters are discussed in the Sec. 4.5.2 which concerns experiments on the regression model. The objective is to predict the best detector combinations for an image dataset. We will also experiment that the proposal allows fitting some parameters.

We assume that the relationship between the complementarity criteria and the mAP is general for all image datasets, and we employ a linear regression model:

$$mAP = \beta_1 Kp + \beta_2 Di_{cs} + \beta_3 Cn_{cs} + \beta_4 Cl_{cs} \tag{15}$$

where β_i are model coefficients.

Our proposed framework is comprised of two stages:

1. Training stage *i.e.,* traing the regression model and

2. Testing stage *i.e.,* prediction of the best detector combination on new datasets.

The block diagram of the proposed framework is depicted in the Fig. 4.9 to give a general overview of the entire framework.

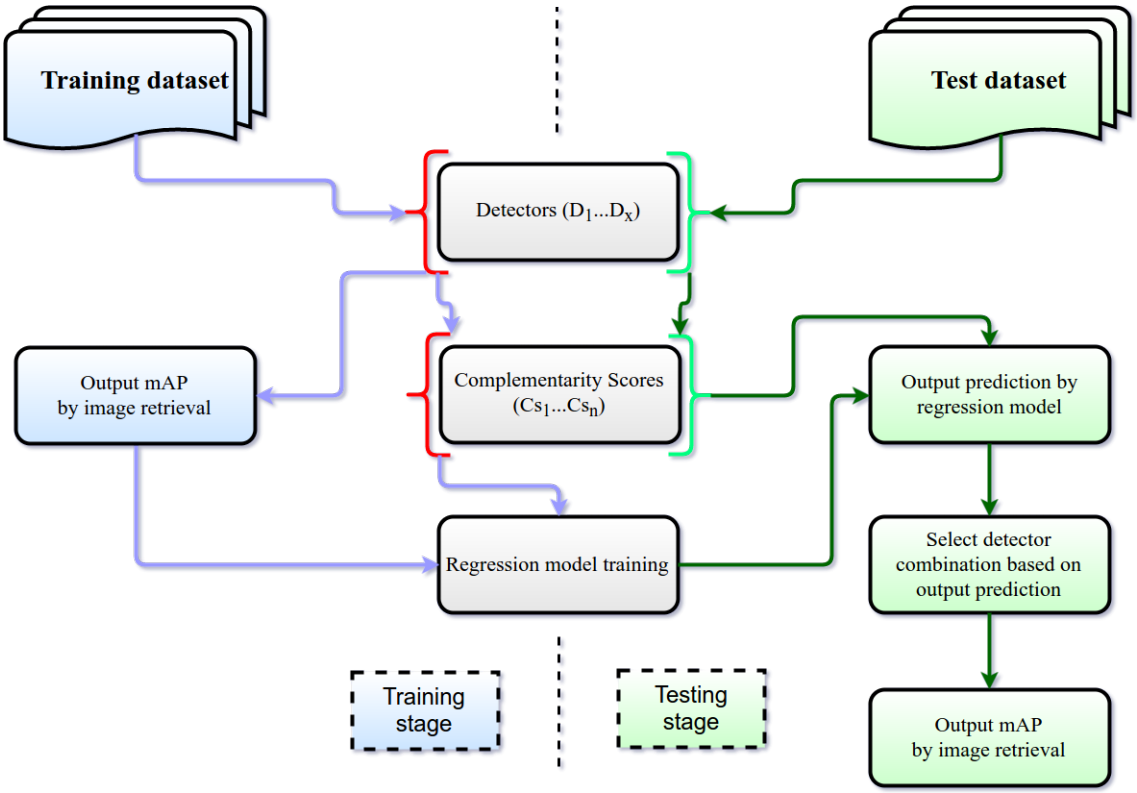


Figure 4.9: Block diagram of the proposed image retrieval framework based on complementarity and regression model.

4.4.1 Training of the regression model

The training step is decomposed into following steps involving complementary criteria and mAP:

1. Several different detectors, *e.g.,* $D_a, \ldots, D_x$, are used to extract keypoints from images, leading to x sets of keypoints.

2. Here, we consider the C_x^2 couples of detectors $(D_i, D_j)_{i \neq j}$ and compute for them the three complementarity scores, such as in 4.2.1, 4.2.2, and 4.2.3, for each image of the dataset. We also keep the number of keypoints (Kp) per image.

3. One mAP is then computed for the images dataset described with a couple of detectors $(D_i, D_j)_{i \neq j}$, using a classical approach of query-by-example retrieval able to use several descriptors jointly, such as in FII image search engine presented in Chapter 3. We obtain C_x^2 $mAPs$.

4. Finally, the relationship between the complementarity scores and the retrieval output (mAP) is learned by a linear regression model according to Eq. 15.

5. Coefficients of determination *i.e.,* adjusted R^2, which measures the explanatory power of regression model with multiple predictors, is calculated (see Sec 4.3.1). Adjusted R^2 is effective to overcome the overfitting issue of a model with multiple inputs, and it analyzes the model fitness and determines the best model for prediction for the given inputs and output.

4.4.2 Prediction of the best detector combination

The prediction steps of the best detector pair for a new dataset are:

1. The detectors $D_1, \ldots, D_x$ extract keypoints on each image of the new dataset. The three complementarity scores of the detector pairs are computed, similarly to step 1 of Sec. 4.4.1.

2. For each detector combination $(D_i, D_j)_{i \neq j}$, we predict the mAP, called mAP^p, using previously trained regression model. The complementarity scores of each detector pair are the inputs for the regression model. The outputs, mAP^p are predicted using the model parameters and the inputs.

3. The detector pair with the highest mAP^p is selected as the suitable detector pair for image retrieval on the new dataset.

The approach of prediction presented above predicts the best detector combination *globally* for a given dataset. It can be directly employed to predict the best combination *for each query image*, which can be different from an image to another; the three complementarity scores are simple criteria that can be computed very quickly online. In the experiment Sec. 4.5, we will see that the quality of image retrieval can be improved even more by considering such an image-by-image prediction. We will also see that the regression model can be employed to predict some other parameters, such as the k during k-NN retrieval.

4.5 Experiments and evaluations

This section presents and discusses the experiments conducted to evaluate the contributions of this chapter.

First we discuss about the evaluation framework in the Sec. 4.5.1, *i.e.,* image datasets and parameter configurations use in the experiments. In the Sec. 4.5.2, we discuss the selection of the regression model for detector combination prediction. It follows by the presentation on a global prediction of the detector combinations based on regression model in the Sec. 4.5.3. Next, to validate the detector combination predictions, the effective retrieval performances are presented in the Sec. 4.5.4 and the comparison with other state-of-the-art approaches are discussed in the Sec. 4.5.5. Section 4.5.6 focuses on the effect of the k nearest neighbor values on image retrieval performances. In the Sec. 4.5.7, we present experimental results of the image-by-image adaptive selection of the detector combination based on the regression model and how it can improve the overall retrieval accuracy, followed by several image retrieval examples in the Sec. 4.5.8.

4.5.1 Framework of evaluation

The experiments are conducted on three public image datasets:

1. Paris_DB: This dataset is introduced in the Sec. 3.5.1 of Chapter 3.
2. Oxford_DB: We introduced this dataset in the Sec. 3.5.7 of Chapter 3.
3. Holiday_DB: We introduced this dataset in the Sec. 3.5.7 of Chapter 3.

Sample images from these datasets are shown in the Fig. 4.10. We have selected 7 detectors from characteristically diverse categories such as blob, corner, symmetry, etc.: Hessian affine (hesaff) [Mikolajczyk and Schmid, 2004], color symmetry (colsym) [Heidemann, 2004], MSER (mser) [Matas et al., 2002], Harris (har) [Schmid and Mohr, 1997], Star (star) [Agrawal et al., 2008], binary robust invariant scalable keypoints (brisk) [Leutenegger et al., 2011] and oriented

Figure 4.10: Samples from the three benchmarks used in our experiments: 1^{st} row for Paris_DB, 2^{nd} row for Oxford_DB, 3^{rd} row for Holiday_DB.

and rotated BRIEF (orb) [Rublee et al., 2011]. Extracted keypoints are described by three complementary local descriptors (*i.e.,* SIFT [Lowe, 2004], SURF [Bay et al., 2008] and SC [Belongie et al., 2002]) and used jointly in the query-by-example image search engine (fusion of inverted index (FII) image search engine which is presented in Chapter 3).

Image retrieval performances are presented with mean Average Precision, *i.e.,* mAP as computed in the Eq. 9 of Chapter 3. Codebook size and value of k during nearest neighbors (k-NN) retrieval are two important parameters of the FII search engine. Optimal codebook size used for Paris_DB and Oxford_DB is 1500000 words. For the Holiday_DB, 30% of the total description points of each detector combination is selected as codebook size. Parameter k is varied in between 2 to 10 for an optimal combination of the nearest neighbors.

4.5.2 Study of the optimal regression model

In this section, we discuss the training of the regression model with different combinations of model inputs, and then the selection of the best suitable regression model for the prediction on the test datasets. As regression model inputs, we consider the complementarity scores and the number of keypoints.

Dataset	Model inputs	Adjusted R^2 value
Paris_DB	Distribution-Contribution	0.123
	Distribution-Cluster	0.126
	Contribution-Cluster	0.003
	Kp-Distribution-Contribution-Cluster	0.166

Table 4.1: Adjusted R^2 value calculation for the regression model with different combinations of the complementarity scores, and with Kp, the number of keypoints in the image.

Paris_DB is used to train the regression model. Model inputs, the complementarity scores, *i.e.,* distribution, contribution, cluster and number of keypoints (Kp), of detectors pairs are computed for the images of Paris_DB. The mAP is calculated using the FII search engine on Paris_DB. We trained our model with different combinations of the inputs and calculated adjusted R^2 to determine the best-fitted model. In the Table 4.1, we present four different configurations and corresponding adjusted R^2. The highest value of adjusted R^2 is achieved with 'Kp-Distribution-Contribution-Cluster - mAP' model.

To confirm the optimality of this particular configuration, we use Paris_DB for prediction with the two best models, *i.e.,* 'Kp-Distribution-Contribution-Cluster - mAP' and 'Distribution-Contribution - mAP'. The prediction steps are explained in the Sec. 4.4.2. The predicted mAP (mAP^p) are shown in the Table 4.2 and Table 4.3 correspondingly. We observe that the best performing detector pair is 'mser-star' according to 'Distribution-Contribution - mAP', and 'hesaff-mser' with 'Kp-Distribution-Contribution-Cluster - mAP'.

Dataset	Detector pair	mAP^p	Detector pair	mAP^p	Detector pair	mAP^p
Paris_DB	hesaff-colsym	0.512	hesaff-mser	0.548	hesaff-har	0.481
	hesaff-star	0.547	hesaff-orb	0.501	hesaff-brisk	0.520
	colsym-mser	0.429	colsym-har	0.384	colsym-star	0.457
	colsym-orb	0.410	colsym-brisk	0.405	mser-har	0.481
	mser-star	0.526	mser-orb	0.510	mser-brisk	0.507
	har-star	0.481	har-orb	0.458	har-brisk	0.479
	star-orb	0.456	star-brisk	0.481	orb-brisk	0.467

Table 4.2: Detector combinations and mAP^p using 'Kp-Distribution-Contribution-Cluster - mAP' model.

Dataset	Detector pair	mAP^p	Detector pair	mAP^p	Detector pair	mAP^p
Paris_DB	hesaff-colsym	0.500	hesaff-mser	0.517	hesaff-har	0.490
	hesaff-star	0.520	hesaff-orb	0.461	hesaff-brisk	0.486
	colsym-mser	0.521	colsym-har	0.496	colsym-star	0.538
	colsym-orb	0.418	colsym-brisk	0.453	mser-har	0.513
	mser-star	0.542	mser-orb	0.453	mser-brisk	0.519
	har-star	0.524	har-orb	0.468	har-brisk	0.513
	star-orb	0.482	star-brisk	0.533	orb-brisk	0.442

Table 4.3: Detector combinations and mAP^p using 'Distribution-Contribution - mAP' model.

When considering effective retrieval with the FII search engine on Paris_DB, the best performing detector pair is 'hesaff-mser' (with mAP = 0.593) compared to 'mser-star' with a mAP of 0.564. These results are presented and more deeply discussed later in this section in the Table 4.6.

Hence, the best-fitted model, 'Kp-Distribution-Contribution-Cluster - mAP', is selected for the prediction on the test datasets and for the following experiments.

4.5.3 Global prediction of the detectors combination performance

In this section, we present the prediction results of detector combinations using the linear regression model selected in the previous Sec. 4.5.2.

The model is trained with Paris_DB as described in Sec. 4.4.1. 'Kp-Distribution-Contribution-Cluster - mAP' model is used for prediction experiments on test datasets, *i.e.,* Oxford_DB and Holiday_DB based on the adjusted R^2 score. For the prediction on the test datasets, the procedure of Sec. 4.4.2 is applied by computing complementarity scores of the detector pairs.

The regression errors obtained for the test datasets, Oxford_DB and Holiday_DB, are 3.9% and 5.6% respectively as presented in the Table 4.4. We also generated a set of random data of size similar to the one of Paris_DB in order to qualify the regression error with the test datasets. The regression error for random data is 19.3% which is quite high compared to test datasets. This confirms us that the errors obtained on the test datasets are acceptable. With the help of these prediction results, detector pairs are selected for image retrieval experiments.

Dataset	Regression error
Oxford_DB	3.9%
Holiday_DB	5.6%
Random data	19.3%

Table 4.4: Regression error for different image datasets.

The predictions of the detector pairs are presented in the Table 4.5, with associated mAP^p. The highlighted table cells represent the best achieved predicted mAP. We observe that detector pairs, 'hesaff-har' for Oxford_DB and 'hesaff-mser' for Holiday_DB are associated with the best predicted mAP (mAP^p) which was trained with Paris_DB. Thus, we consider them as the best combinations for the image retrieval on these datasets.

4.5.4 Effective performance for image retrieval

In this section, the image retrieval results, measured with mAP, using FII search engine for the training dataset (Paris_DB) and the test datasets (Oxford_DB and Holiday_DB), are presented. Here, mAP retrieval results with FII is denoted by effective mAP (mAP^e). We selected top three performing detector combinations according to the prediction Tables 4.2 and 4.5 for image retrieval. We also selected the worst performing combination, *i.e.,* 'har-colsym' for Paris_DB, 'colsym-orb' for Oxford_DB and Holiday_DB. Additionally, another detector com-

Dataset	Detector pair	mAP^p	Detector pair	mAP^p	Detector pair	mAP^p
Oxford_DB	hesaff-colsym	0.501	hesaff-mser	0.537	hesaff-har	0.615
	hesaff-star	0.524	hesaff-orb	0.503	hesaff-brisk	0.523
	colsym-mser	0.482	colsym-har	0.554	colsym-star	0.459
	colsym-orb	0.360	colsym-brisk	0.504	mser-har	0.579
	mser-star	0.492	mser-orb	0.465	mser-brisk	0.527
	har-star	0.575	har-orb	0.561	har-brisk	0.459
	star-orb	0.441	star-brisk	0.504	orb-brisk	0.492
Holiday_DB	hesaff-colsym	0.442	hesaff-mser	0.461	hesaff-har	0.427
	hesaff-star	0.450	hesaff-orb	0.392	hesaff-brisk	0.441
	colsym-mser	0.402	colsym-har	0.415	colsym-star	0.354
	colsym-orb	0.338	colsym-brisk	0.413	mser-har	0.400
	mser-star	0.395	mser-orb	0.376	mser-brisk	0.415
	har-star	0.409	har-orb	0.420	har-brisk	0.3815
	star-orb	0.389	star-brisk	0.400	orb-brisk	0.424

Table 4.5: Detector combinations and predicted mAP using 'Kp-Distribution-Contribution-Cluster - mAP' model for test datasets.

bination, 'star-brisk' for Paris_DB, 'har-star' for Oxford_DB and 'mser-star' for Holiday_DB, which are approximately in the middle in the prediction sequence.

In the Table 4.6, the effective mAPs for Paris_DB are presented. The highlighted table cells represent the best achieved results.

Dataset	Detector pair	k-NN	mAP^e
Paris_DB	hesaff-mser	2	0.589
	hesaff-star	2	0.570
	mser-star	2	0.564
	star-brisk	2	0.496
	har-colsym	2	0.371

Table 4.6: Effective mAP (mAP^e) of detector pairs using the FII search engine for Paris_DB dataset.

According to the prediction (see Table 4.2), the best predicted combination is 'hesaff-mser'. The best effective mAP (mAP^e), *0.589*, is also obtained with 'hesaff-mser' followed by 'hesaff-star'. As predicted, the worst mAP^e is obtained with 'har-colsym'.

For Oxford_DB, which was trained with Paris_DB, the best effective result should be obtained with 'hesaff-har' pair (see Table 4.5). Indeed, the highest mAP^e is achieved with this combination (see Table 4.7). Also, the mAP^e of 0.269 is achieved with 'colsym-orb' which is the worst performing combination.

Dataset	Detector pair	k-NN	mAP^e
Oxford_DB	hesaff-har	2	0.549
	mser-har	2	0.456
	har-star	2	0.450
	har-star	2	0.450
	colsym-orb	2	0.269

Table 4.7: Effective mAP (mAP^e) of detector pairs using the FII search engine for Oxford_DB dataset.

Dataset	Detector pair	k-NN	mAP^e
Holiday_DB	hesaff-mser	2	0.683
	hesaff-star	2	0.666
	hesaff-colsym	2	0.643
	mser-star	2	0.535
	colsym-orb	2	0.499

Table 4.8: Effective mAP (mAP^e) of detector pairs using the FII search engine for Holiday_DB dataset.

For Holiday_DB, even if all the effective mAPs, as presented in the Table 4.8, are not in the same range of the predicted ones, we made the experiments with the detector combinations of the sorted sequence of the predicted mAP, and we can confirm that the effective mAP of the sorted sequence reflects the retrieval results. The highest mAP^e is obtained with 'hesaff-mser' combination followed by 'hesaff-star' and 'hesaff-colsym'. The 'mser-star' combination is in between the best and the worst performing combination according to the prediction (see Table 4.5) and the same sequence was reflected with effective results (with mAP^e of 0.535).

This first set of experiments confirms us that the spatial complementarity scores, employed with the linear regression model, are able to correctly estimate the performance of a detectors combination for image retrieval, then to enable the use of the best detector combination for a given dataset.

4.5.5 Comparison with state-of-the-art fusion approaches

In this section, we compare our results with the two state-of-the-art fusion approaches, *i.e.,* late fusion (LF) [Neshov, 2013] and early fusion based on feature concatenation (CBoF) [Yu et al., 2013] which are already exploited in the experiment section of Chapter 3.

The results are presented in the Table 4.9. We selected the two best performing detector pairs for LF and CBoF approaches for each dataset. The comparison results demonstrate that our proposed detector combination selection approach and then FII image retrieval outperforms the other two fusion retrieval approaches.

Dataset	Detector pair	k-NN	mAP^e		
			LF [Neshov, 2013]	CBoF [Yu et al., 2013]	FII
Paris_DB	hesaff-mser	2	0.541	0.283	0.589
	hesaff-star	2	0.535	0.241	0.570
Oxford_DB	hesaff-har	2	0.450	0.254	0.549
	mser-har	2	0.334	0.247	0.456
Holiday_DB	hesaff-mser	2	0.630	0.398	0.683
	hesaff-star	2	0.599	0.386	0.666

Table 4.9: Comparison with state-of-the-art fusion approaches for all datasets.

Paris_DB			Oxford_DB			Holiday_DB		
Detector	k-NN	mAP^e	Detector	k-NN	mAP^e	Detector	k-NN	mAP^e
hesaff	2	0.546	hesaff	2	0.498	hesaff	2	0.646
mser	2	0.523	har	2	0.421	mser	2	0.505

Table 4.10: Effective mAP of single detector using the FII search engine for the different datasets.

In Table 4.10, the retrieval results with single detectors are presented in order to compare with detector pair results of Tables 4.6, 4.7, and 4.8. We observe that image retrieval using detector pair outperforms using single detector retrieval.

These sets of experiment results demonstrate the relevance of the use of several complementary detectors in the representation of the image content.

4.5.6 Effect of k-NN parameter on retrieval and its prediction

The FII image search engine (discussed in Sec. 3.4 of Chapter 3) is designed to combine multiple local image features based on codebook and inverted multi-indices structure. The k-NN retrieval of the nearest neighbors is one of the central steps in FII search engine. The nearest neighbors search concerns here the retrieval of the k similar points to the query point. The optimal value of k is not easy to determine because it concerns the retrieval of similar points, and this value cannot be predetermined intuitively or learned easily. In general, the problem is addressed by fixing k value for the whole dataset, after having tested the retrieval performances with different values.

First, we present retrieval results in the Tables 4.11, 4.12, and 4.13 by varying k ($k = 2, 5, 10$), and observe the consequence on mAP^e. We selected the two best performing detector pairs for each dataset.

The effective mAP results are presented in Table 4.11. We observed that the best mAP^e is globally obtained with $k = 2$ for the two detector pairs. The accuracy difference is 1.8%

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Paris_DB	hesaff-mser	0.589	0.571	0.531
	hesaff-star	0.570	0.544	0.512

Table 4.11: Effective mAP for Paris_DB by varying k-NN (k=2,5,10).

between $k=2$ and $k=5$ and 5.8% between $k=2$ and $k=10$ for 'hesaff-mser' in Paris_DB. Also, the difference is 2.6% between for $k=2$ and $k=10$ 'hesaff-star' combination.

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Oxford_DB	hesaff-har	0.549	0.547	0.533
	mser-har	0.456	0.430	0.420

Table 4.12: Effective mAP for Oxford_DB by varying k-NN (k=2,5,10).

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Holiday_DB	hesaff-mser	0.683	0.677	0.670
	hesaff-star	0.666	0.661	0.650

Table 4.13: Effective mAP for Holiday_DB by varying k-NN (k=2,5,10).

The similar trend is observed in Oxford_DB and Holiday_DB as presented in the Tables 4.12 and 4.13. The difference is 1.6% and 3.6% between $k=2$ and $k=10$ for 'hesaff-har' and 'mser-har' correspondingly. In the Holiday_DB, the accuracy difference is 0.6% between $k=2$ and $k=2$ and 1.3% between $k=2$ and $k=10$ for 'hesaff-mser'.

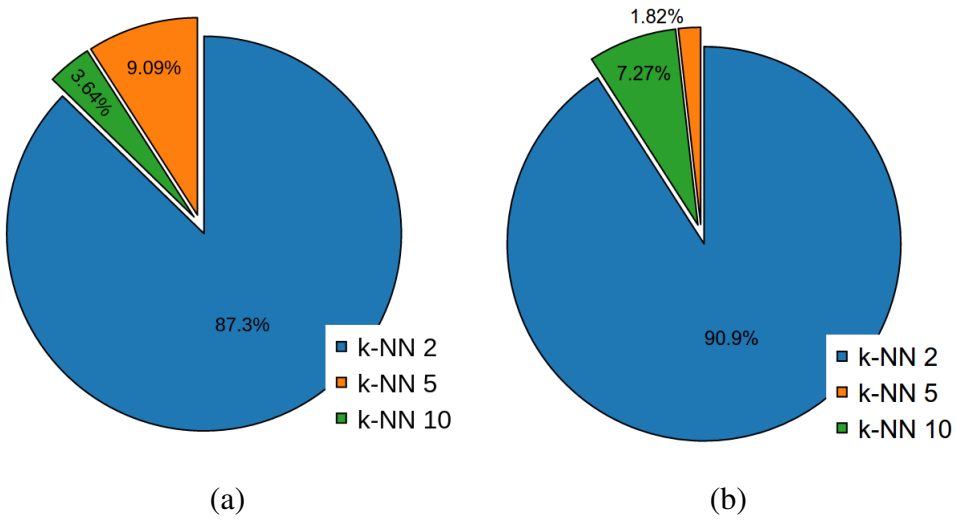
Nevertheless, we observed through dedicated experiments that the optimal value of k may be different from the general trend for some queries. During the search for the nearest neighbors of the query point, sometimes higher values of k might include more similar or dissimilar neighbors in the k-NN lists. In the Sec. 4.5.8, we present an image retrieval example (see Fig. 4.17) to emphasize this scenario.

In the following, we experiment the capability of our model to adapt the best value of k *for each query* instead of estimating it globally for the dataset. Instead of choosing a fixed k-NN value for all query images, we choose a different k for each image. The procedure of Sec. 4.4 is applied by varying k ($k=2,5,10$) and the predicted mAP obtained allows to adapt k to each query.

In the Tables 4.14, 4.15, and 4.17, the highlighted mAPs correspond to the mAP^e obtained by adapting k to each query. The other mAPs correspond to k-NN values ($k=2,5\&10$) fixed for all the queries.

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Paris_DB	hesaff-mser	0.589	0.571	0.531
		0.591 (adaptive k 2,5 & 10)		
Paris_DB	hesaff-star	0.570	0.544	0.512
		0.572 (adaptive k 2,5 & 10)		

Table 4.14: Effective mAP for Paris_DB by adapting k-NN (k=2,5,10).



(a) (b)

Figure 4.11: Distribution of predicted k values across the queries for Paris_DB, (a) hesaff-mser combination (b) hesaff-star combination.

The increment of mAP^e is 0.2% for both 'hesaff-mser' and 'hesaff-star' for Paris_DB (see Table 4.14) compared to the previous best obtained with $k = 2$. Next, we observe, in Fig. 4.11, the distribution of the k selected adaptively across the queries for Paris_DB. The majority of the best results are associated with $k = 2$, in relation with the best mAP^e obtained, followed by $k = 5$ and $k = 10$. Approximately 87% of the queries are selected with $k = 2$ for 'hesaff-mser' followed by $\sim$9% from $k = 5$ and $\sim$4% from $k = 10$. Similarly, $\sim$91% are selected from $k = 2$, followed by $k = 10$ and $k = 5$ for 'hesaff-star' combination.

For Oxford_DB, the mAP^e is increased by 1.8% and 0.4% for 'hesaff-har' and 'mser-har' respectively (see Table 4.15) compared to the previous best ones with $k = 2$.

To observe how the mAP is affected with the varying k-NN values, we selected some particular queries from Oxford_DB and present the retrieval mAPs in the the Table 4.16. Although, $k = 2$ performs better globally for 'hesaff-har' combination, but we observe for some queries $k = 5$ or $k = 10$ performs better than $k = 2$, which is predicted by our prediction model. Hence, the prediction strategy helps to increase the accuracy. We also observe, from the Fig. 4.12, that

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Oxford_DB	hesaff-har	0.549	0.547	0.533
		0.567 (adaptive k 2,5 & 10)		
Oxford_DB	mser-har	0.456	0.430	0.420
		0.460 (adaptive k 2,5 & 10)		

Table 4.15: Effective mAP for Oxford_DB by adapting k-NN (k=2,5,10).

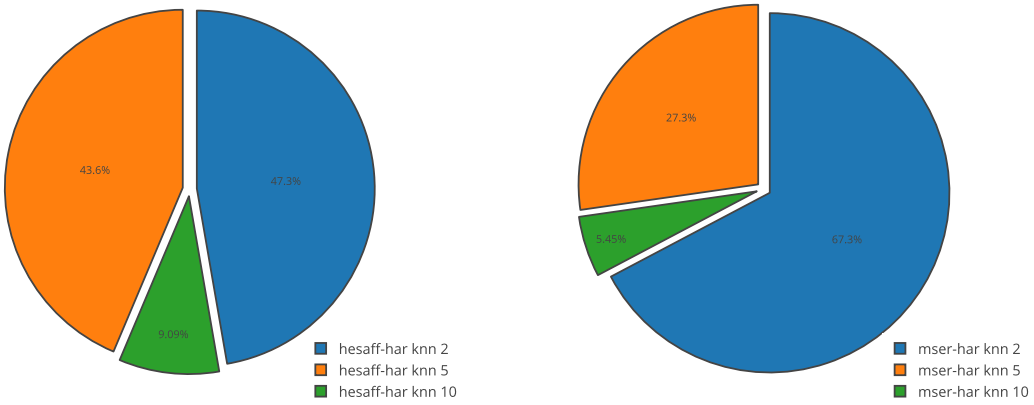
Dataset	Retrieval mAP		
	$k=2$	$k=5$	$k=10$
Oxford_DB	0.447	0.517	0.515
	0.760	0.729	0.673
	0.746	0.794	0.763
	0.614	0.676	0.644
	0.779	0.812	0.817
	0.705	0.735	0.740
	0.807	0.835	0.716
	0.561	0.498	0.504
	0.514	0.510	0.467
	0.469	0.383	0.418

Table 4.16: Retrieval mAP image-by-image using 'hesaff-har' combination with varying k-NN values for Oxford_DB.

47% of the queries are executed with $k = 2$, 44% with $k = 5$, and only 9% with $k = 10$ for 'hesaff-har combination. It is interesting to observe the balanced distribution between $k = 2$ and $k = 5$ for 'hesaff-har' combination (see Fig. 4.12(a)), which conduces to a clear global improvement (mAP^e from 0.549 to 0.567) by considering queries optimized with $k = 5$.

As expected, the mAP^e is also improved for Holiday_DB. mAP^es are increased by 0.8% and 0.5% for 'hesaff-mser' and 'hesaff-star' as presented in the Table 4.17. The distribution of the selected k-NN values are depicted in the Fig. 4.13. We observe, that ∼90% and ∼87% of them are selected with $k = 2$ followed by $k = 5$ and $k = 10$.

These results particularly highlight the importance of the automatic selection of k-NN value for each query.



(a) (b)

Figure 4.12: Distribution of predicted k values across the queries for Oxford_DB, (a) hesaff-har combination (b) mser-har combination.

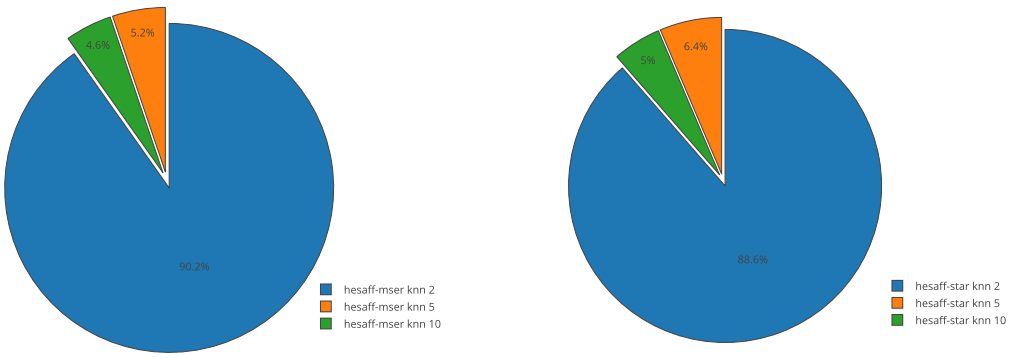
Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
Holiday_DB	hesaff-mser	0.683	0.677	0.670
		0.691 (adaptive k 2,5 & 10)		
Holiday_DB	hesaff-star	0.666	0.661	0.650
		0.671 (adaptive k 2,5 & 10)		

Table 4.17: Effective mAP for Holiday_DB by adapting k-NN (k=2,5,10).

4.5.7 Image-by-image prediction of the best detector combination

In this section, we refine the retrieval results obtained in Secs. 4.5.4 and 4.5.6 by adapting the selection of the best detector combination *to each image*, by applying the prediction strategy of Sec. 4.4 for detectors combination, to each query image instead of globally on the whole dataset. Six different combinations of mAP^e obtained with two best detector pairs and three k-NN value ($k = 2, 5, 10$) are consolidated.

In Table 4.18, we observe that mAP^e is increased by 0.8%, 2.5% and 21.1% compared to previous best ones with $k = 2$ for Paris_DB, Oxford_DB and Holiday_DB respectively. For Holiday_DB, the achieved retrieval accuracy is 0.894, which is one of the best in the state of the art to our knowledge, compared to Ref. [Li et al., 2015]. The approach in the work of [Li et al., 2015] is based on bag of words and also tested on the Holiday_DB and achieved an accuracy of 0.892.



(a) (b)

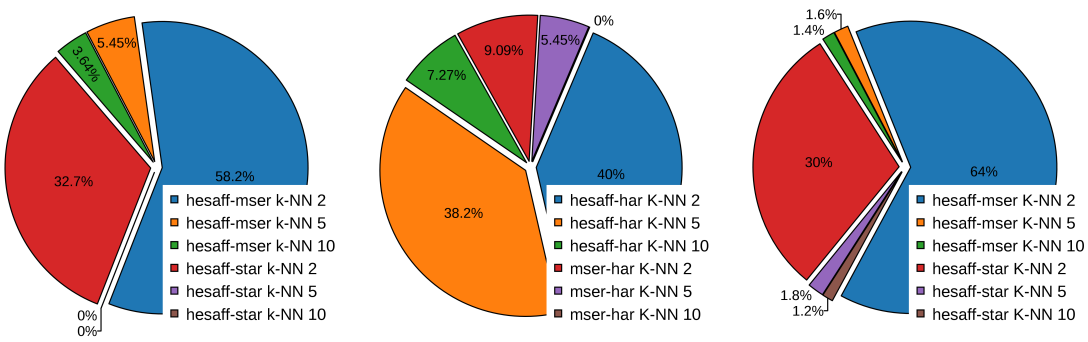
Figure 4.13: Distribution of predicted k values across the queries for Holiday_DB, (a) 'hesaff-mser' (b) 'hesaff-star'.

Dataset	Detector pair	mAP^e		
		$k = 2$	$k = 5$	$k = 10$
Paris_DB	hesaff-mser	0.589	0.571	0.531
	hesaff-star	0.570	0.544	0.512
	Adaptive detector combination	0.597 (Adaptive k 2,5 & 10)		
Oxford_DB	hesaff-har	0.549	0.547	0.533
	mser-har	0.456	0.430	0.420
	Adaptive detector combination	0.574 (Adaptive k 2,5 & 10)		
Holiday_DB	hesaff-mser	0.683	0.677	0.670
	hesaff-star	0.666	0.661	0.650
	Adaptive detector combination	0.894 (Adaptive k 2,5 & 10)		

Table 4.18: Effective mAP obtained for all the datasets, by selecting the optimal detector pair and the optimal value k for each query image.

As depicted in Fig. 4.14, the majority of the selections are done with $k = 2$ for all datasets. For Paris_DB, 90.9% are selected for $k = 2$ of both pairs of detectors, while 5.49% are with $k = 5$. Approximately 67% of the queries are executed with the best performing 'hesaff-mser' combination, while ∼33% are selected from 'hesaff-star' combination.

As anticipated, the statistical observation for Oxford_DB (see Fig. 4.14) indicates that the majority of the queries are selected with $k = 2$ of 'hesaff-har' and 'mser-har' pairs . Approx-



(a) (b) (c)

Figure 4.14: Distribution of predicted values of k and detectors pairs across the queries, (a) 'hesaff-mser' & 'hesaff-star' for Paris_DB. (b) 'hesaff-har' & 'mser-har' for Oxford_DB (c) 'hesaff-mser' & 'hesaff-star' for Holiday_DB.

imately 49% are selected from $k = 2$, while only 7% are selected from $k = 10$. Also most of the mAP^e, *i.e.,* 85% are from 'hesaff-har' pair while remaining 15% are from 'mser-har' pair.

Similar trend is observed with Holiday_DB, where 67% of the queries are taken from 'hesaff-mser' and 33% from 'hesaff-star' combination. Also for Holiday_DB, 94% are selected from $k = 2$ of both the detector pairs, 3.4% with $k = 5$ and remaining 2.6% form $k = 10$.

Even if the statistical analyses (Figs. 4.11, 4.12, 4.13 and 4.14) have highlighted the dominance of some particular detectors pairs and values of k, we observe that exploiting other ones adaptively for each query allows one to refine the results notably. In the example depicted in the Fig. 4.15, we explain the importance of the adaptive selection of the detector combination. More generally, these experiments also clearly highlight the impact of the spatial complementarity of the selected features on the retrieval performance.

4.5.8 Image retrieval examples

In this section, we present image retrieval examples from different datasets which are used in our experiments. The queries are executed with different detector combinations and different parameter configurations of FII search engine within the scope of the regression based image retrieval framework.

Image retrieval examples from Holiday_DB are depicted in the Fig. 4.15. In this example, there are three relevant images which are present in the dataset for the query according to the ground truth. Globally, 'hesaff-mser' is the best performing detector combination for Holiday_DB according to the prediction (see Sec. 4.5.3), as well as the effective retrieval results presented in the Table 4.8. However, when we apply the adaptive selection of the detector combination for each image, the best performing combination is 'hesaff-star' instead of globally best 'hesaff-mser' combination for this query image depicted in the Fig. 4.15. We observe that, 'hesaff-star'

allows to retrieve relevant three images of query at the top three position (see 4.15(b)); while 'hesaff-mser' allows retrieving two relevant images at the beginning and another relevant image at the 5^{th} position (see 4.15(a)). Therefore, this example emphasizes the importance of the adaptive selection of the detector combination image by image, which leads to better retrieval accuracy.

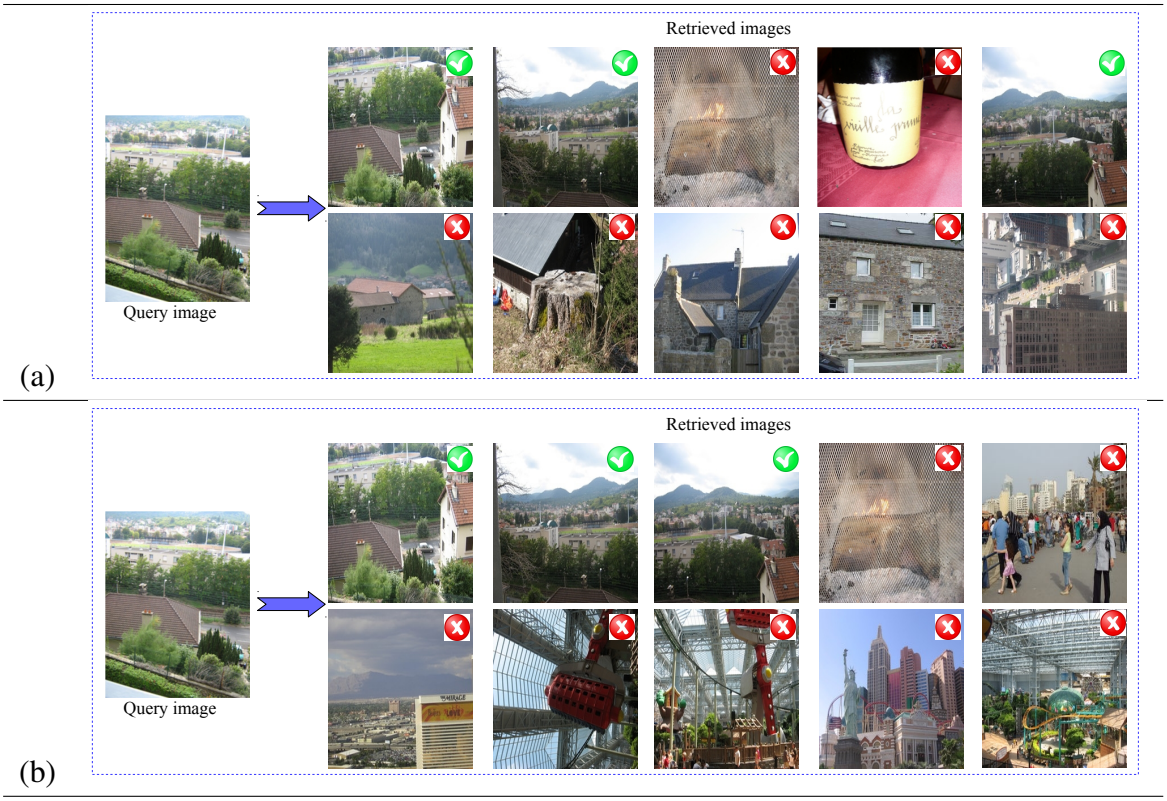


Figure 4.15: The 10 first retrieved results by decreasing order of similarity, from left to right and top to bottom with the FI' search engine using two different combinations of detectors for Holiday_DB: (a) 'hesaff-mser' (b) 'hesaff-star'.

An image retrieval example of Oxford_DB is presented in the Fig. 4.16. The query is executed with 'mser-har' combination. Globally, 'mser-har' combination of Oxford_DB performs better with $k = 2$ as shown in the Table 4.12. This trend is ascertained in the example shown in the Fig. 4.16. The best performing k-NN value is $k = 2$ for this particular query, followed by $k = 5$ and $k = 10$, which was also reflected on the prediction mAP. We observe that 8 retrieved images are relevant out of top ten retrieved images with $k = 2$ as depicted in the Fig. 4.16(a). When the same query is executed with $k = 5$ and $k = 10$ configurations, the retrieval accuracy is decreased; $k = 5$ retrieved 7 and $k = 10$ retrieved 6 relevant images in 10 first retrieval.

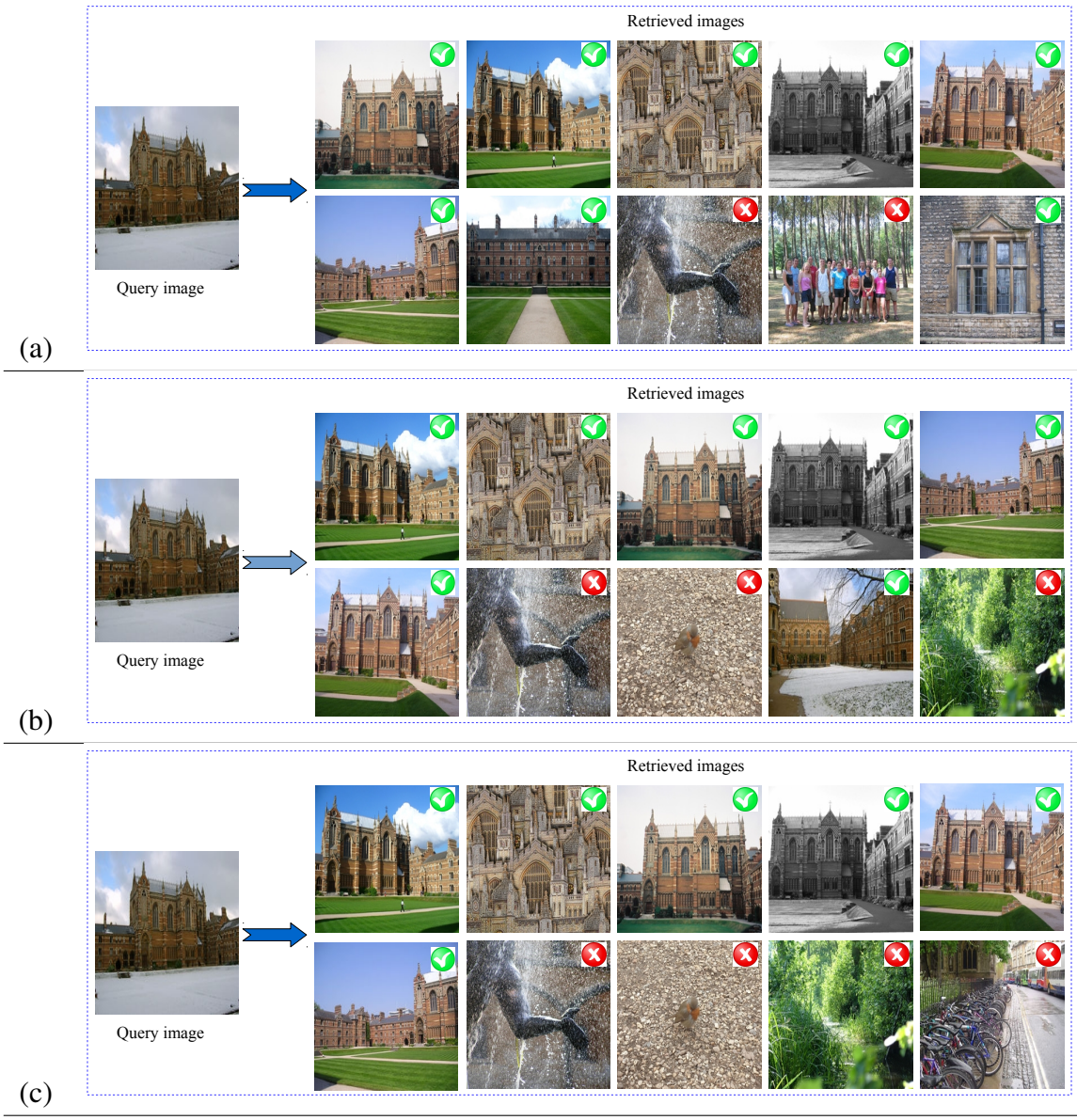


Figure 4.16: The 10 first retrieved results by decreasing order of similarity, from left to right and top to bottom with the FII search engine using 'mser-har' combination and varying k-NN value for Oxford_DB: (a) $k = 2$ (b) $k = 5$ (c) $k = 10$.

The example depicted in the Fig. 4.17, we give an insight on the adaptive selection of the k-NN values for a query image. For Oxford_DB, image retrieval with 'hesaff-har' combination using $k = 2$ has a slight advantage compared to $k = 5$ globally. This analysis was depicted in the Fig. 4.12(a) of Sec. 4.5.6. However, the query image in the example shown in Fig. 4.17, the retrieval accuracy is better when it is executed with $k = 5$, instead of $k = 2$. With $k = 2$, 8 retrieved images are relevant out of top ten retrieved images as shown in the Fig. 4.17(a), while 9 images are correctly retrieved with $k = 5$ for the same query (see Fig. 4.17(b)).

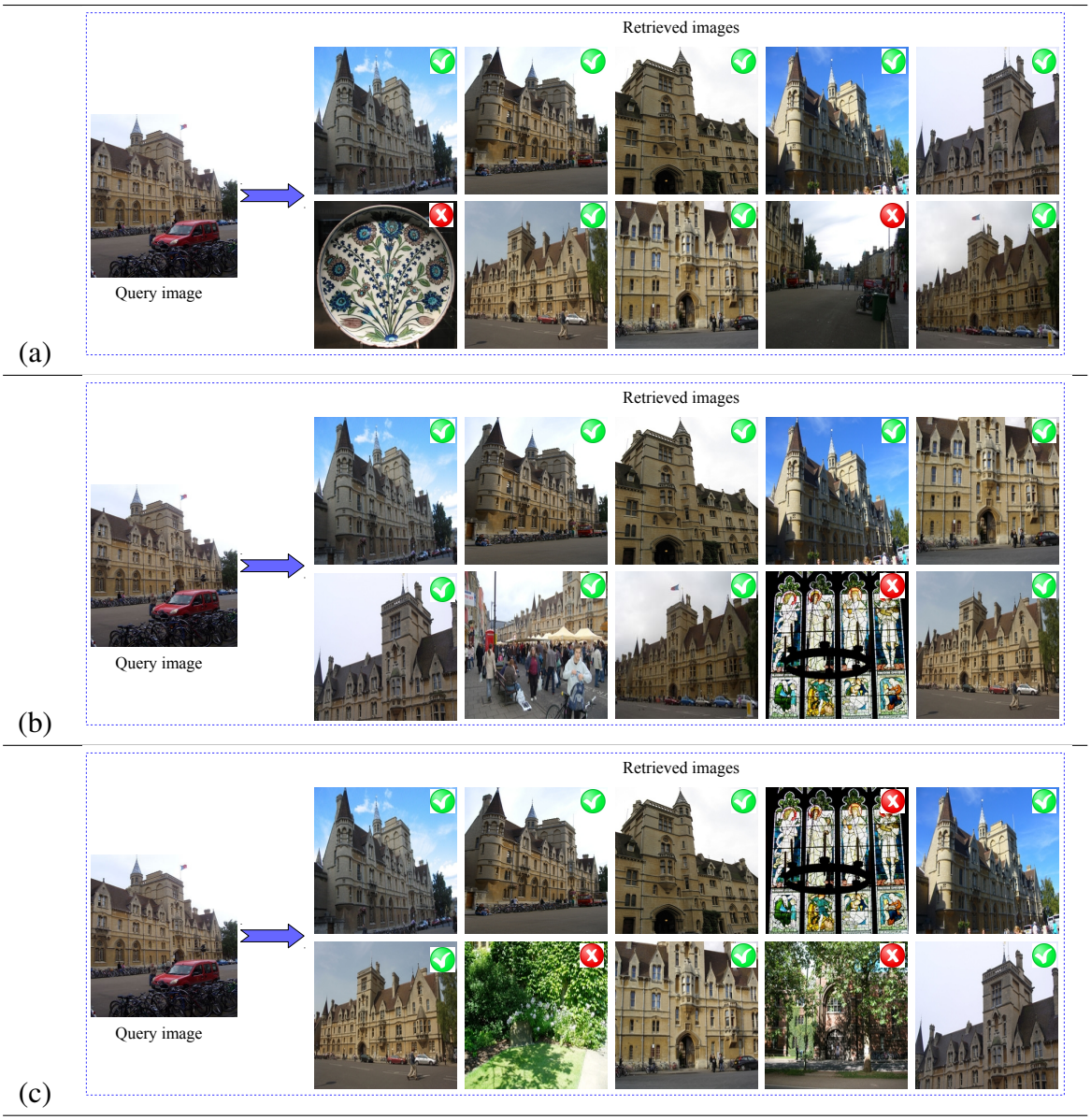


Figure 4.17: The 10 first retrieved results by decreasing order of similarity, from left to right and top to bottom with the FII search engine using 'hesaff-har' combination and varying k-NN value for Oxford_DB: (a) $k = 2$ (b) $k = 5$ (c) $k = 10$.

4.6 Conclusions

The idea behind the work presented in this chapter is how to adopt the suitable feature detectors, which can provide an adequate representation of a particular image dataset. The main contribution of the work presented in this chapter is the proposal of a regression model based on several spatial complementarity criteria between local image features, in order to estimate the optimal combination of detectors for the description of a given image, within the scope of query-by-example image retrieval. In that way, we can make a prediction on the best suitable feature detector combinations which might provide the best similarity search results.

Even if the statistical analyses presented to highlight the dominance of some particular detector combinations (and values of k), we observed that using other ones adaptively - for some images - allows refining the results favorably and notably. The proposal is appraised on three state-of-the-art datasets to validate its effectiveness and stability. The experimental results highlight the importance of spatial complementarity of the features to improve retrieval and prove the advantage of using this model to optimally adapt detectors combination and some parameters. Facing state of the art strategy [Li et al., 2015], we have demonstrated our proposal provides better results, on dataset Holiday_DB.

The higher complementarity scores imply a more distinctive representation of the content. The proposed framework can effectively reduce the overall experimental time by narrowing down the choice of detectors, and the adaptive selection of some parameters, such as k during the nearest neighbor retrieval, improves, even more, the retrieval accuracy.

In the Chapter 5, where applications of this work are presented, we will highlight again the relevance of the proposal, by considering image datasets of very diverse contents, where the adaptive selection of detectors and parameters such as k clearly brings distinctiveness to the representation.

Chapter 5

Cross-domain image retrieval

5.1 Introduction

With the growing acquisition of contents in various professional and general public domains, cross-domain image retrieval is a topical subject. It questions the problem of comparing, indexing and searching for contents potentially acquired by different sources or modalities, such as different cameras, paintings, sketches, street views, etc. and at different times. In the Chapter 3, we have proposed the fusion of inverted indices (FII) image search engine. In following Chapter 4, we have put forward a regression based model, incorporating in FII image search engine, to envisage the optimal integration of features for refining retrieval performances. In this chapter, we make use of the previous proposals to deal with the problem of cross-domain image retrieval, which is a challenging task for images across different domains. Additionally, two applications of cross-domain image retrieval are explored. First, we present cross-domain retrieval for image exploration in museum image collection. Second, we explore the topic of image-based localization, where the pose of a landmark is estimated from geo-localized reference images that visually match the query images, which are acquired under various conditions, such as old photographs, paintings, photos taken at a particular season, etc. The framework is evaluated on different datasets and the experiments prove its advantage over classical retrieval approaches.

To begin with, let us exemplify what is cross-domain retrieval. Cross-domain image retrieval is the process to retrieve visually similar images where the query images are different in characteristics from the database images. In other words, the query images belong to one domain, such as painting, postcards, sketches, etc. and the querying image database is from a different domain, such as street view, a camera captured images, digitally scanned images, etc. in cross-domain. An example of cross-domain contents is depicted in the Fig. 5.1.

As we are living in a digital era, the cross-domain image retrieval becomes one of the challenging tasks thanks to the leaps and bounds growing in the multimedia data acquisition by

Figure 5.1: Corss-domain examples.

museums, content sharing/storing companies, etc. In the last decades, several museums and archive companies have started image digitization on a large scale. At the same time, the rapid development of production of the digital images confronts professional users and individuals with a huge number of images which are difficult to exploit due to the volume. Not only that, image datasets are becoming more and more complex due to the diverse contents of the images and the images belong to different domains, making their organization an essential stage for any datasets. When we consider the example of photographic museums (illustrated in Fig. 5.2 with photographs from Musée Nicéphore Niépce[1]), the image indexing, structuring and retrieval steps are mostly done manually.

These steps are strongly influenced by several factors such as the use of the database, cultural background of the archivist and his professional references. The cultural and professional references of the archivists instigate the standard of indexing and thesaurus used in the museums which are unknown to the general public. Enhancement of digital or scanned images through, for example, publishing or through the organization of an exhibition, creates numerous exchanges between interlocutors of different professions, each one having its respective needs (curator, editor, museum curator, picture editor, archivist). Today, there are no common tools that meet the needs of all potential actors, so as to facilitate a collaborative work.

When preparing an exhibition, the selection of photographs is usually driven by the registrar or the curator of the exhibition; a first filter takes place. Archivists extract from keywords photographs that might be of interest to the commissioner, thereby constituting a second filter. In the case of an exhibition project involving several institutions, these factors are thereby increased. Additionally, managing different databases and virtual sharing of iconographic collections are made difficult by the disparity of standards, the language barriers, the control of software used and the management of documentary parasites inherent in any search by keywords. Finally, based on lexical fields of language backgrounds which are different from country to country,

[1]Nicéphore Niépce museum website: http://en.museeniepce.com

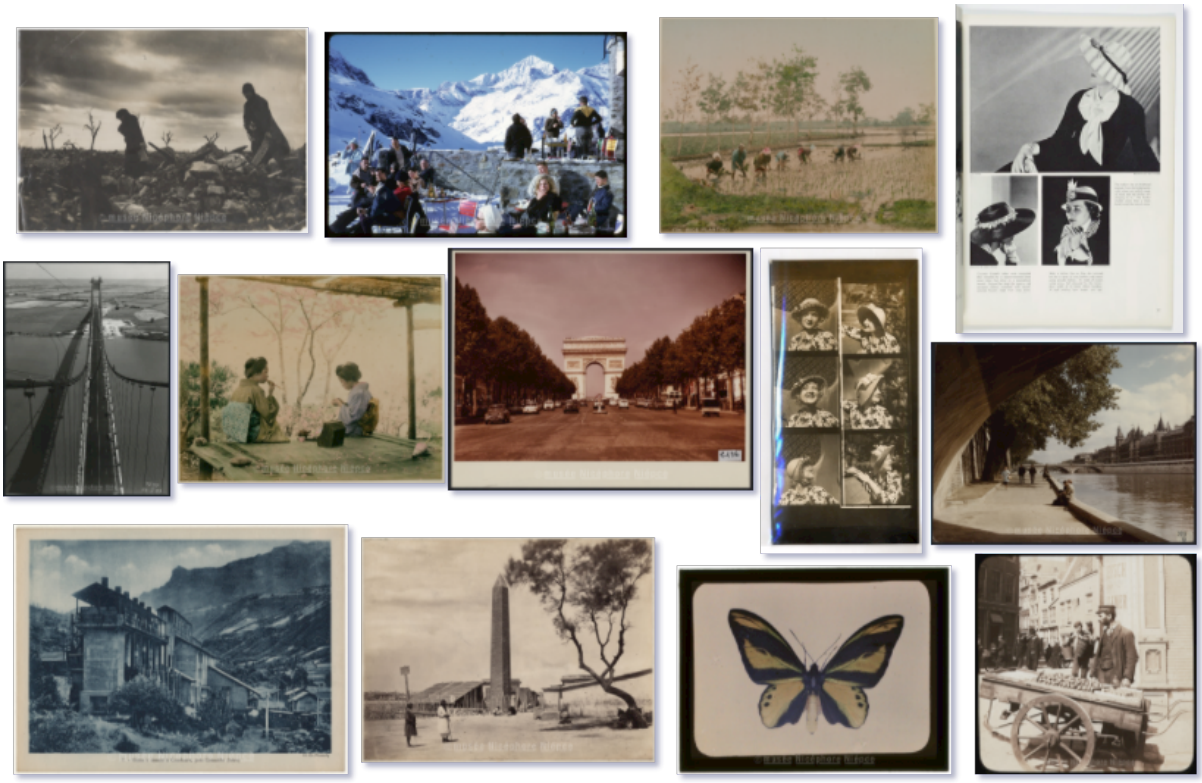


Figure 5.2: Examples of digitized photographs from the archives of the Musée Nicéphore Niépce, France. The museum has a variety of image contents from different sources.

indexing and searching for images in a database may often be exclusive to one country.

Once established, the collection is then exhibited to the general public according to a given spatial organization when in situ, and virtual exhibitions usually consist in interacting via a website with the collection by selecting categories or keywords, looking at photographs through lists, thumbnails or slideshows.

Although several strategies for CBIR can be found in the literature over the period time, the cross-domain retrieval problem is grossly over looked. We presented a literature review on CBIR in the Chapter 2. Few recent cross-domain retrieval strategies are proposed in the literature [Shrivastava et al., 2011; Huang et al., 2015a; Wang et al., 2016; Russell et al., 2011]. In the work of [Shrivastava et al., 2011], a 'data-driven uniqueness' method is proposed. In this method, the best discriminative elements/features in the images are considered and accordingly the weights are assigned. The HOG feature is used to represent the images. Recently, we have witnessed the use of deep learning based approaches [Wan et al., 2014; Hoffman et al., 2013; Chen et al., 2015; Gopalan et al., 2011; Babenko et al., 2014; Chopra et al., 2013] to solve CBIR problem. However, cross-domain retrieval is not considered in the most of the deep learning based approaches except few. A deep learning strategy is proposed by [Wang et al., 2016] for cross-domain retrieval in between sketch and natural images. The deep neural network is trained by mixing natural images and the sketches. [Huang et al., 2015a] proposed a dual attribute-aware ranking network for cross-domain application of clothing image retrieval

from online shopping stores. This method used both semantic attribute information, such as cloth color, shape, style, and visual similarity constraints during the learning stage. With similar ideas on the matching of cross-domain features, [Aubry et al., 2014] proposed a technique to align 2D descriptions of architectural site images with the 3D model. In the context of artwork and cultural heritage retrieval, different strategies are proposed in the literature, such as in [Tsai, 2007; Jiang et al., 2005]. For cultural heritage retrieval, [Vrochidis et al., 2009] proposed a hybrid multimedia retrieval model which combines low-level visual features based retrieval and semantic annotation retrieval on finding the similar images. Reference [Yen et al., 2006] developed an image retrieval system based on AdaBoost [Freund and Schapire, 1997] and relevance feedback for painting image retrieval. Several strategies for classification of cultural heritage buildings, events are addressed in the works of [Obeso et al., 2017; Shalunts et al., 2011; Chu and Tsai, 2012; Salvador et al., 2015]. [Obeso et al., 2017] used deep convolution networks and sparse features to classify Mexican historical buildings. In the work of [Salvador et al., 2015], CNN features were used in SVM classifier for cultural event classification. Clustering of local feature (*e.g.,* SIFT) was proposed in the work of [Shalunts et al., 2011] for classification of different architectural facade windows. SCULPTEUR [Goodall et al., 2004] and MIRS [Mahdi and Ibadi, 2014] are two tools for museum multimedia retrieval. SCULPTEUR is designed for 3-D and 2-D content retrieval, while MIRS used several low levels visual features individually, such as color and texture features to retrieve similar images. However, most of the approaches discussed mostly consider a single feature rather than combining multiple features.

Therefore, we are interested in addressing the cross-domain image retrieval problem using FII image search engine. This will be useful in the context of museum collection in order to organize the voluminous and varied image datasets. Indexing, comparing and retrieving images by measuring the similarity of their content open very interesting perspectives when applied to museum uses. In addition to helping the archivist in automating the indexing step in large collections (*e.g.,* automatic propagation of keywords, automatic linking of similar contents, etc.) while minimizing the subjective and cultural factors cited above, this paradigm may provide new standard and personalized tools to the museum professionals and users, thus improving management of the collections by their experts as well as contributing to its enhancement close to the general public.

To begin with, the proposed framework is evaluated on the ParisCrossDomain benchmark, which is created as a part of this work, to prove its effectiveness. Once we establish the effectiveness of the framework, additional experiments are conducted on the museum image database for cross-domain applications, such as museum collection exploration and database indexing by intra-linking the image content within the scope of the French ANR project POEME[2]. The purpose of POEME is to design, prototype and implement an immersive environment integrates the collection of the museum and the content-based search engine. By the way of innovative

[2] www.agence-nationale-recherche.fr/en/anr-funded-project/?tx_lwmsuivibilan_pi2%5BCODE%5D=anr-12-cord-0031

and intuitive visual metaphors, it provides new visualization tools with the aims of better highlighting the responses provided by the content-based search engine and of enriching navigation in the collection. To make navigation and querying natural, an innovative and intuitive man/-machine interface is also be designed, based on dedicated interaction metaphors. The project target outcomes are technologies for creating personalized and engaging digital cultural or play experiences on digitalized photographs collections, in particular on the collections of the museum. Thus, the FII image search engine is being integrated with interactive immersive image exploration system to help the professionals of a French museum at different levels from the archiving up to the exhibition. We expand cross-domain retrieval to inter link the image content between museum collections and the public databases. Not only for the museum applications, the cross-domain retrieval for image localization application, *i.e.,* the pose estimation of a landmark from visually similar content by inter-linking image contents, is further explored in this chapter.

The rest of the chapter is organized as follows: Sec. 5.2 is dedicated to the experiments conducted for cross-domain image retrieval. It follows by the Sec. 5.3, which is dedicated to the evaluation and applications of FII search engine for the museum image collections exploration application. Section. 5.4 will present cross-domain image localization application before concluding in the Sec. 5.5.

5.2 Experiments and evaluation on cross-domain image retrieval

This section presents and discusses the experiments conducted for cross-domain image retrieval using proposed FII search engine with adaptive feature selection model.

The experiment section begins with the evaluation framework in the Sec. 5.2.1, where details about image datasets and parameter configurations are presented. It follows by the global prediction of the detector combinations experiments for the test dataset in the Sec. 5.2.2. Section 5.2.3 presents the effective retrieval performances of the test dataset using different detector combinations. Section 5.2.4 presents the retrieval performances with optimal configuration of the k nearest neighbor values. Image retrieval experimental results with an adaptive selection of the detector combinations with optimal k-NN values are presented in the Sec. 5.2.5. The section is wrapped up with cross-domain image retrieval examples in the Sec. 5.2.6.

5.2.1 Framework of evaluation

The experiments are conducted on the two image datasets mention below:

1. Paris_DB: This dataset is discussed in the Sec. 3.5.1 of Chapter 3. This dataset is used to

train the regression model.

2. ParisCrossDomain_DB (PCD_DB): This is a newly constructed dataset with approximately 6500 images. It consists of the Paris_DB[3] images and additional old images, such as paintings, postcards, of the popular Paris monuments. Old/modified Paris monument images are used as query. The samples from this dataset are depicted in the Fig. 5.3.

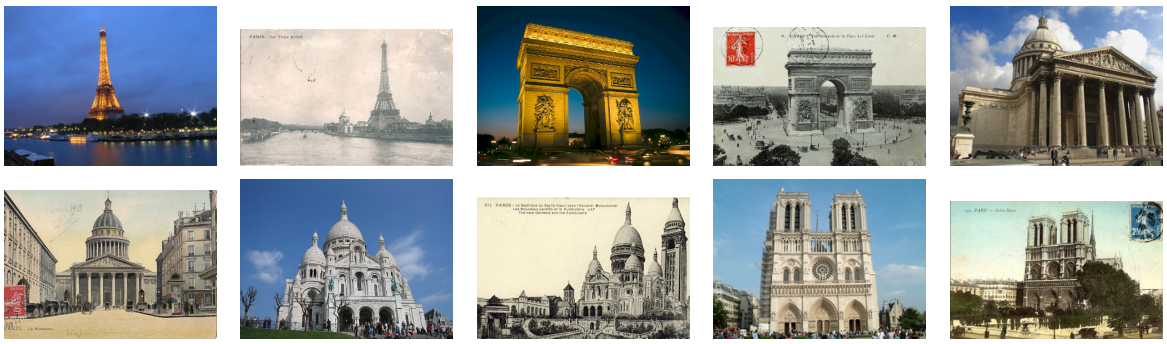


Figure 5.3: Sample images from the ParisCrossDomain dataset used in the experiments.

We have selected 6 detectors from characteristically diverse family such as corner, blob, symmetry, etc.: Hessian affine (hesaff) [Mikolajczyk and Schmid, 2004], color symmetry (colsym) [Heidemann, 2004], MSER (mser) [Matas et al., 2002], Harris (har) [Schmid and Mohr, 1997], Star (star) [Agrawal et al., 2008], and oriented and rotated BRIEF (orb) [Rublee et al., 2011]. Three local descriptors, such as SIFT [Lowe, 2004], SURF [Bay et al., 2008] and shape context (SC) [Belongie et al., 2002], are used to describe the extracted points and then used the descriptions jointly or individually in the fusion of inverted index (FII) search engine for image retrieval. Although we begin the experiments with the combination of these three descriptors, later we use single descriptor as well as the combinations of the two descriptors to compare the results. Performances are presented with mean Average Precision (mAP). Optimal codebook size used is $\sim$20% of the total description points of each detector combination. Parameter k is varied in between 2 to 10 for optimal configuration of the nearest neighbors.

5.2.2 Detector combination performances prediction for cross-domain retrieval

The prediction strategy based on spatial complementarity measures and regression model (see Sec. 4.4 of Chapter 4) is employed to predict the best feature combination for PCD_DB. The regression model is trained with Paris_DB. The results for the global prediction of the best suitable detector combinations are presented in the Table 5.1.

[3] http://www.robots.ox.ac.uk/~vgg/data/parisbuildings/

Detectors pair	mAP^p	Detectors pair	mAP^p
hesaff-colsym	0.497	hesaff-mser	0.535
hesaff-har	0.537	hesaff-star	0.549
hesaff-orb	0.512	colsym-mser	0.468
colsym-har	0.485	colsym-star	0.458
colsym-orb	0.375	mser-har	0.519
mser-star	0.494	mser-orb	0.465
har-star	0.521	har-orb	0.493
star-orb	0.461		

Table 5.1: Detector combinations and predicted mAP^p using 'Kp-Distribution-Contribution-Cluster - mAP' regression model for PCD_DB.

The detector combination 'hesaff-star' is associated with best predicted mAP (mAP^p), followed by 'hesaff-har' for PCD_DB; therefore these combinations are used for FII image retrieval in the following experiments.

5.2.3 Image retrieval effective performances

The effective image retrieval results of the PCD_DB are discussed in this section. The results of the best three predicted detector combinations are presented. We also select one of the worst predicted combination to validate the prediction results. According to the Table 5.1, the best performance (effective mAP) should be obtained with the 'hesaff-star' combination for PCD_DB. The retrieval results presented in Table 5.2. the highest effective mAP^e (=0.409) is obtained with 'hesaff-star' combination, followed by 'hesaff-har' combination. The worst mAP^e is obtained with 'colsym-orb' combination, which satisfies the prediction pattern. Thus, these experimental results affirm that the regression model is capable of reckoning the detector combination performance for cross-domain image retrieval.

Dataset	Detector pair	k-NN	mAP^e
PCD_DB	hesaff-star	2	0.409
	hesaff-har	2	0.398
	hesaff-mser	2	0.388
	mser-colsym	2	0.289
	colsym-orb	2	0.227

Table 5.2: Effective mAP using FII for PCD_DB.

The next set of experiments is performed to compare the results of Table 5.2 with a state-of-the-art late fusion (LF) image retrieval technique [Neshov, 2013]. The two best performing detector

combinations for LF retrieval are selected for comparison. The LF retrieval results, presented in the Table 5.3, demonstrate that the performance of our detector combination selection method is superior. Additionally, we observe that for PCD_DB with 'hesaff-star', FII outperforms LF for the first ten retrievals with an mAP of 0.656 (vs. 0.420). This is important for applications which require accurate retrieval at the top, such as image-based localization.

Dataset	Detector pair	k-NN	mAP^e	
			LF [Neshov, 2013]	FII
PCD_DB	hesaff-star	2	0.365	0.409
	hesaff-har	2	0.362	0.398

Table 5.3: Comparison of FII with the late fusion (LF) technique [Neshov, 2013] for PCD_DB.

To compare, the retrieval mAP^es (using FII) for a single detectorare exhibited in the Table 5.4. It is quite evident that the detector combinations performed better compared to single detector.

Dataset	Detector	k-NN	mAP^e
PCD_DB	hesaff	2	0.351
	star	2	0.287

Table 5.4: Effective mAP^e of single detector using FII for PCD_DB.

The combination of SIFT, SURF and SC descriptors is used in all the experiments conducted so far in this section. In the Table 5.5, we present more experimental results which are performed using the combinations od two descriptors and as well as using only a single descriptor. We select the two best combinations of detectors to extract interest points.

Dataset	Detector pair	Descriptors	k-NN	mAP^e
PCD_DB	hesaff-star	SIFT	2	0.251
		SURF	2	0.322
		SC	2	0.255
		SIFT-SC	2	0.309
		SIFT-SURF	2	0.349
		SIFT-SURF-SC	2	0.409
	hesaff-har	SIFT	2	0.301
		SURF	2	0.315
		SC	2	0.297
		SIFT-SC	2	0.328
		SIFT-SURF	2	0.337
		SIFT-SURF-SC	2	0.398

Table 5.5: Effective mAP^e of single detector using FII for PCD_DB.

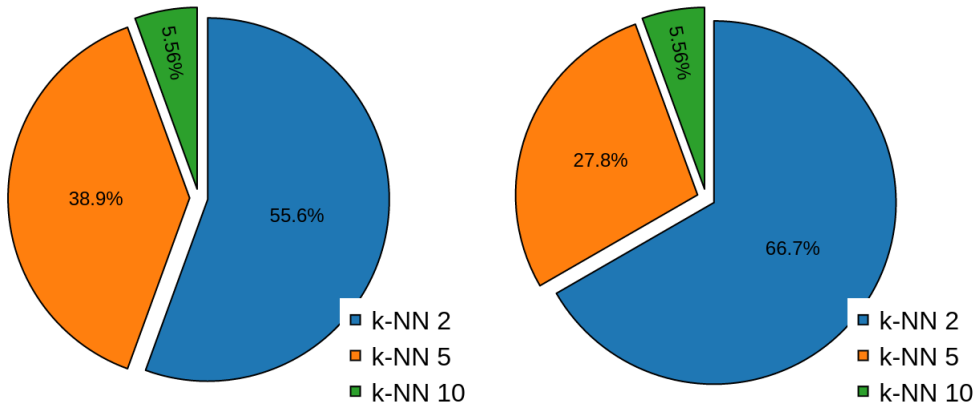
We can certainly conclude that the use of detector combinations with the combination of three descriptors improve the content representation of the images and the advantage of feature fusion is clear. Therefore, the combinations of SIFT, SURF and SC are used in the following experiments.

5.2.4 Effect of k-NN on effective retrieval

The effective retrieval results obtained by varying k ($k = 2, 5, 10$) during the nearest neighbor retrieval are presented in this section.

Dataset	Detector combination	mAP^e		
		$k = 2$	$k = 5$	$k = 10$
PCD_DB	hesaff-star	0.409	0.402	0.372
		0.421 (adaptive k 2,5 & 10)		
	hesaff-har	0.398	0.380	0.351
		0.401 (adaptive k 2,5 & 10)		

Table 5.6: Effective mAP^e using varying k-NN (k=2,5,10) and adapting it with the prediction model for PCD_DB.



(a) (b)

Figure 5.4: Distribution of k values across the queries for PCD_DB: (a) hesaff-star (b) hesaff-har

In general, the best mAP^e is achieved with $k = 2$, followed by $k = 5$ and $k = 10$. The retrieval accuracy difference is 0.7% between $k = 2$ and $k = 5$ and 3.8% between $k = 2$ and $k = 10$ for 'hesaff-star' in PCD_DB as presented in the Table 5.6. Higher value of k includes noisy neighbors during the nearest neighbor search of the query, leads to curtail the retrieval

accuracy. Our model is also capable to predict the best k from varying k values ($k = 2, 5, 10$) and accordingly adapt the best k value for each query. The mAP^e obtained by adapting k is represented by the highlighted table cells in the Table 5.6. We observe that the accuracy is increased by 1.2% for 'hesaff-star' compared to previous best results with $k = 2$. The statistical observation on the distribution of the adaptively selected k values is illustrated in the Fig. 5.4. Approximately 56% of the queries are executed with $k = 2$, followed by $k = 5$ and $k = 10$ for 'hesaff-star' combination.

5.2.5 Adaptive selection of the k-NN and the best detector combination

The mAP^e can be further refined by adaptive selection of detector combinations with varying k values by employing the prediction strategy of the regression model. In the Table 5.7, 6 dif-

Dataset	Detector combination	mAP^e		
		$k = 2$	$k = 5$	$k = 10$
PCD_DB	hesaff-star	0.409	0.402	0.372
	hesaff-har	0.398	0.393	0.354
	Adaptive detector combination	0.445 (Adaptive k 2,5 & 10)		

Table 5.7: Effective mAP obtained by selecting optimal detector combinations and optimal value k for each query image for PCD_DB.

ferent combinations of mAP^e (the highlighted table cells) obtained with two best performing combinations and 3 varying k values ($k = 2, 5, 10$) are consolidated for PCD_DB. The mAP^e is increased by 3.6% for PCD_DB compared to the previous best with $k = 2$. We also observe from the Fig. 5.5, that the majority of the queries, *i.e.,* approximately 50% are executed with $k = 2$ of both detector combinations for PCD_DB, followed by $k = 5$ and $k = 10$. Also, ∼56% of the queries are selected from hesaff-star combination, which is the best performing combination for PCD_DB. Although certain detector combinations and k values dominate positively during image retrieval, the adaptive inclusion of the other combinations and k values further improve the image retrieval performance. Again, these results demonstrate the relevance of the adaptation of the detectors to each image content for cross-domain retrieval.

5.2.6 Cross-domain image retrieval examples

Several examples of cross-domain image retrieval by querying in PCD_DB are exhibited in this section. The query images used here are old postcards, posters, paintings, etc. of various monuments in Paris.

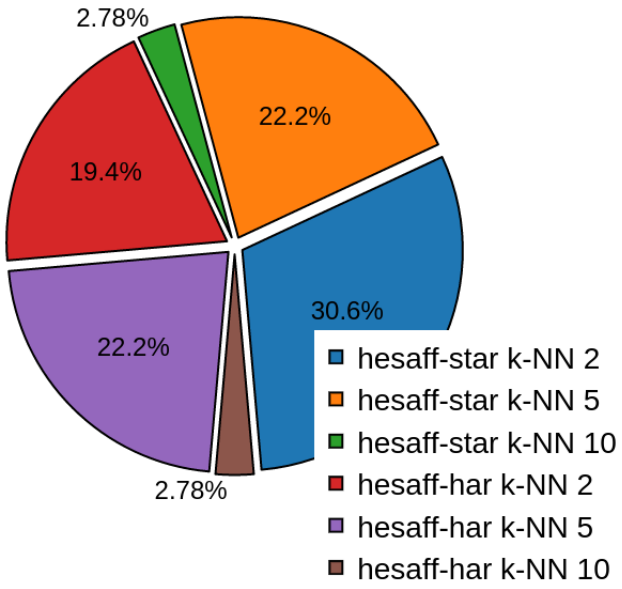


Figure 5.5: Distribution of predicted values of k and detectors combinations across the queries for PCD_DB.

In the example depicted in the Fig. 5.6, each query is executed with hesaff-star combination. Although the queries are different by nature from the search dataset, the retrieval results are quite accurate.

Similarly, in the example of Fig. 5.7, each query is executed with 'hesaff-har' combination and the first 10 retrieved images belong to the same monument category.

5.3 Exploration of a digitized photographic museum collection

We apply FII search framework for the Nicéphore Niépce museum photographic collection application. The experiments and evaluation details are presented in the Sec. 5.3.1 and it follows by the illustration of image retrieval application on the museum collection in the Sec. 5.3.2.

Before jumping to the evaluation, let us introduce the brief history of Nicéphore Niépce museum. It is in 1861 that were gathered pictures, personal items and the first cameras in the world used from 1816 onwards by Nicéphore Niépce, inventor of photography and a native of Chalon-sur-Saône. One of the oldest surviving photographs, 'Le Point de vue du Gras', created by Nicéphore Niépce 1826 or 1827 is depicted in Fig. 5.8. The recognition by researchers and historians of the great value of this collection led to the creation of the Niṕce museum which opened to the public in 1974. However, the Niépce museum, even if it has the name of the

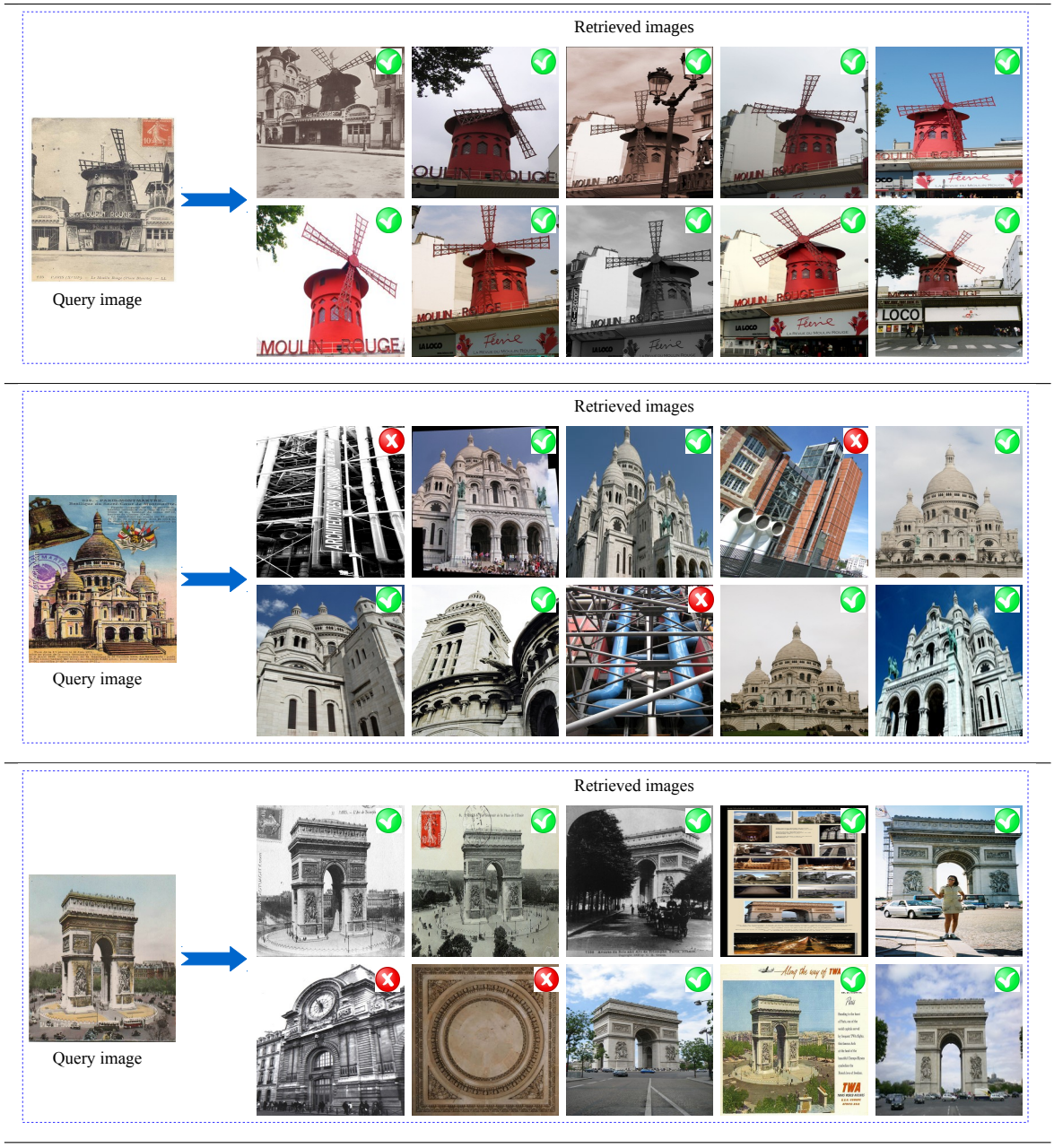


Figure 5.6: Cross-domain retrieval example by querying in PCD_DB using 'hesaff-star' combination and $k = 2$ configuration. The first 10 retrieved images are presented by decreasing order of similarity, from left to right and top to bottom.

inventor of photography, is not only dedicated to him.

From its creation and thanks to the influx of donations and an active acquisition policy, the Niépce museum has set the aim of telling the whole photography history in its technical and artistic aspects as in its popular and commercial uses. Dedicated to photography, it proposes to explain all the aspects of a practice, since its emergence in the 19^{th} century to its current developments. From Niépce heliographies to the first color photographs by Louis Ducos du Hauron (1868), from daguerreotype to tintype (photographies on metal made by fairground and itinerant photographers in the 19^{th} century), from film photography to digital photography, from

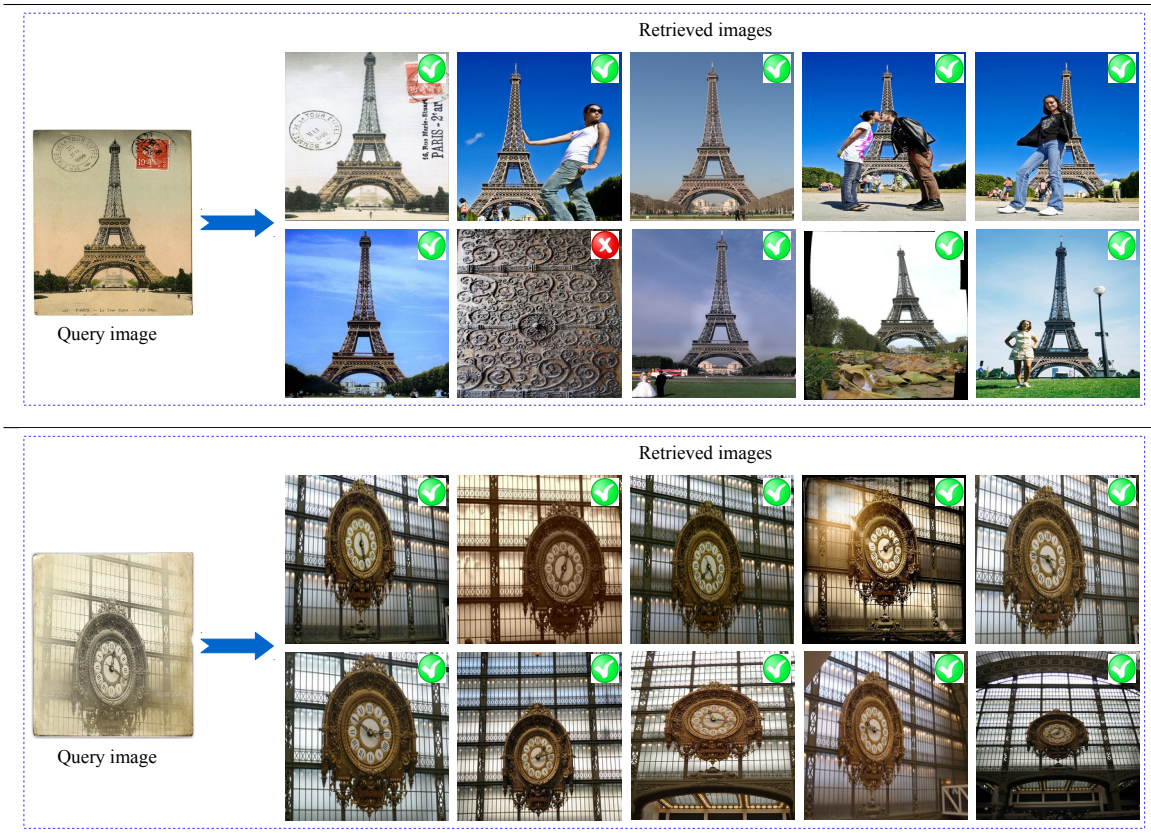


Figure 5.7: Cross-domain retrieval example by querying in PCD_DB using 'hesaff-har' combination and $k = 2$ configuration. First 10 retrieved images are presented by decreasing order of similarity, from left to right and top to bottom.

Figure 5.8: 'Le Point de vue du Gras': One of the oldest photographs created by Nicéphore Niépce.

Pictorialism to the French humanism of the 1950s, through the modernity of the New Vision in the 1930s, from street photography to studio photography, the museum covers all fields of photography. The digitization of its collections, which began in 1999 and the creation of its database in 2003 enabled the Niépce museum to develop several interactive multimedia devices for the general public presented permanently in its showrooms and to make available some of those contents via virtual exhibitions on the internet. Some examples of the digitized contents of the museum are shown in Fig. 5.2.

5.3.1 Evaluation of the proposal for the Niépce collection

The structure and the sequence of the experiments follow the similar pattern of the Sec. 5.2. The experiments are conducted on the two image datasets, *i.e.,* Paris_DB and Museum Nicéphore Niépce collection.

1. Paris_DB: This dataset is discussed in the Sec. 3.5.1 of Chapter 3. It is used to train the regression model.
2. Museum Nicéphore Niépce dataset (MNN_DB): This is a newly constructed benchmark where approximately 4000 images of 19 classes are sampled from Nicéphore Niépce collection. Few sample images from this dataset are depicted in the Fig. 5.9.

Figure 5.9: Sample images from the Museum Nicéphore Niépce dataset used in the experiments.

The experimental configuration and parameter settings are kept mostly similar to the Sec. 5.2.1. Additionally, we include BRISK [Leutenegger et al., 2011] detector in the experiments.

To begin with, the different combinations of detectors and descriptors are used to extract the features from the images. Then the prediction strategy based on spatial complementarity measures and a regression model is employed to stipulate the best feature combinations. The regression model is trained with Paris_DB. The results for the prediction of the best suitable detector combinations are presented in the Table 5.8: combination 'hesaff-orb' is associated with the highest predicted mAP, followed by 'hesaff-har'. Therefore, these combinations are used in the following experiments.

Detector pair	mAP^p	Detector pair	mAP^p	Detector pair	mAP^p
hesaff-colsym	0.461	hesaff-mser	0.431	hesaff-har	0.478
hesaff-star	0.468	hesaff-orb	0.482	hesaff-brisk	0.476
colsym-mser	0.419	colsym-har	0.439	colsym-star	0.435
colsym-orb	0.440	colsym-brisk	0.441	mser-har	0.420
mser-star	0.421	mser-orb	0.430	mser-brisk	0.429
har-star	0.452	har-orb	0.460	har-brisk	0.442
star-orb	0.450	star-brisk	0.433	orb-brisk	0.431

Table 5.8: Different detector combinations and mAP^p using the regression model, for the MNN_DB.

Based on the results presented in the Table 5.8, the effective retrieval results of the detector combinations are presented in the Table 5.9. The highlighted table cell represents the best achieved result. The best effective mAP (mAP^e) is achieved with 'hesaff-orb' combination, as predicted by the regression model. In accordance with the prediction result, the second best effective mAP^e is achieved with 'hesaff-har'. Similarly, the worst performing combination is also obtained with 'colsym-mser' (see Table 5.9).

Dataset	Detector pair	k-NN	mAP^e
MNN_DB	hesaff-orb	2	0.612
	hesaff-har	2	0.609
	hesaff-brisk	2	0.602
	star-orb	2	0.517
	colsym-mser	2	0.401

Table 5.9: Effective mAP^e of detector combinations, for the MNN_DB.

To compare, Table 5.10 provides results by using these detectors alone. Detector combinations produce superior results compare to the single detector for image retrieval. Again, these experiments on the museum image collections demonstrate the advantage of using detector combinations and the regression framework proposed.

Dataset	Detector	k-NN	mAP^e
MNN_DB	hesaff	2	0.593
	orb	2	0.467

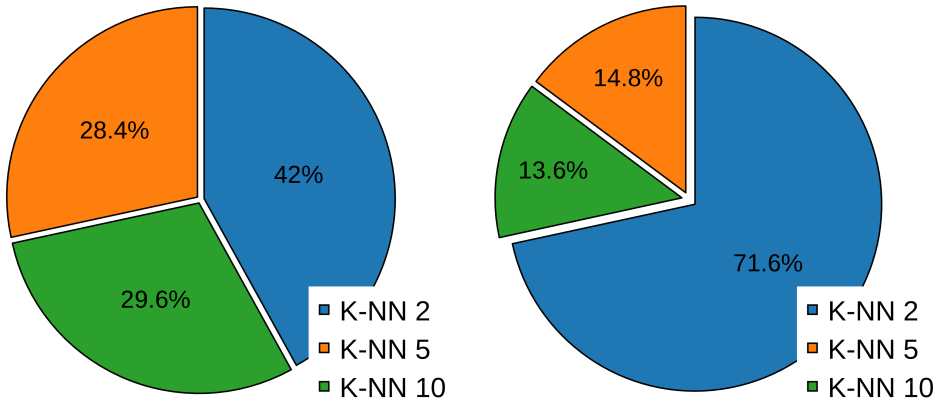
Table 5.10: Effective mAP^e of single detectors, for the MNN_DB.

Table 5.11 presents the retrieval results by varying k value ($k = 2, 5, 10$) and observing the change on mAP^e.

We observe that the accuracy is increased by exploiting the adaptive selection of k for each query image. The mAP^e (the highlighted table cell) is increased by 1.1% compared to $k = 2$

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
MNN_DB	hesaff-orb	0.612	0.602	0.591
		0.623 (adaptive k 2,5 & 10)		
	hesaff-har	0.0.609	0.584	0.550
		0.612 (adaptive k 2,5 & 10)		

Table 5.11: Effective mAP (mAP^e) for the MNN_DB, by varying k-NN (k=2,5,10) and adapting it with the prediction model.



(a) (b)

Figure 5.10: Distribution of k values across the queries for museum image collections: (a) hesaff-orb (b) hesaff-har.

for 'hesaff-orb'. Figure 5.10 depicts the distribution of the k for all the queries. For 'hesaff-orb' combination, 42% of the queries are selected with $k = 2$.

It is further increased by adaptive selection from the 2 best performing detector combinations and by varying k-NN values: as presented in the Table 5.12, there is an increment of an accuracy of 3.9% compared to the previous best result obtained 'hesaff-orb' and $k = 2$. To understand more precisely, the Fig. 5.11 shows how the selections are distributed across the queries. Most of the selections, approximately 50%, are with $k = 2$ from both the combinations. Also, 60.5% are selected with the best performing pair, 'hesaff-orb', and remaining 39.5% with 'hesaff-har'.

5.3.2 Image exploration applications on the museum collection

In collaboration with Nicéphore Niépce museum, the work presented in this chapter is evaluated for several scenarios at different levels of the museum needs: from the archiving up to the

Dataset	Detector pair	mAP^e		
		$k=2$	$k=5$	$k=10$
MNN_DB	hesaff-orb	0.612	0.602	0.591
	hesaff-har	0.609	0.584	0.550
	Adaptive detector combination	0.651 (Adaptive k 2,5 & 10)		

Table 5.12: Effective mAP^e obtained by selecting optimal detector pairs and optimal value k for each query image, for the MNN_DB.

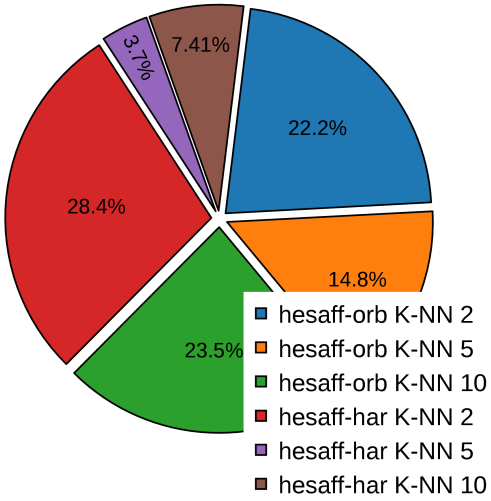


Figure 5.11: Distribution of k and detectors pairs across the queries, for the Niépce collection.

exhibition, with the objectives of providing new tools for the experts and to better highlight their photographic collections for the general public. In the following subsection 5.3.2.1, we illustrate these objectives, *i.e.,* linking the images within museum collections by cross-domain retrieval. Then Sec. 5.3.2.2 present an overview of the POEME image exploration immersive environment and the adaptation of the proposed FII search engine within this entire system.

5.3.2.1 Intra-linking of contents in the Museum Nicéphore Niépce collection

The Museum Nicéphore Niépce collections are hugely diverse and these collections belong to different domains. Thus the cross-domain retrieval using FII image search engine is beneficial for intra linking the image contents within the museum collections. In this section, two applications are explored:

1. Online application for image exploration and visualization: the exploration of the museum collection by retrieving visual similar images to the user given query.

2. Offline application for database organization: a tool for database indexing by linking the visually similar image acquired from different sources. This could be useful for the

curator, archivist to organize the collections by annotating the images and propagating annotation.

We begin with the online image exploration examples from the MNN_DB as queries. The image retrieval results, searching for a particular content in the Niépce collection, are depicted in Fig. 5.12. Here, the dominant configuration exhibited by the selection model is hesaff-har and hesaff-orb as the detector combination and k-nearest neighbor value is 2.

Figure 5.12: Image retrieval example by querying the Niépce collection: for a query, the 10 first retrieved images are presented by decreasing order of similarity, from left to right and top to bottom by FII search engine.

The example in Fig. 5.12(a) concerns a large collection of scans with similar layout. Here, the retrieved images are related to bridge and port constructions, they exhibit the different construction stages of the same barrage and the other similar bridges or ports, and then allow to focus on and isolate a thematic subset of the collection. The example is shown in Fig. 5.12(b) returned photographs that belong to several collections, in relation to war and conflict. Being able to link these contents automatically helps the archivist in indexing the collections as they are digitized, as well as it may help the curator and commissioner in the selection of interesting contents for an exhibition, as a complement to traditional approaches of selection usually based on the experience, memory and archivist indexing. It can also provide to the visitor a new way of browsing the collection, by querying it with particular photographs or parts of photographs (the local descriptors employed enable partial queries).

The Niépce museum also developed an immersive and interactive environment *in situ*, which integrates the proposed FII image search engine, with the ambition of proposing virtual exhibitions centered on the visitor who will have the possibility to organize himself the navigation by querying the collections through several modalities.

Another scenario is related to the variety of the sources of the Nicéphore Niépce museum, which keeps complete archives of photographers such as negatives, contact sheets, and prints. Thus, different formats of the same image could exist on several physical media. The classification of these archives is often lost before arriving at the museum. The first task of a museum, when it received a photographer's collection, is to reorganize the collection and match each part of archives to each other. For example, Jean Moral[4] was mainly a fashion photographer in the 1930s. His collection includes thousands of photo prints, images of photo negatives, and contact sheets of small print of (6 x 6) cm. Jean Moral published hundreds of images in fashion magazine Harper's Bazaar[5] (HB). Two-thirds of the Jean Moral's collection is related to fashion and the images were never dated and captioned. To put the date, to caption the models, archivists are compelled to compare each image with each page of HB magazine where an image of Moral was published. Therefore for an archivist, the magazine's exploration takes several of hours for matching between the publication of the magazines and the photographer's collections. However, this tedious and rigorous manual work can be simplified and automated by introducing our image retrieval framework.

Two examples of the matching between photographer's collections which were exploited in the HB magazines are depicted in Figs. 5.13 and 5.14. The several hundreds of scan pages of HB magazines are used as a dataset and Jean Moral's photograph's collections, *i.e.,* photo print, contact sheets are queried in this dataset.

Figure 5.13: Image retrieval example by querying in the Niépce Harper's Bazaar collection: Photo contact sheet of Jean Moral's collection is used as query. The 6 best results are presented by decreasing order of similarity, from left to right.

In the Fig. 5.13, the same photo from the contact sheet was identified in two different issues of HB, *i.e.,* March-1935 and May-1935 issues, retrieved at the two first positions respectively. In the example shown in the Fig. 5.14, the query image is the developed photo print of the same contact sheet (as in Fig. 5.13) from Jean Moral's collection. Interestingly, the first two retrieved images were exactly same as in Fig. 5.13. Therefore, for an archivist or museum curator, the

[4] https://fr.wikipedia.org/wiki/Jean_Moral
[5] http://www.harpersbazaar.com/

Figure 5.14: Image retrieval example by querying in the Niépce Harper's Bazaar collection: Single developed photo print from the contact sheet of Jean Moral's collection is used as query. The 6 best results are presented by decreasing order of similarity, from left to right.

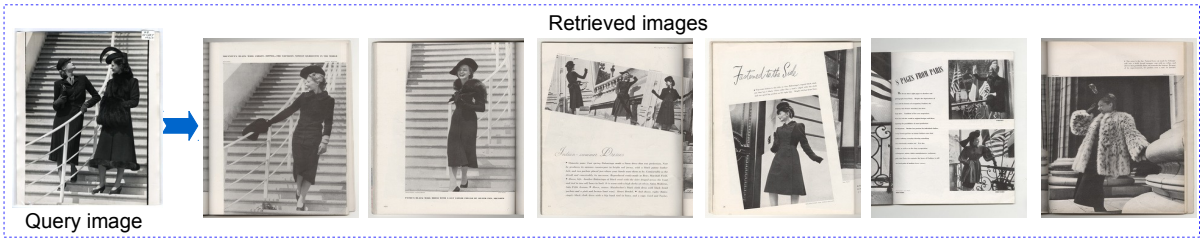


Figure 5.15: Image retrieval example by querying in the Niépce Harper's Bazaar collection: Photo print of Jean Moral's collection is used as query. The 6 best results are presented by decreasing order of similarity, from left to right.

task of image organization by content, context, the format has become comparatively easier employing our image retrieval framework. The entire process of image retrieval and linking up between the similar image contents present in the different formats and different collections have become swift and unadorned.

Jean Moral's photo prints were queried in the HB collection, as depicted with the example of Fig. 5.15. Interestingly here, we discovered that the first two retrieved images do not correspond to the same item but to very similar shooting contexts (they were published in the September-1937 HB issue), thus providing to the archivist an additional insight on the photograph's collection.

Few more retrieval examples from Niépce collections are depicted in the Figs. 5.16 and 5.17, where photo prints and contact sheets are used as query images.

Figure 5.16: Image retrieval example by querying in the Niépce Harper's Bazaar collection: Jean Moral's collection is used as query. The query is executed with 'hesaff-orb' and $k = 2$ configuration. The 6 best results are presented by decreasing order of similarity, from left to right.

Figure 5.17: Image retrieval example by querying in the Niépce Harper's Bazaar collection: Jean Moral's collection is used as query. The query is executed with hesaff and $k = 2$ configuration. The 6 best results are presented by decreasing order of similarity, from left to right.

5.3.2.2 Overview of the interactive image exploration system

One ambition of the project POEME is to design, prototype and implement the immersive environment inside the Nicṕhore Niṕce Museum in Chalon-sur-Saône, with the double aim of improving management of the collection by their experts as well as contributing to its enhancement close to the general public. The target outcomes are technologies for creating personalized and engaging digital cultural or play experiences on digitalized photographs collections, in particular on the collections of the Museum.

An engaging issue for CBIR is visualization and interaction. The interaction between man

and machine is an active field of research, as the increasing capabilities offered by computers and new information technologies open new possibilities. An important part of the research is focused on the development of intuitive interaction metaphors which would help users operate in a natural way with a system. The development of worldwide networks also contributed to the emergence of new research projects in the field of collaborative interfaces.

The simplest man/machine interface may be the standard combo screen and keyboard/mouse. Contrary to this, more immersive interfaces exist, like CAVEs (CAVE Automatic Virtual Environment [Cruz-Neira et al., 1993]). Between these two extreme cases, it is, of course, possible to design semi-immersive environments. Depending on the degree of immersion we need, different input devices can be chosen. These devices can either be haptic or non-haptic. Haptic devices, such as haptic gloves, represent a very interesting choice in virtual reality applications [Burdea, 1996, 1999]. Amongst non-haptic devices, we can find graphic tablets with pens. Another interesting input device is touch-screens. One interesting advantage of touch-screens is their multi-touch capability. This gives the possibility to implement intuitive interfaces by using more than one finger. Another advantage is, of course, the possibility to display information on it.

When manipulating image collections, all the aforementioned interfaces do not integrate jointly innovative tools based on CBIR and immersive devices. Therefore, we design and create an immersive environment. POEME is a digital installation that invents new ways to browse, an immersive environment in which to explore photography collections of the museum and the content-based search engine, such as FII search engine. This state-of-the-art man/machine interface provides the users efficient ways to analyse the large quantity of data. In order to help the user, the tool integrates innovative visualization methods and interaction metaphors. The immersive aspect is established using new input and output devices like touch screens. The overview of the POEME set up is shown in the Fig. 5.18.

Figure 5.18: Entire set up of the POEME image exploration system.

The entire setup consists of several devices such as multiple projectors, touch screen, computer, Kinect sensor as shown in the Figs. 5.19, 5.20. The POEME system is capable of handling a large amount of multivariate and multi-dimensional data. The visualization methods and interfaces integrated into visualization software. This software is able to handle massive data and to show them in an intuitive manner.

Figure 5.19: Different components of POEME system.

Figure 5.20: Image navigation, zoom in, zoom out, etc., using Kinect senor in POEME system.

In addition, the developed prototype serves as an innovation box by bringing together, in one tool, a set of technologies: a combination of image descriptors for query-by-example search, relevant feedback system, visual metaphors and data representation, interaction systems, immersive space for image collections analysis. These different features are depicted in the Figs. 5.21, 5.22, 5.23.

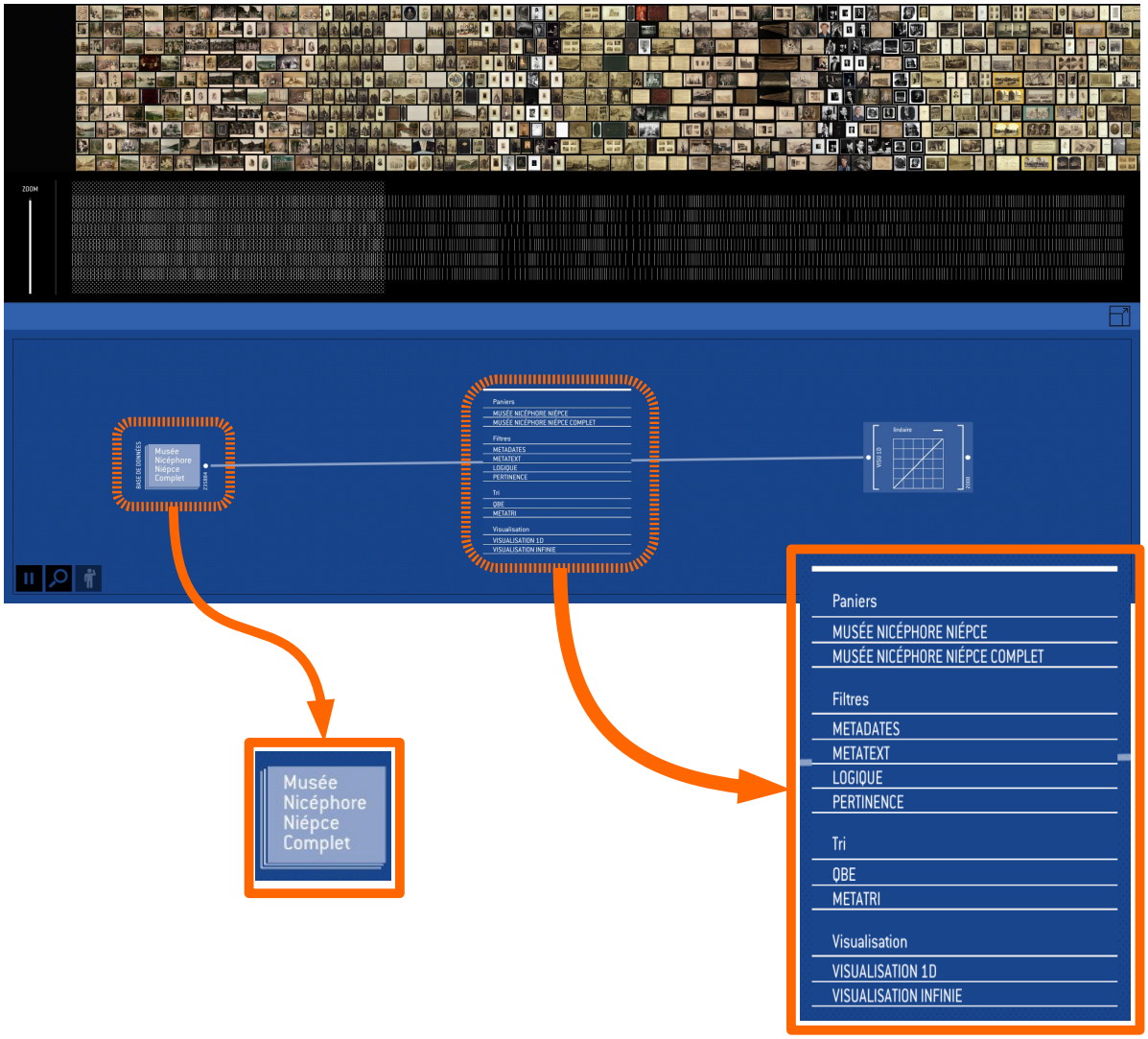


Figure 5.21: The interactive home screen of POEME system. Images can be explored by different search criteria: query-by-example, text-based search, relevance feedback, author, year, associate keywords, etc.

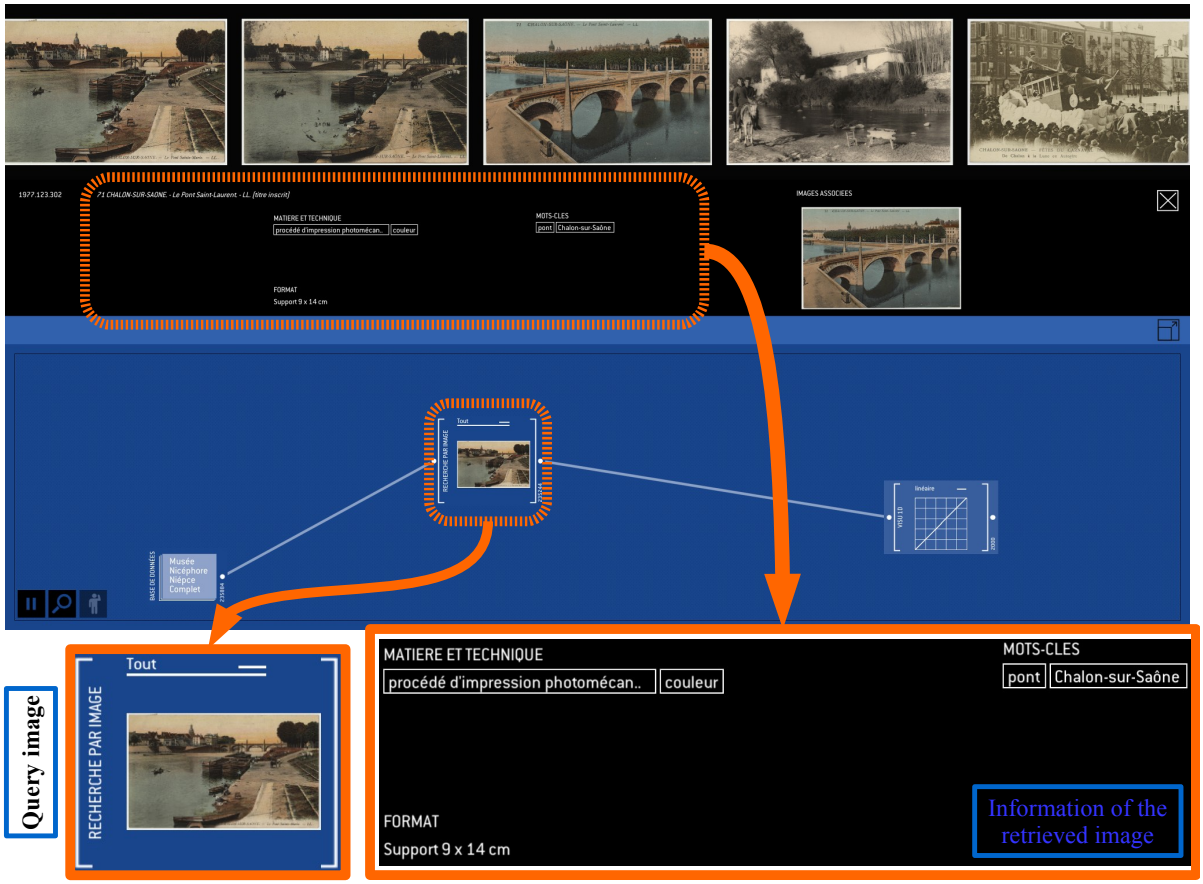


Figure 5.22: Query-by-example image search in POEME system.

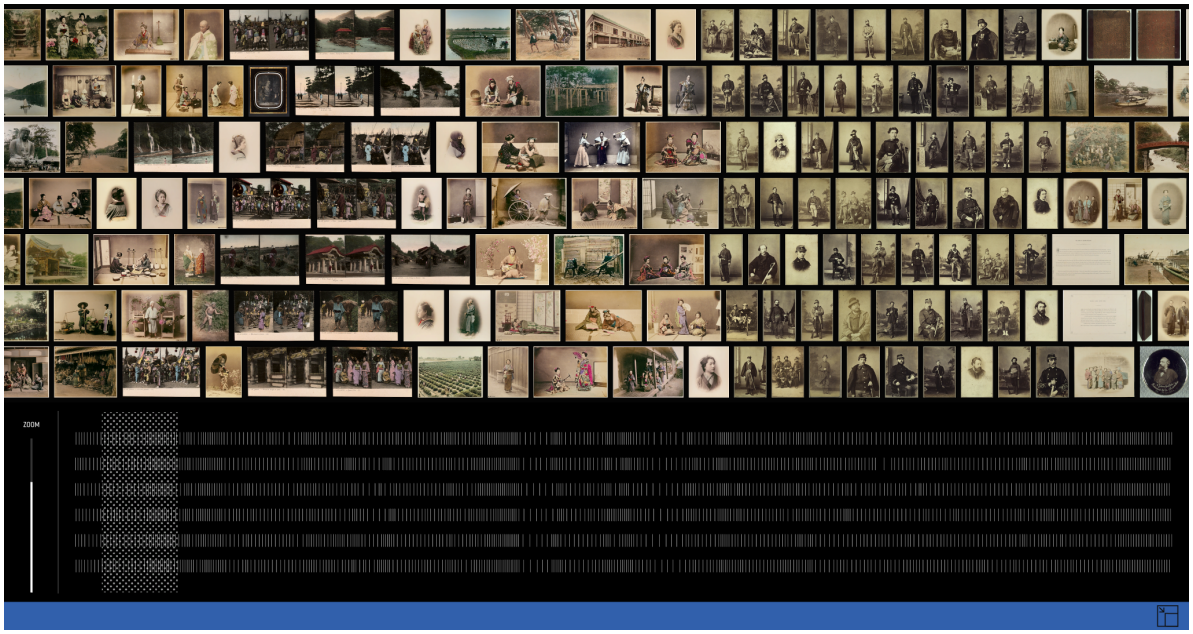


Figure 5.23: Image exploration in POEME system.

5.4 Inter-linking of the contents with application to image localization

Cross-domain retrieval is useful for inter-linking the media content task, *i.e.,* linking the visually similar images between two characteristically diverse image databases. In this section, we present two more applications which inter link image contents using cross-domain retrieval:

1. Inter-linking of the contents between museum collections and public databases and
2. Image-based localization by cross-domain retrieval using geo-referenced databases.

Sections 5.4.1 and 5.4.2 are dedicated for these two applications.

5.4.1 Inter-linking of the contents between museum collections and public databases

In this application, selected images of Niépce collections, which are mostly old images of landmarks, monuments, are linked with the public databases. Here, the illustration of this problem with photographs of the Niépce collection and camera captured or street-view imagery (*e.g.,* Flickr, Google street-view) is explained. Figure 5.24 shows two examples of retrieval within the public Paris_DB dataset, with digitized old pictures of monuments from the Niépce collection as queries.

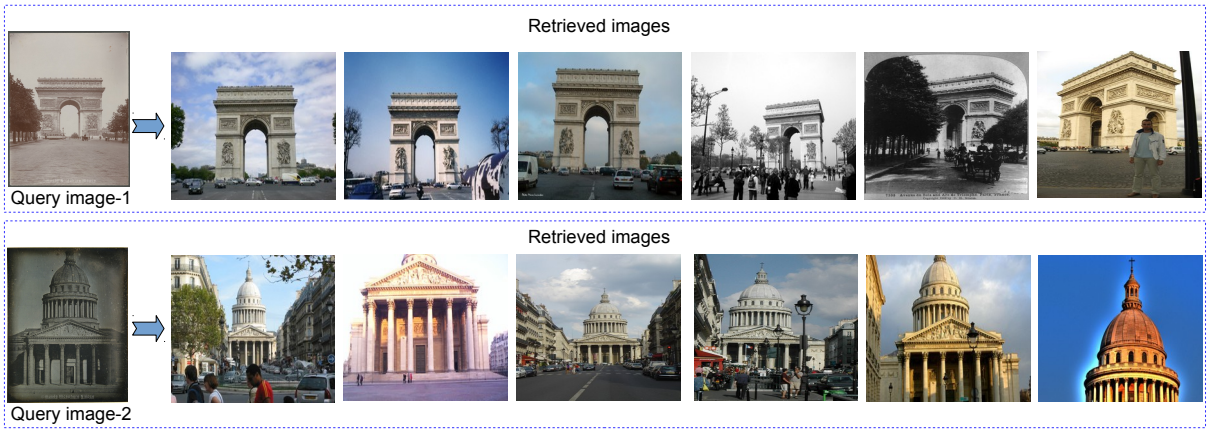


Figure 5.24: Illustration of cross-domain image retrieval: the query images are taken from the Niépce collection while the dataset is Paris_DB (Flickr). For each query, the 6 best results are presented by decreasing order of similarity, from left to right.

Although the query images were taken many years earlier with different technologies and different surroundings, we are able to retrieve images that represent the same monument or geographical area. Such cross-domain application is of great interest. By considering a georeferenced dataset (acquired with a mobile mapping system, for instance, the one of the French

Mapping Agency [Paparoditis et al., 2012]) such historical or cultural contents can be precisely re-localized. Moreover, cross-domain linking opens the door to their promotion outside the museum by connecting them with official mapping databases and services such as the French Géoportail[6] or a 3D web mapping engine (*e.g.,* itowns [Nguyen et al., 2015]), along the same lines as the linked Data initiative, which intends to connect distributed data across the web.

5.4.2 Image-based localization by cross-domain retrieval

Nowadays, the rapid growth of image acquisition in different domains makes cross-domain localization an emerging topic where a major challenge is to compare images of various domains, such as postcards vs street views, sketch vs geo-referenced images, etc. Section 5.4.2.1 presents the topic and it follows by the localization examples in the Sec. 5.4.2.2.

5.4.2.1 Related work on image-based localization

Image-based localization is the ability to provide an information of the position of an image sensor. In general, it relies on two families of approaches:

1. Direct and precise 6D pose estimation: In general, a small geographical area is covered, due to the complexity of the approaches involved.

2. Indirect approach: Retrieval of visually similar images from a georeferenced database which covers a large geographical area, and then localization of the query from the best matched retrieved images using the first approach.

These two families of approaches can be combined to perform precise 6D localization from a large geographical area.

In the direct pose estimation method, the 6-DOF (3-DOF spatial position and 3-DOF rotation) pose of the query is estimated on the fly using different techniques, such as Structure from Motion (SfM) [Wu, 2013] or Simultaneous Localization and Mapping (SLAM) [Milford and Wyeth, 2012]. Query pose can be estimated in the direct method by three ways. The first strategy relies on prior information of the query. The information can be obtained by different sensors, such as GPS [Poglitsch et al., 2015; Arth et al., 2015], magnetic compass [Svärm et al., 2014, 2017], etc. In the work of [Poglitsch et al., 2015], a particle filter is introduced to perform localization with the position information from a GPS sensor. The second strategy [Sattler et al., 2011; Li et al., 2012b; Frahm et al., 2010; Irschara et al., 2009] is feature point matching or 3D point cloud matching. In 3D point cloud, the 2D to 3D registration process is used to find matches between 2D points and 3D structure points. The localization estimation can be

[6] https://en.wikipedia.org/wiki/Geoportail

obtained from these matched points. The work of [Sattler et al., 2011] proposes to accelerate 2D-to-3D matching by associating 3D points with visual words and prioritizing certain words. In this context, the work of [Li et al., 2012b] considers worldwide image pose estimation. They propose a co-occurrence prior based RANdom SAmple Consensus (RANSAC) model [Fischler and Bolles, 1981] and bidirectional matching to maintain efficiency and accuracy. Irschara et al. [Irschara et al., 2009] consider sparse location recognition using 3D point clouds associated with SIFT features. They not only use real views but also generate synthetic views to extend localization capability. In other works, such as in [Lim et al., 2012], real-time 6-DOF estimation in large scenes for auto-navigation is addressed. The pose regression strategy is the third category. It learns and returns the pose from the input data by regression forest [Shotton et al., 2013; Valentin et al., 2015], CNN [Kendall et al., 2015; Liu et al., 2016b], etc. CNN in localization task was first introduced in the work of [Kendall et al., 2015] for small-scale relocalization.

In indirect pose estimation methods [Schindler et al., 2007; Shrivastava et al., 2011; Sattler et al., 2012; Song et al., 2016], the visually similar images can be retrieved by using different CBIR approaches, such as BoF models, machine learning, etc. In the work of [Song et al., 2016], for a given query, CBIR is used to retrieve the similar images from a large geo-tagged dataset. In other sense, CBIR reduces the outlier in the candidate image lists. The pose is estimated from these retrieved candidate images. In this work, only single feature, *i.e.,* SURF [Bay et al., 2008], is used for image retrieval. In other work, such as in [Schindler et al., 2007], a city-scale location recognition scheme is proposed. This work uses a vocabulary tree [Nister and Stewenius, 2006] to index SIFT features [Lowe, 2004] with improved strategies for tree construction and traversal. Zamir and Shah [Zamir and Shah, 2010] use Google street-view images for location recognition. They distinguish single image localization and image group localization. Corresponding voting and post-processing schemes are derived to refine the matching. The localization of mobile phone images using street-view databases is studied in the work of [Chen et al., 2011]. They propose to enhance the matching by aggregating the query results from two datasets with different viewing angles. They also find that histogram equalization and upright feature points are useful in the application. Zhang et al. [Zhang et al., 2011a] address performance degradation in large urban environments by dividing the search area into multiple overlapping cells.

Both direct and indirect methods have their pros and cons. Direct method can estimate more accurate pose of the query compared to the indirect method. However, in the direct method, the pose can estimate only from a small geographical area. On the contrary, the indirect method is useful when the query pose needs to be estimated from a very large geographical area.

Our work belongs to the indirect approach category. The goal is to localize query images, which are different in characteristics from database images, such as postcards vs. street view images, using cross-domain retrieval. The two steps of indirect approach are depicted in the Fig. 5.25. This application focuses only on the first step. We use the FII search engine with query-adaptive

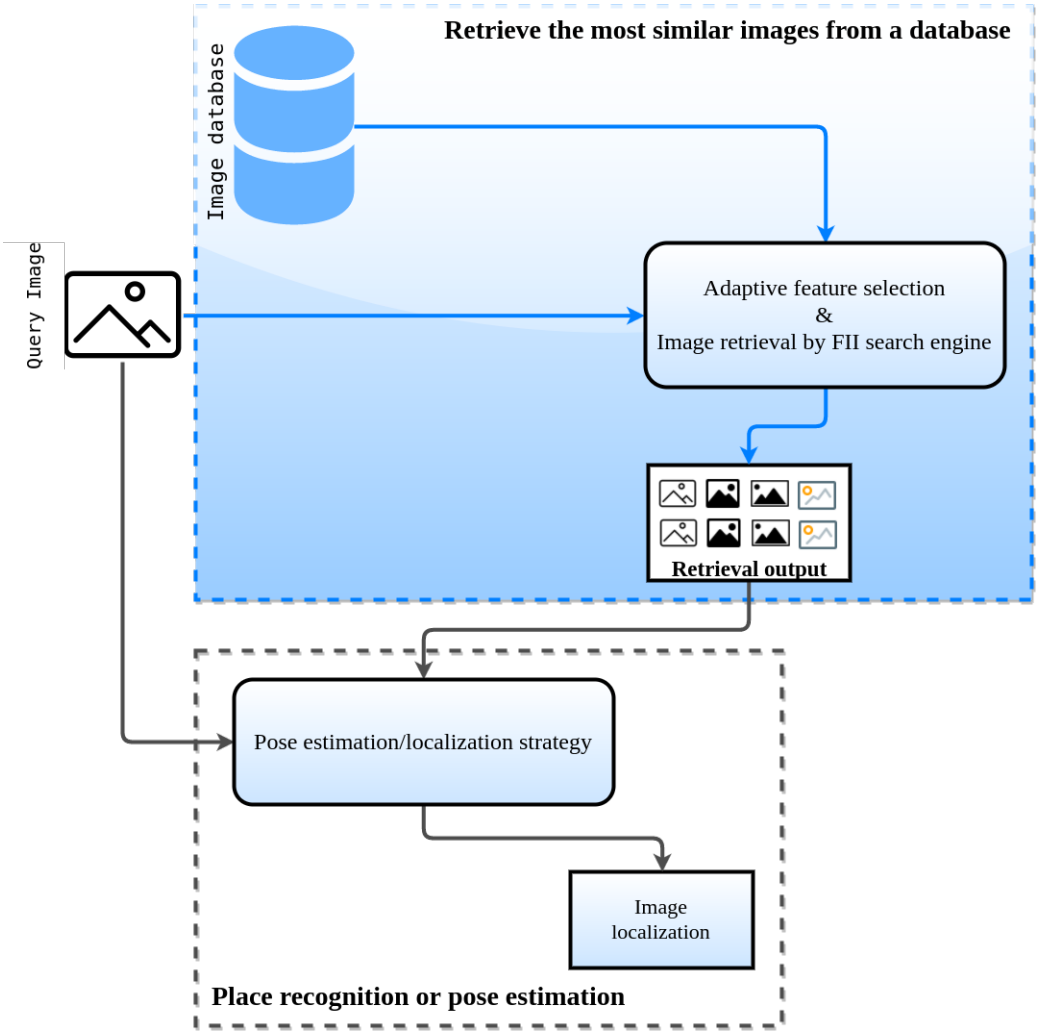


Figure 5.25: Overview of a classical image-based localization framework.

feature selection framework to retrieve visually similar images from cross-domain databases.

5.4.2.2 Image-based localization examples with FII approach

In this section, several experiments are conducted to address the cross-domain image localization challenge by exploiting the multiple feature fusion in FII image retrieval framework.

To begin with, a set of street-level geo-referenced images acquired by a mobile mapping system called Stereopolis [Paparoditis et al., 2012] is used as the image dataset for cross-domain image retrieval and localization. We used approximately 7000 images from 4^{th} and 5^{th} districts of Paris covering a distance of 51 km. The sample images are shown in the Fig. 5.26. About 6% of these images contain partial or full views of some historical monuments such as Notre Dame, Sorbonne, Pantheon, etc. Let us highlight that the known pose parameters of images are not required for training or retrieval. These parameters are only used in order to show the pose of retrieved images on a map and to check if the monument of interest is effectively in their field of view. Non-georeferenced images, such as old postcards of Paris monuments, are used

as queries for localization.

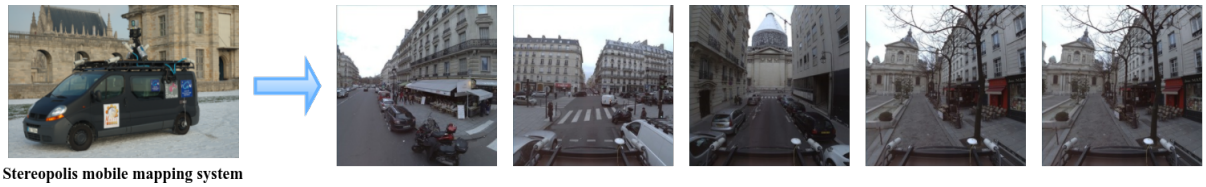


Figure 5.26: Sample images acquired by the Stereopolis used in the experiments.

The same detectors, descriptors, and other parameter configuration (such as k-NN values) as those used in Sec. 5.2.1 are used for these experiments.

An example of image localization is depicted in the Fig. 5.27. The query image (Panthéon) is executed with 'hesaff-mser' combination according to the prediction model trained on Paris_DB. Although the query is different in style, our retrieval technique is able to find relevant images that contain the same monument or similar geographic areas. After the retrieval, we use the pose of the retrieved images to mark on a geographical map in Fig. 5.28. The indicated numbers in the Fig. 5.28 are the retrieved images rankings according to the similarity with the query. For localization purposes, it is crucial to retrieve the most similar images in the first responses. The seven out of ten marks point to the correct monument.

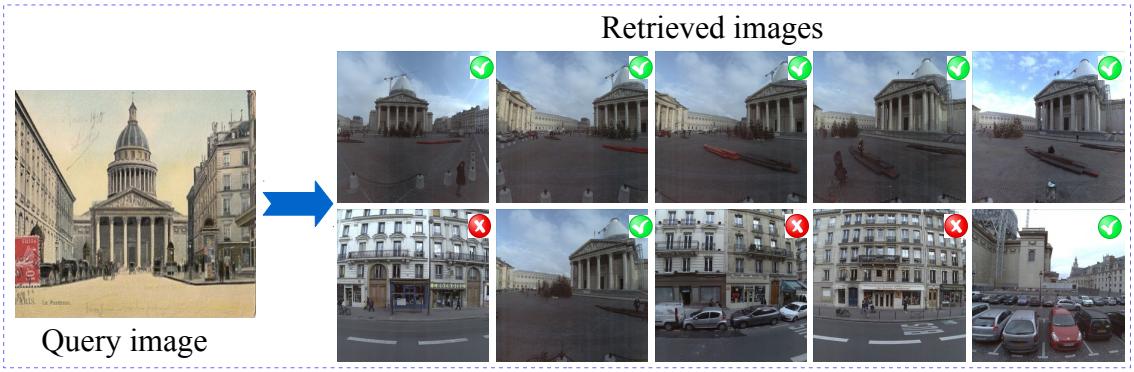


Figure 5.27: Illustration of cross-domain image localization of Panthéon postcard as a query using hesaff-mser feature combination: the 10 best results are presented by decreasing order of similarity, from left to right and top to bottom.

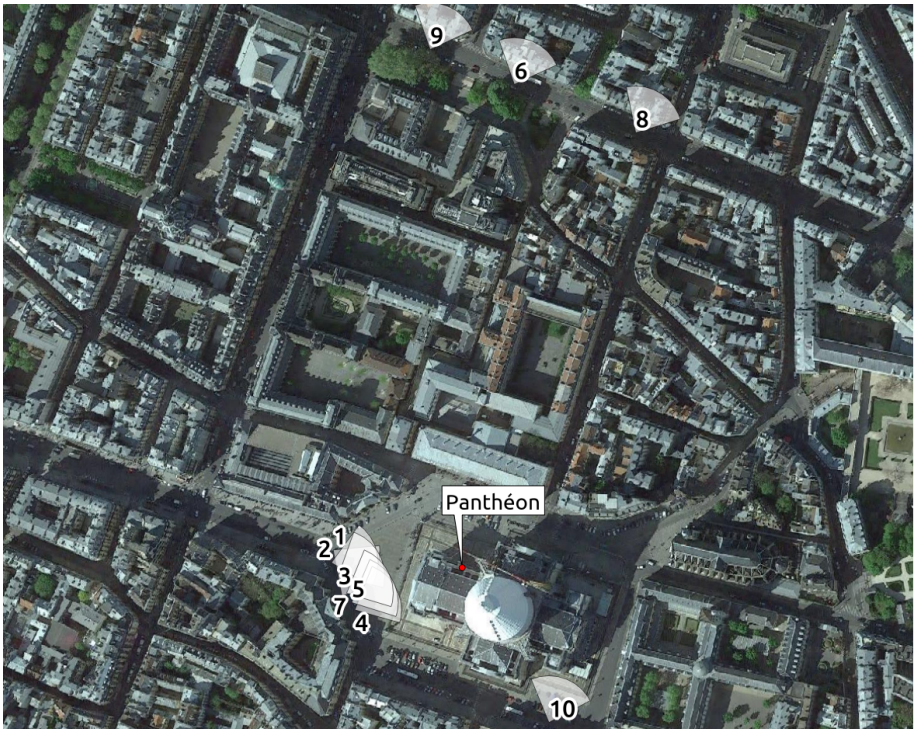


Figure 5.28: Image localization of the Panthéon postcard on the geographical map.

In the Fig. 5.29, the retrieval results of the same Panthéon query is presented using single feature (*e.g.,* hesaff). Only four out of first ten images are correctly retrieved. Thus the fusion approach is effective for image cross-domain image retrieval localization application.

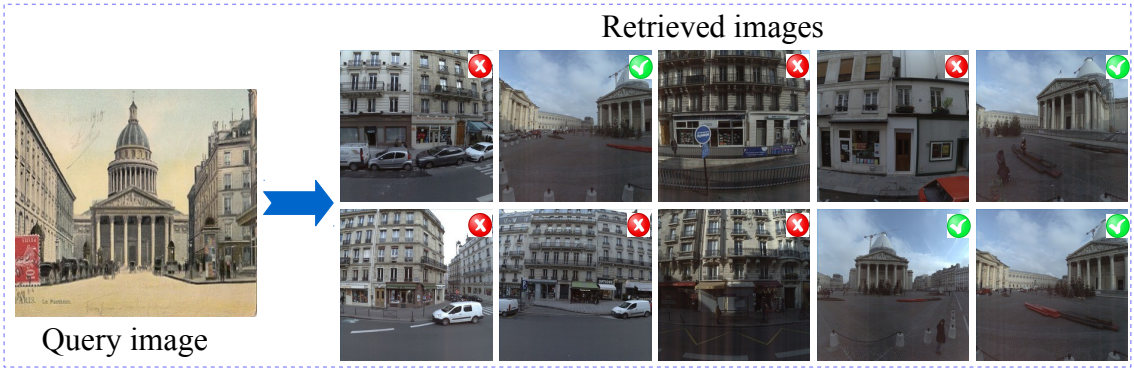


Figure 5.29: Illustration of cross-domain image localization of Panthéon query using the single feature - hesaff: the 10 best results are presented by decreasing order of similarity, from left to right and top to bottom.

Another example is shown in Figs. 5.30, 5.31 and with a photo query of Notre Dame - all ten retrieved images include the correct monument, which again validates the advantage of using the proposed method.

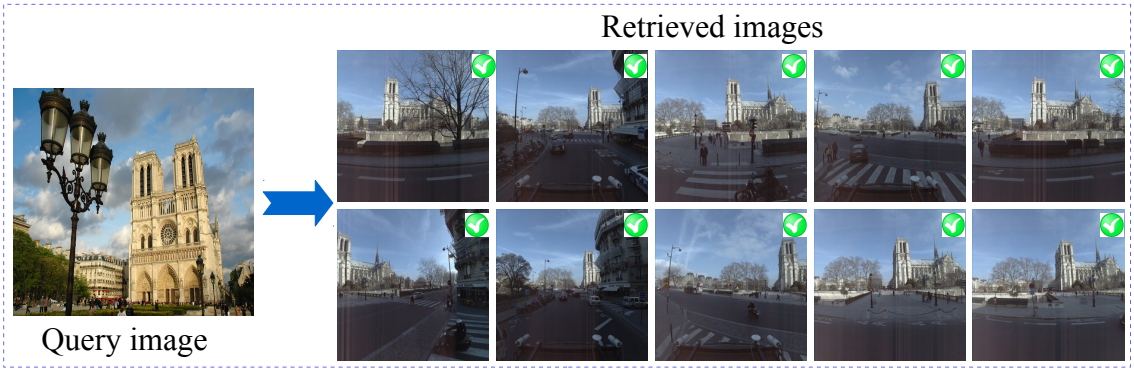


Figure 5.30: Illustration of cross-domain image localization of Notre Dame query: the 10 best results are presented by decreasing order of similarity, from left to right and top to bottom.

Figure 5.31: Image localization of the Notre Dame query on the geographical map.

Our proposed image retrieval framework could be very useful to remove outliers in the candidate image lists for different location-based applications, *e.g.,* 6-DOF image localization [Song et al., 2016], where image retrieval step is involved. In other works of [Frahm et al., 2010; Sattler et al., 2011], 3D point cloud model is employed for 6-DOF localization. The reason behind the using 3D point cloud is to remove the outliers by using RANSAC model. In this scenario, our image retrieval framework could also be useful for removing or simplifying the traditional steps of RANSAC model.

5.5 Conclusions

In this chapter, the proposed image retrieval search engine, the fusion of inverted indices search engine with adaptive feature selection model, is used to address the cross-domain image retrieval. This work focuses on the retrieval of similar images, which is a challenging task for images across different domains. The proposed feature combination based image retrieval technique is also used to address the cross-domain localization problem. It is able to perform combinations adaptively for each image, by training a regression model on the spatial complementarity of the features. Compared with the state of the art, this adaptive model improves the precision of retrieval across different visual domains where the relevance of a description may vary from one to another.

Additionally, two main applications of cross-domain retrieval, *i.e.,* intra-linking of museum image collections of the French museum Nicéphore Niépce and the cross-domain image localization by inter-linking the image content, are explored using the proposed framework. The proposed image retrieval framework has demonstrated its potential for the exploration and promotion of the museum contents at different levels. Several tasks can be accomplished, such as archiving of the images up to their exhibition in the museum and outside, and linking the museum image content with other categories of contents, such as geographical mapping contents. The proposed framework is used to address the cross-domain image localization issue, *i.e.,* the ability to estimate the pose of a landmark from visual content acquired under various conditions, such as old photographs, paintings, photos were taken at a particular season, etc. This improvement has a strong impact on image-based localization frameworks involving an image retrieval step, such as in [Song et al., 2016] where the 6-DOF estimation relies on the retrieval of images from a geolocalized dataset. The capability of localizing cross-domain content from artwork, multimedia, etc. opens the opportunity to link content with map databases and provide new tools for the promotion of visual and geographical content in various domains including culture, tourism, history, social sciences, etc.

Then we gave an overview of the POEME image exploration immersive environment and the adaptation of the proposed FII search engine within the entire system. To our knowledge, state-of-the-art projects do not integrate jointly innovative tools based on CBIR and immersive devices. With POEME system, the solution is provided to use of CBIR and visualization and interaction the most appropriate mutually, for efficient and effective exploration in image collections within an immersive environment. The successful implementation of POEME system obtains a synergy making more natural, richer and better personalized the navigation and interaction of users in cultural photographic collections. Thus, the entire proposition achieved its main objective: to bring together the needs of different users, such as archivist, editor, iconographer, curator, etc., for collective and collaborative work around important digitized photographic collections.

Chapter 6

Conclusions and perspectives

6.1 Main contributions

To conclude this work, we summarize our contributions before presenting perspectives in this chapter. There were two main objectives of this work. The first objective and the core of the thesis rested on the proposal of a query-by-example image retrieval strategy for combining low level image descriptors. The second objective concerned the combining of low-level descriptors by evaluating their complementarity according to spatial criteria.

Query-by-example image retrieval by multi-descriptor fusion: In this objective, the focus was on the proposal of a model for combining low level and generic descriptors in order to obtain a descriptor of a higher semantic level. Considering the diverse nature of image contents, the fusion strategy should maintain the genericity in order to be able to index different types of visual contents. Not only that, the complexity of the strategy should meet the reduced retrieval times, even with the large volume of image dataset.

Thus, a novel content-based image retrieval tool, named Fusion of Inverted indices (FII) image search engine, was proposed. The proposed fusion strategy was developed on the concept of inverted multi-indices structure. It is generic enough and robust to combine any number of multi-dimensional image descriptors by integrating their responses to a query in finer subdivisions. The experiments performed for similarity search on several image datasets of diverse sizes and contents have demonstrated the relevance of their combination through this structure: the combination of different image features clearly improves the content representation, and the strategy of fusion brings distinctiveness during the nearest neighbor search. The FII search strategy has demonstrated its superiority facing two state-of-the-art fusion approaches. Additionally, we have shown that the use of dimension reduction techniques as description decomposition, PCA and PLS, contributes to improving distinctiveness during similarity search, while potentially reducing the volume of manipulated features, and then limiting the computational

complexity despite the multiple descriptions involved.

Fusion of descriptors based on their effective spatial complementarity: The substantial number of available local feature descriptors in the present literature of Computer Vision and CBIR, with respective advantages and drawbacks, makes it arduous to determine the most relevant descriptors for a given task and a given dataset. Therefore the second objective of the thesis was to evaluate complementarity of the local features and combining them for image retrieval task.

Hence, a solution to predict the optimal combination of local features, for improving image retrieval performances, based on the spatial complementarity of interest point detectors, was proposed. The main contribution of this proposal is the possibility to select adaptively the best detector combination for not only globally for an image dataset, also for each query, depending on the image content, in a query-by-example image retrieval, such as in FII image search engine. The solution proposed rests on the use of spatial complementarity criteria between local features and on a linear regression model that models the relationship between complementarity and optimal performances during image retrieval. Additionally, this proposal allows to optimally fit some other parameters of the FII image search engine, such as the best k during the k-nearest neighbor search. Although the statistical analysis highlights the dominance of some detectors pairs, the adaptive selection of the features allows refining the results favorably. The conducted experiments have clearly highlighted the impact of the spatial complementarity of the selected features on the image retrieval performance: the higher complementarity scores imply a more distinctive representation of the content. We have demonstrated that our image retrieval framework achieved better retrieval accuracy compared to state of the art strategy [Li et al., 2015] on dataset Holiday_DB. The proposed framework can effectively reduce the overall experimental time by narrowing down the choice of detectors.

In addition, several applications were explored as well. The proposed image search with adaptive feature selection framework, is quite successful to address the problem of cross-domain image retrieval which is a challenging task, where the images are matched between the different characteristics of image domain. The cross-domain image matching scheme may be very useful for the following applications, which we have explored:

First, exploration of a digitized photographic museum collection: The FII image search engine is applied to the cultural photographic collections of a French museum, where it has demonstrated its potential for the exploration and promotion of these contents at different levels from their archiving up to their exhibition in the museum and outside, and their linking with other categories of contents, such as geographical mapping contents. Not only that, FII image search engine is incorporated in the POEME image exploration system. POEME image exploration system is a digital installation of an immersive environment, that invents new ways to explore historical, cultural photography collections using CBIR. Thus, the successful implementation of POEME system obtains a synergy making more natural, richer and better personalized the navigation and interaction of users in cultural photographic collections.

Second, cross-domain image localization by inter-linking the image contents: The problem of cross-domain image retrieval and localization, *i.e.,* the ability to estimate a position of a landmark from visual content acquired under various conditions, such as old photographs, paintings, photos were taken at a particular season, etc., is addressed in this proposal. The cross-domain localization problem is addressed by using FII image search engine which is able to perform feature combinations adaptively for each image, by training a regression model on the spatial complementarity of the descriptions. The proposed cross-domain localization strategy reduces the outliers for precise post estimation which is carried out in a very large geographical area. Not only that, the capability of localizing cross-domain content from artwork, multimedia, etc. opens the opportunity to link content with map databases and provide new tools for the promotion of visual and geographical content in various domains including culture, tourism, history, social sciences, etc.

In a nutshell, the entire work is illustrated in the below Fig. 6.1.

6.2 Perspectives

We have designed and implemented the FII image search engine with effective results in CBIR. So, where can we go from here? Is there any room for improvement in the image retrieval system? Or in which possible directions the proposed work could be expanded? We point out some future prospects of our work.

First: Nowadays, the literature on image features, *i.e.,* detectors and descriptors, is very rich. It provides several families to describe different image characteristics for different targets. Be it conventional hand-crafted features or lately popular deep learning features, the number of different image feature types are quite large. It is not always feasible to experiment with all variety of features or features combinations due to the time and resource constraints. Thus, one of the plausible ways to expand this work is to include additional new and diverse image features during the image retrieval related experiments. In this work, we focus on the combinations of the conventional hand-crafted features. It would be quite interesting to consider using deep learning features in the existing image retrieval framework. Also, the fusion of conventional features with deep features could achieve exciting performances and research in this direction needs to be explored. This is our short term perspective.

Second: Although our image search engine is capable and efficient enough to handle a volume of images, still there are opportunities to make the system further productive. We can work in two directions to achieve this mid term perspective. First, nowadays the sophisticated physical infrastructure and coding languages, such as parallel computation in clouds, GPU/CUDA programing in OpenCL, OpenGL, SYCL frameworks, the latest C++17 standard, Scala, Python for Apache Spark, etc., are available. These advanced infrastructures and coding frameworks could be incorporated in the existing image search engine framework and it will make the search en-

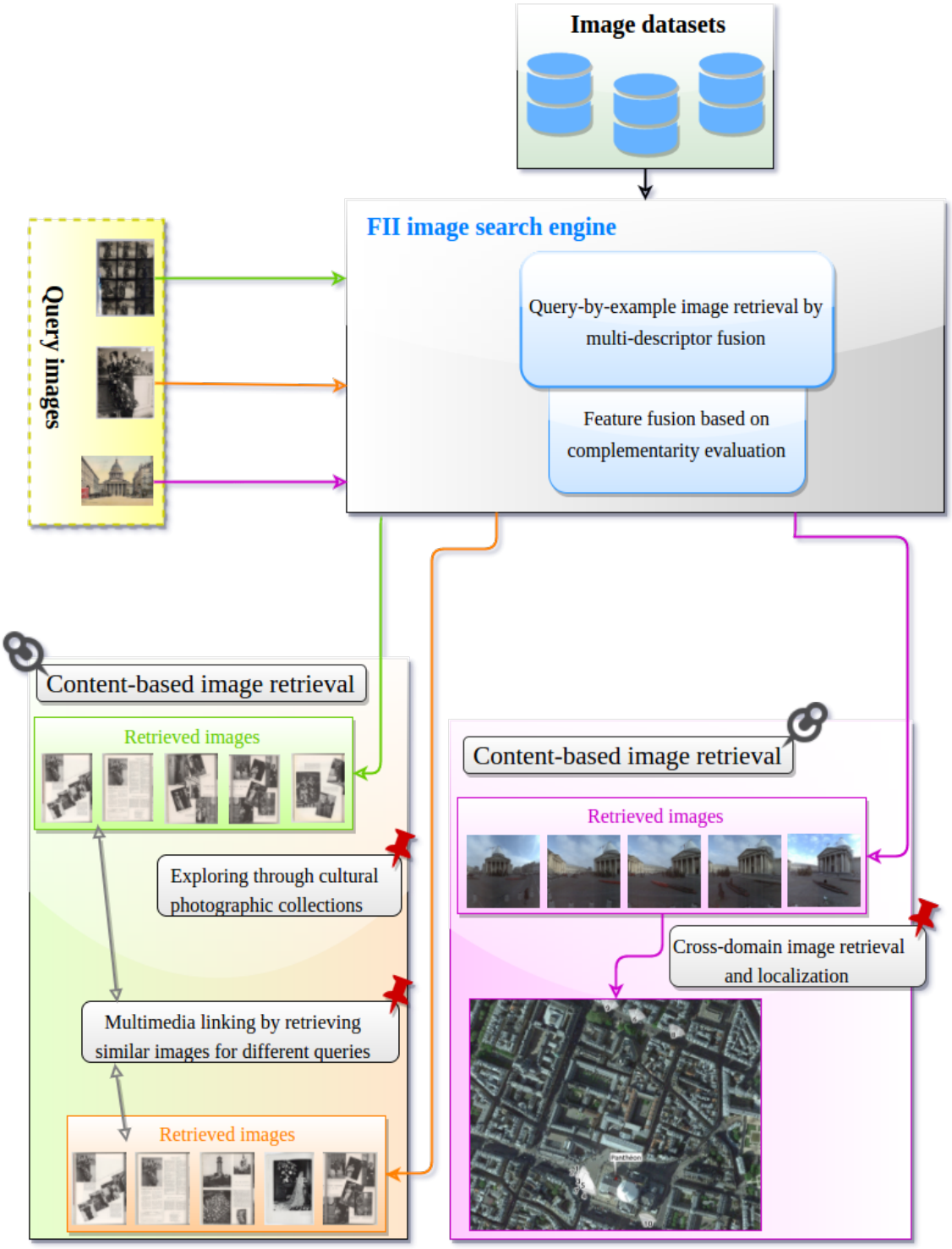


Figure 6.1: Objectives, contributions and applications of the thesis work.

gine to potentially capable enough to manage billions of images. The second interesting option is to adapt our image search engine for mobile applications. In other words, it would be interesting if we can create a mobile application from the proposed image search engine. Therefore, the existing framework needs to be re-designed and adapted according to the different mobile operating systems, *i.e.,* Android, iOS, Windows.

Third: The third and long term perspective is about the impact of this work on different fronts. For example, in the application of image-based localization to estimate the pose of a landmark involving image retrieval step, our FII search framework has a strong impact. Nowadays, the georeferenced datasets are becoming more complex and increasing in size quite rapidly due to the use of the sophisticated image capturing techniques. One of the examples is Stereopolis mobile mapping system, which captures street-level georeferenced images over different places, of French Mapping Agency (IGN). As the georeferenced databases cover very large area, the localization of a landmark needs to be estimated from a very large volume of the database. Hence the process becomes complex. One of the major bottlenecks in this scenario is how to remove the outliers or unrelated images to the query from the database to improve the precision of the retrieval system. This bottleneck can be tackled quite effectively by improving content representation using our proposed image retrieval framework. There is further scope for improvement in the direction of content representation of structured contents, such as urban structures acquired at IGN. One possibility is to use different characteristics of features, such as deep features. Not only that, the spatial relationship between the features, such as in geometric burstiness problem, is an important aspect of content representation. Still more research need to be conducted in this direction and incorporating different solutions in our framework could be profitable for image retrieval. It would also be interesting to apply visual saliency approach, which filters out irrelevant features and it already has been proven beneficial for image retrieval. IGN has a large volume of georeferenced images and in our work, we used the images which covered approximately 51 km of geographical area in Paris. In future, it would be interesting to include more and more georeferenced images by expanding the search area all over the Paris or France. Additionally, the image retrieval framework could be useful for several other localization related applications, such as image-based navigation, landmark-based travel recommendation, etc.

Finally, how our work could be beneficial for the society? We already discussed the usefulness of the image search engine in the context of cultural photographic collections, where it has demonstrated its potential for the exploration and promotion of these contents at different levels from their archiving up to their exhibition in the museum and outside, and their linking with other categories of contents, such as geographical mapping contents. The inter/intra-linking between image contents, such as old and new images, by cross-domain image matching, is one of the efficient and easy ways to understand the evolution in our society, changes in architecture, geography, cultures, history over the periods of time. The more use of cross-domain image retrieval applications will help us to know the past and as well as to understand the present. Apart from that, image search engine could be useful in educational institutes, such as in schools, universities, where learners will be able to link and browse through multimedia easily and quickly.

Appendices

Appendix A

Content-based image retrieval tools

There are several content-based image search engines available. In this appendix, we list down (in the Tables A.1, A.2, and A.3), and A.4) notable CBIR search engines available for commercial and research purpose.

Tool name	Description	Usage type	Link
Google Images	Google's image search engine: search by image and text	Commercial / Public	https://images.google.com/
Pixolution	Search by text, color, logo	Commercial / Private	http://demo.pixolution.de/
Picalike	Visually similar product search in eCommerce	Commercial / Private	http://www.picalike.com/
Elastic vision	Image search tool with content-based clustering images and documents	Commercial / Private	http://elastic-vision.software.informer.com/
Yahoo image	Search by text, color	Commercial / Public	https://images.search.yahoo.com/
Bing image	Search by text, color	Commercial / Public	https://www.bing.com/images
Yandex Image	Search by image and text	Commercial / Public	https://yandex.com/images/
Querbie	Search by query image	Commercial / Private	http://www.querbie.com/
Oddconcepts	Search engine to find eCommerce products	Commercial / Private	https://oddconcepts.kr/

Table A.1: List of available CBIR tools: Part 1.

Tool name	Description	Usage type	Link
Picscout	QbE search to track image usage on web	Commercial / Private	https://picscout.com/
Tineye	Reverse image search engine	Commercial / Private	https://www.tineye.com/
Shopachu	CBIR search engine for fashion and shopping	Commercial / Private	http://www.shopachu.com/
Empora	Fashion product search engine	Commercial / Private	http://www.empora.com/
Piximilar	Search by color from Flickr database	Commercial / Private	http://research.cs.wisc.edu/vision/piximilar/
Chic engine	CBIR engine for fashion products	Commercial / Private	http://www.chicengine.com/
Imense image search	Analysis, search and annotation of digital images and video	Commercial / Private	http://imense.com/
Idmypill	Pill/medicine identification search engine	Commercial / Private	http://www.idmypill.com/
Baidu image	Search by image and text	Commercial / Public	http://image.baidu.com/
Akiwi	Semi-automatic image tagging system to suggest keywords for uploaded images	Research / University	http://www.akiwi.eu/
SIMPLIcity / ALIPR	image retrieval engine and real-time automatic image annotation system	Research / University	http://wang.ist.psu.edu/docs/home.shtml
Anaktisi	QbE by selecting visual descriptor	Research / University	http://orpheus.ee.duth.gr/anaktisi/
BRISC	Framework for texture feature extraction and similarity comparison of computed tomography images	Research / University	http://brisc.sourceforge.net/
PIBE	Adaptive image browsing system to provide users with an intuitive, easy-to-use, structured view of the images in a collection	Research / University	http://www-db.deis.unibo.it/PIBE/

Table A.2: List of available CBIR tools: Part 2.

Tool name	Description	Usage type	Link
Shiatsu	Framework for the management of large video collections	Research / University	http://www-db.deis.unibo.it/Shiatsu/
Windsurf	Wavelet-based indexing of images using region fragmentation	Research / University	http://www-db.deis.unibo.it/Windsurf/
Viral	Visually similar image search from the database and estimates its location using BoF model	Research / University	http://viral.image.ntua.gr/
Quicklook	Similarity search by relevance feedback	Research / University	http://projects.ivl.disco.unimib.it/quicklook/main.html
Pixcavator	Search by compareing distributions of objects (signatures)	Research / University	http://inperc.com/wiki/index.php?title=Pixcavator_image_search
PIRIA	Search by comparing color, texture, shape features	Research / Institute	http://www.kalisteo.org/demo/piria/
Pastec	Image recognition index and search engine for your mobile apps	Research / University	http://www.pastec.io/
MBrowser	Multimedia browsing and retrieval with visual rendering, browsing, QbE by retrieval, video summarisation	Research / University	http://muvis.cs.tut.fi/
Mifile	Image search and annotation using deep features	Research / University	http://mifile.deepfeatures.org/
Imsearch	Image search by image and text	Research / University	http://lucignolo.isti.cnr.it/
Lire	Retrieve images and photos based on color and texture	Research / University	http://www.lire-project.net/
IOSB	QbE image search software	Research / Institute	http://www.iosb.fraunhofer.de/servlet/is/28046/
img (Rummager)	Search by feature comparison and combining keyword and visual similarity	Research / University	http://chatzichristofis.info/?page_id=213

Table A.3: List of available CBIR tools: Part 3.

Tool name	Description	Usage type	Link
Fire	QbE search using different image features and texual information	Research / University	http://thomas.deselaers.de/fire/
Caliph & Emir	Digital photo and image annotation using using MPEG-7 descriptors and retrieval	Research / University	http://www.semanticmetadata.net/features/

Table A.4: List of available CBIR tools: Part 4.

Appendix B

Examples from Chapter 3

This appendix presents the examples of different steps involved in fusion of inverted indices search image engine, which is proposed in the Chapter 3.

B.1 Inverted unique indices example

In this appendix, we discuss about the example of inverted unique indices (IUI) creation which is presented in the Sec. 3.4.1.3 of Chapter 3.

Let us consider a dataset of labeled descriptors (consisting of 3 descriptors from 3 different images) and, their respective codebook with 3 codewords.

Labeled descriptors dataset	
Description ids	Image Ids - Descriptions
q_1	$im1$ - 15 98 43
q_2	$im1$ - 22 99 61
q_3	$im1$ - 27 02 95
q_4	$im2$ - 62 05 72
q_5	$im2$ - 51 54 21
q_6	$im2$ - 69 01 04
q_7	$im3$ - 61 98 80
q_8	$im3$ - 88 38 37
q_9	$im3$ - 44 12 14

Codebook	
Codewords	Descriptions
cw_1	32.67 98.33 61.33
cw_2	44.50 03.50 83.50
cw_3	63.00 26.25 19.00

The nearest codeword for each description is find, then each description image id is added to the matched codeword.

Description ids	Codewords	Distances
$q_1{}^{im1}$	cw_1	648.33
$q_2{}^{im1}$	cw_1	114.33
$q_3{}^{im1}$	cw_2	440.75
$q_4{}^{im2}$	cw_2	440.75
$q_5{}^{im2}$	cw_3	918.06
$q_6{}^{im2}$	cw_3	898.53
$q_7{}^{im3}$	cw_1	1151.33
$q_8{}^{im3}$	cw_3	1087.06
$q_9{}^{im3}$	cw_3	589.06

With the above information, the inverted index algorithm will build a table with a sequence of image ids associated to one codeword, *i.e.,* one list of image ids can have repeated elements. The below table presents the results (the distances between the image and the codeword are in gray color).

Codewords	Inverted Index
cw_1	$im1^{(648.33)}$, $im1^{(114.33)}$, $im3^{(1151.33)}$
cw_2	$im1^{(440.75)}$, $im2^{(440.75)}$
cw_3	$im2^{(918.06)}$, $im2^{(896.56)}$, $im3^{(1087.06)}$, $im3^{(589.06)}$

The inverted unique indices algorithm associates each codeword to a set of image ids. The final IUI file is presented below.

Codewords	Inverted Unique Indices
cw_1	$im1^{(114.334)}$, $im3^{(1151.33)}$
cw_2	$im1^{(440.750)}$, $im2^{(440.750)}$
cw_3	$im2^{(898.562)}$, $im3^{(589.062)}$

B.2 Search k-Nearest neighbor example

In this appendix, we discuss about the example of k nearest neighbor search which is explained in the Sec. 3.4.2.1 of Chapter 3.

Let us consider a query descriptions, which is consisting of 2 descriptions, and the codebook with 3 codewords are given below.

Query image	
Description ids	Descriptions
q_1	15 98 43
q_2	22 99 61

Codebook	
Codewords	Descriptions
cw_1	32.67 98.33 61.33
cw_2	44.50 03.50 83.50
cw_3	63.00 26.25 19.00

Let us consider, the k-NN value is 2, *i.e.,* SearchkNN will look for 2 closest codewords to the each query description. The output of the SearchkNN is given below:

k-NN List	Codewords	Distances
$kNNL_1$	cw_1	25.46
	cw_3	89.60
$kNNL_2$	cw_1	10.70
	cw_3	93.47

We must also normalize the $kNNL$ by dividing the distance by the estimated $Dist_{max}$ ($= 93.47$) distance. The final normalized k-NN lists are computed as:

k-NN List	Codewords	Distances
$kNNL_1$	cw_1	0.272
	cw_3	0.958
$kNNL_2$	cw_1	0.114
	cw_3	1.000

B.3 Candidate list creation example

In this appendix, we discuss the example of candiate list ceation which is explained in the Sec. 3.4.2.2 of Chapter 3.

In order to create a candidate list ($CList$), we must have a k-NN list and the corresponding inverted unique indices (IUI) to the k-NN list codewords.

k-NN list	
Codewords	Distance
cw_3	0.50
cw_2	0.65
cw_5	0.69
cw_1	0.77
cw_6	0.82
cw_8	0.89

Codewords	Inverted Unique Indices
cw_1	$im01^{(154.702)}, im12^{(259.876)}, im25^{(779.210)}$
cw_2	$im13^{(440.750)}, im25^{(440.750)}$
cw_3	$im01^{(114.334)}, im12^{(851.330)}, im18^{(902.435)}$
cw_4	$im26^{(541.542)}$
cw_5	$im02^{(898.562)}, im18^{(589.062)}, im25^{(702.612)}$
cw_6	$im22^{(890.135)}, im45^{(983.678)}$
cw_7	$im03^{(213.210)}$
cw_8	$im19^{(240.125)}, im02^{(289.740)}$
cw_9	$im22^{(432.870)}, im01^{(534.422)}$
cw_{10}	$im04^{(199.238)}, im09^{(211.890)}, im12^{(298.112)}$

Lets consider the variable $T = 7$ as a reference of the approximate maximum number of image ids to be included. The candidate list creation algorithm is exemplified in the following iterations:

Before Iterating

$imageIds = \{\varnothing\}$

Iteration 1

$$\begin{aligned} imageIds_{cw_3} &= imageIds_{cw_3} - (imageIds \cap imageIds_{cw_3}) \\ &= \{im01, im12, im18\} - (imageIds \cap \{im01, im12, im18\}) \\ &= \{im01, im12, im18\} \\ imageIds &= imageIds \cup imageIds_{cw_3} \\ &= \{\varnothing\} \cup \{im01, im12, im18\} \\ &= \{im01, im12, im18\} \end{aligned}$$

$Candidatelist$		
Codewords	Distance	ImageIds
cw_3	0.50	$im01, im12, im18$

As the for the first iteration the imageIds set was empty, the algorithm will consider all the image ids.

Iteration 2

$$
\begin{aligned}
imageIds_{cw_2} &= imageIds_{cw_2} - (imageIds \cap imageIds_{cw_2}) \\
&= \{im13, im25\} - (imageIds \cap \{im13, im25\}) \\
&= \{im13, im25\} \\
imageIds &= imageIds \cup imageIds_{cw_2} \\
&= \{im01, im12, im18\} \cup \{im13, im25\} \\
&= \{im01, im12, im18, im13, im25\}
\end{aligned}
$$

$Candidatelist$		
Codewords	Distance	ImageIds
cw_3	0.50	$im01, im12, im18$
cw_2	0.65	$im13, im25$

As none of the image ids from cw_2 were in the intersection the entire set of the cw_2 is kept.

Iteration 3

$$
\begin{aligned}
imageIds_{cw_5} &= imageIds_{cw_5} - (imageIds \cap imageIds_{cw_5}) \\
&= \{im02, im18, im25\} - (imageIds \cap \{im02, im18, im25\}) \\
&= \{im02\} \\
imageIds &= imageIds \cup imageIds_{cw_5} \\
&= \{im01, im12, im18, im13, im25\} \cup \{im02\} \\
&= \{im01, im12, im18, im13, im25, im02\}
\end{aligned}
$$

$Candidatelist$		
Codewords	Distance	ImageIds
cw_3	0.50	$im01, im12, im18$
cw_2	0.65	$im13, im25$
cw_5	0.69	$im02$

As $im18$ and $im25$ from cw_5 are in the intersection, only $im02$ is included.

Iteration 4

$$
\begin{aligned}
imageIds_{cw_1} &= imageIds_{cw_1} - (imageIds \cap imageIds_{cw_1}) \\
&= \{im01, im12, im25\} - (imageIds \cap \{im01, im12, im25\}) \\
&= \{\varnothing\} \\
imageIds &= imageIds \cup imageIds_{cw_1} \\
&= \{im01, im12, im18, im13, im25, im02\} \cup \{\varnothing\} \\
&= \{im01, im12, im18, im13, im25, im02\}
\end{aligned}
$$

$Candidatelist$		
Codewords	Distance	ImageIds
cw_3	0.50	$im01, im12, im18$
cw_2	0.65	$im13, im25$
cw_5	0.69	$im02$

As all the image ids from cw_1 are already in the intersection, this codeword is not included in the candidate list.

Iteration 5

$$
\begin{aligned}
imageIds_{cw_6} &= imageIds_{cw_6} - (imageIds \cap imageIds_{cw_6}) \\
&= \{im22, im45\} - (imageIds \cap \{im22, im45\}) \\
&= \{im22, im45\} \\
imageIds &= imageIds \cup imageIds_{cw_1} \\
&= \{im01, im12, im18, im13, im25, im02\} \cup \{im22, im45\} \\
&= \{im01, im12, im18, im13, im25, im02, im22, im45\}
\end{aligned}
$$

$Candidatelist$		
Codewords	Distance	ImageIds
cw_3	0.50	$im01, im12, im18$
cw_2	0.65	$im13, im25$
cw_5	0.69	$im02$
cw_6	0.82	$im22, im45$

As none of the image ids from cw_6 were in the intersection the entire set of the cw_6 is kept. So the number of images id in the candidate list will be slightly greater than T.

In the k-NN list, there is still one codeword, cw_8, to add but it is not included in the candidate list as the number of image ids already reaches it's maximum limit.

B.4 MultiSequence pair algorithm example

In this appendix, we discuss about the example of MultiSequence pair algorithm which is explained in Sec. 3.4.2.3 of Chapter 3.

Let us consider two candidate lists ($CList$), denoted by $List_u$ and $List_v$ with the size of 5 and 4 elements, as shown below:[1]

u	$List_u$	
	Element	Distance
1	El_1	1.0
2	El_2	2.2
3	El_3	4.1
4	El_4	5.7
5	El_5	6.0

v	$List_v$	
	Element	Distance
1	El_1	1.3
2	El_2	2.5
3	El_3	3.0
4	El_4	3.6

Then we can imagine of a matrix which contains the sum of the distances of all vs all elements from both lists.[2]

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Let us define $dist(u, v)$ as the value of the summed distances of the element in positions u and v of their respective $List_u$ and $List_v$. Also, we are denoting u' and v' as predecessor indices of u and v, this mean $u' \leqslant u$ and $v' \leqslant v$. As both candidate list elements are ordered by their increasing distances we can conclude that: $dist(u, v) \geqslant dist(u', v')$.

The previous conclusion leads us to observe that in order to consider the pair(u, v) to be included in the *Final List* all the pairs(u', v')[3] must have been already included. However, instead of checking if every pair (u', v') has been inserted in the *Final List*, for each new candidate pair(u, v) we can check if the immediate predecessors $(u - 1, v)$ and $(u, v - 1)$ have been already accessed or not.

In order to check if the immediate predecessors of pair(u, v) were accessed we use the vector of indices $LastIdx_v$ containing many slots as the size of $List_u$. Therefore, the last pairs, per each column, accessed in the matrix are represented in the $LastIdx_v$ vector by the index of

[1] We do not need to consider the candidate lists corresponding image ids for this example.

[2] Notice that this sum of distances is just done for didactic purposes. Actually the algorithm is only doing the sum of distances **after** a pair is selected, so it is not a criteria to select it.

[3] Excluding the case in which $u = u'$ and $v = v'$

the slot (u being the column) and the value contained in that index (v being the row). Finally for any pair(u, v) we will know if their immediate predecessors were accessed by verifying if $LastIdx_v[u-1] = v-1$ and if $LastIdx_v[u+1] \geqslant v+1$.

Example

For the following iterations the matrix distance's values will be remarked with different colors meaning:

- Red: The pair has already been popped from *Dists* and considered in the *Final List*.
- Green: The pair is pushed to the *Dists* and considered as a candidate to be inserted into the *Final List*.
- Gray: The pair is not even reached by the algorithm, and probably would never be reached.

Before Iterating

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

In order to start, the MultiSequence algorithm must push to the *Dists* queue the pair $(1, 1)$ as it refers to the shortest distance.

LastIdx_v

1	2	3	4	5
0	0	0	0	0

Dists

u	v	$dist$
1	1	2.3

Final List

u	v	$dist$
$\varnothing$	$\varnothing$	$\varnothing$

Iteration 1

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v): $(1, 1)$

	1	2	3	4	5
LastIdx_v	1	0	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(2, 1)$ and $(u, v + 1)$:$(1, 2)$

- Pair$(2, 1)$ is considered
  $u \leqslant 5$ and $v = 1$ so there is no Pair$(u + 1, v - 1)$:$(2, 0)$ to check.
- Pair$(1, 2)$ is considered
  $v \leqslant 4$ and $u = 1$ so there is no Pair$(u - 1, v + 1)$:$(0, 2)$ to check.

Dists

u	v	$dist$
2	1	3.5
1	2	3.5

Final List

u	v	$dist$
1	1	2.3

Iteration 2

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v): $(2, 1)$

	1	2	3	4	5
LastIdx_v	1	1	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(3, 1)$ and $(u, v + 1)$:$(2, 2)$

- Pair$(3, 1)$ is considered
  $u \leqslant 5$ and $v = 1$ so there is no Pair$(u + 1, v - 1)$:$(3, 0)$ to check.
- Pair$(2, 2)$ is not considered
  $v \leqslant 4$ but the Pair$(u-1, v+1)$:$(1, 2)$ is not in the final list, that we know because the last pair inserted in column 1 $(u-1)$ is not 2 $(v+1)$ as expected but 1 (LastIdx_v$[2-1] \neq 1 + 1$).

Dists

u	v	$dist$
1	2	3.5
3	1	5.4

Final List

u	v	$dist$
u	v	$dist$
1	1	2.3
2	1	3.5

Iteration 3

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v): $(1, 2)$

	1	2	3	4	5
LastIdx_v	2	1	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(2, 2)$ and $(u, v + 1)$:$(1, 3)$

- Pair$(2, 2)$ is considered
  $u \leqslant 5$ and Pair$(u + 1, v - 1)$:$(2, 1)$ is already in the list (LastIdx_v$[1 + 1] = 2 - 1$).
- Pair$(1, 3)$ is considered
  $v \leqslant 4$ and $u = 1$ so there is no Pair$(u - 1, v + 1)$:$(0, 3)$ to check.

Dists

u	v	$dist$
1	3	4.0
2	2	4.7
3	1	5.4

Final List

u	v	$dist$
1	1	2.3
2	1	3.5
1	2	3.5

Observe: The $Dists$ queue has reordered its elements in function of the distances.

Iteration 4

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v): $(1, 3)$

	1	2	3	4	5
LastIdx_v	3	1	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(2, 3)$ and $(u, v + 1)$:$(1, 4)$

- Pair$(2, 3)$ is not considered
  $u \leqslant 5$ but the Pair$(u+1, v-1)$:$(2, 2)$ is not in the final list, that we know because the last pair inserted in column 2 $(u+1)$ is not 2 $(v-1)$ as expected but 1 (LastIdx_v$[1+1] \neq 3 - 1$).
- Pair$(1, 4)$ is considered
  $v \leqslant 4$ and $u = 1$ so there is no Pair$(u - 1, v + 1)$:$(0, 3)$ to check.

Dists

u	v	$dist$
1	4	4.6
2	2	4.7
3	1	5.4

Final List

u	v	$dist$
1	1	2.3
2	1	3.5
1	2	3.5
1	3	4.0

Iteration 5

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v):$(1, 4)$

	1	2	3	4	5
LastIdx_v	4	1	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(2, 4)$ and $(u, v + 1)$:$(1, 5)$

- Pair$(2, 4)$ is not considered
  $u \leqslant 5$ but the Pair$(u+1, v-1)$:$(2, 3)$ is not in the final list, that we know because the last pair inserted in column 2 $(u+1)$ is not 3 $(v-1)$ as expected but 1 (LastIdx_v$[1+1] \neq 4 - 1$).
- Pair$(1, 5)$ is not considered
  $v \nleqslant 4$.

Dists

u	v	$dist$
2	2	4.7
3	1	5.4

Final List

u	v	$dist$
1	1	2.3
2	1	3.5
1	2	3.5
1	3	4.0
1	4	4.6

Iteration 6

$v \backslash u$	1	2	3	4	5
1	2.3	3.5	5.4	7.0	7.3
2	3.5	4.7	6.6	8.2	8.5
3	4.0	5.2	7.1	8.7	9.0
4	4.6	5.8	7.7	9.3	9.6

Popped Pair(u, v): $(2, 2)$

	1	2	3	4	5
LastIdx_v	4	2	0	0	0

Pairs to analyze for insertion $(u + 1, v)$:$(3, 2)$ and $(u, v + 1)$:$(2, 3)$

- Pair$(3, 2)$ is not considered
  $u \leqslant 5$ but the Pair$(u+1, v-1)$:$(3, 1)$ is not in the final list, that we know because the last pair inserted in column 3 $(u+1)$ is not 1 $(v-1)$ as expected but 0 (LastIdx_v$[2+1] \neq 2 - 1$).
- Pair$(2, 3)$ is considered
  $v \leqslant 4$ and Pair$(u - 1, v + 1)$:$(1, 3)$ is already in the list (LastIdx_v$[2 - 1] \geqslant 2 + 1$)

Dists

u	v	$dist$
2	3	5.2
3	1	5.4

Final List

u	v	$dist$
1	1	2.3
2	1	3.5
1	2	3.5
1	3	4.0
1	4	4.6
2	2	4.7

Observation: As we are using a LastIdx_v vector the condition for `line 13` in the $MultiSequencePair$ algorithm must be "$\geqslant$" and not just "$=$" as it is in `line 11`. If we would be using a LastIdx_u vector instead then the condition for `line 11` in the $MultiSequencePair$ algorithm must be "$\geqslant$" and "$=$" for `line 13`. In this 6th iteration it is show why just an "$=$" comparison in `line 13` would not be correct.

B.5 Voting algorithm example

In this section, we discuss about the example of voting algorithm which is explained in Sec. 3.4.2.4 of Chapter 3.

Let us consider a final list, FL, composed by fl^1_{xy} and fl^2_{xy}, which are depicted in the table below.

fl^1_{xy}		
Element	Distance	Image Ids
$cw_x1 - cw_y1$	0.50	$im1, im5$
$cw_x1 - cw_y2$	0.65	$im2$
$cw_x2 - cw_y1$	0.70	$im6$
$cw_x1 - cw_y3$	0.85	$im3$

fl^2_{xy}		
Element	Distance	Image Ids
$cw_x5 - cw_y6$	0.30	$im4, im6$
$cw_x5 - cw_y9$	0.45	$im2, im1$
$cw_x5 - cw_y3$	0.70	$im5$
$cw_x3 - cw_y7$	0.95	$im3$

Next, the respective weights are coputed depends on the distance associated with each image ids. The weight lists are generated as below.

wl^1_{xy}		
Element	Weight	Image Ids
$cw_x1 - cw_y1$	0.50 (1.00-0.50)	$im1, im5$
$cw_x1 - cw_y2$	0.35 (1.00-0.65)	$im2$
$cw_x2 - cw_y1$	0.30 (1.00-0.70)	$im6$
$cw_x1 - cw_y3$	0.15 (1.00-0.85)	$im3$

wl^2_{xy}		
Element	Weight	ImageIds
$cw_x5 - cw_y6$	0.70 (1.00-0.30)	$im5, im6$
$cw_x5 - cw_y9$	0.55 (1.00-0.45)	$im2, im1$
$cw_x5 - cw_y3$	0.30 (1.00-0.70)	$im4$
$cw_x3 - cw_y7$	0.05 (1.00-0.95)	$im3$

Once the weights are computed, the corresponding frequency of the each images from the both lists are computed in the frequency list, FqL, which is presented as follow:

FqL	
Image Id	Frequency
$im1$	0.525 ($\frac{0.50+0.55}{2}$)
$im2$	0.450 ($\frac{0.35+0.55}{2}$)
$im3$	0.100 ($\frac{0.15+0.05}{2}$)
$im4$	0.150 ($\frac{\varnothing+0.30}{2}$)
$im5$	0.600 ($\frac{0.50+0.70}{2}$)

Finally, the frequency list is sorted from the most to the least similar image according to the frequencies.

Sorted FqL	
Image Id	Frequency
$im5$	0.600
$im1$	0.525
$im2$	0.450
$im4$	0.150
$im3$	0.100

Appendix C

Examples from Chapter 4

In this appendix, we present the examples of spatial complementarity evaluation criteria, which is presented in the Chapter 4.

C.1 Example of spatial coverage complementarity

In this appendix, we present the examples of spatial coverage complementarity which is presented in the Sec. 4.2.1 of Chapter 4. Different scenarios are explained through examples.

Scenario 1.

Let us consider detector D_a and D_b have detected $n = 3$ and $m = 2$ different set of points respectively in an image (Im) as shown in below Table. The values in the parenthesis indicate x and y coordinates of the detected points.

Detector	Points
D_a	$d_a^1(5,8), d_a^2(34,10), d_a^3(8,9)$
D_b	$d_b^1(8,20), d_b^2(11,13)$

Next, we consider a keypoint as a reference point and calculate $(n+m-1) = 4$ distances with other points. We consider each point as a reference point.

Distances	
$ED^{d_a 1}$	$29.068, 3.162, 12.369, 6.708$
$ED^{d_a 2}$	$29.068, 26.019, 27.856, 23.194$
$ED^{d_a 3}$	$3.162, 26.019, 11, 5$
$ED^{d_b 1}$	$12.369, 27.856, 11, 7.615$
$ED^{d_b 2}$	$7.810, 23.194, 5, 7.615$

The mean of the distances are computed as below:

Mean Distances	
$EDMean_{nm}^{d_a1}$	6.890
$EDMean_{nm}^{d_a2}$	26.341
$EDMean_{nm}^{d_a3}$	6.201
$EDMean_{nm}^{d_b1}$	11.800
$EDMean_{nm}^{d_b2}$	7.960

The distribution complementarity score is computed as

$Di_{cs} = 10.6384$

Scenario 2.

Let us consider detector D_a and D_c have detected $n = 3$ and $m = 2$ points respectively in an image (Im) as shown in below Table. The values in the parenthesis indicate coordinates of the detected points. The points detected by D_c are also detected by D_a.

Detector	Points
D_a	$d_a^1(5,8), d_a^2(34,10), d_a^3(8,9)$
D_c	$d_c^1(5,8), d_c^2(8,9)$

Next we consider one keypoint as a reference point and calculate $(n + m - 1) = 4$ the distances with other points. We consider each point as a reference point. The computed eucledean distances are presented in the below Table.

Distances	
ED^{d_a1}	$19.068, 3.162, 0, 3.162$
ED^{d_a2}	$29.068, 26.019, 29.068, 26.019$
ED^{d_a3}	$3.162, 26.019, 3.162, 0$
ED^{d_c1}	$0, 29.068, 3.162, 3.162$
ED^{d_c2}	$3.162, 26.019, 0, 3.162$

The computed mean of these distances are presented in the below Table.

Mean Distances	
$EDMean_{nm}^{d_a1}$	4.498
$EDMean_{nm}^{d_a2}$	27.472
$EDMean_{nm}^{d_a3}$	4.471
$EDMean_{nm}^{d_c1}$	4.498
$EDMean_{nm}^{d_c2}$	4.471

The distribution complementarity score is computed as

$Di_{cs} = 5.385$

As the D_c detector keypoints are also detected by the D_a detector, therefore the overall score is reduced compared to Scenario 1.

Scenario 3.

Let us consider detector D_a and D_d have detected $n = 3$ and $m = 2$ points respectively in an image (I) as shown in below Table. The values in the in parenthesis indicate coordinates of the detected points. There is one keypoint is common between these descriptors.

Detector	Points
D_a	$d_a^1(5,8), d_a^2(34,10), d_a^3(8,9)$
D_d	$d_d^1(5,8), d_d^2(11,13)$

We consider one keypoint as a reference point and calculate $(n + m - 1) = 4$ distances with other points. We consider each point as a reference point.

Distances	
$ED^{d_a 1}$	$29.068, 3.162, 0, 7.810$
$ED^{d_a 2}$	$29.068, 26.019, 29.068, 23.194$
$ED^{d_a 3}$	$3.162, 29.068, 3.162, 5$
$ED^{d_d 1}$	$0, 29.068, 3.162, 7.810$
$ED^{d_d 2}$	$7.810, 23.194, 5, 7.810$

Next we compute mean of these distances.

Mean Distances	
$EDMean_{nm}^{d_a 1}$	6.266
$EDMean_{nm}^{d_a 2}$	26.604
$EDMean_{nm}^{d_a 3}$	4.592
$EDMean_{nm}^{d_d 1}$	6.267
$EDMean_{nm}^{d_d 2}$	8.012

The distribution complementarity score is computed as

$Di_{cs} = 7.150$

As the number of common point is one, *i.e.,* less number of similar point compare to Scenario 2, therefore the overall coverage score has increased.

C.2 Example of contribution measure

In this appendix, we discussed about the examples of contribution measure which is explained in the Sec. 4.2.2 of Chapter 4. Different scenarios are explained through examples.

Scenario 1.

Let us consider detector D_a and D_b have detected $n = 6$ and $m = 5$ points respectively in an image (Im) as shown in below Table. The values in the parenthesis indicate coordinates of the detected points.

Detector	Points
D_a	$d_a^1(5,8), d_a^2(34,10), d_a^3(8,9), d_a^4(10,21), d_a^5(19,21), d_a^6(19,8)$
D_b	$d_b^1(8,20), d_b^2(11,13), d_b^3(20,20), d_b^4(13,26), d_b^5(9,12)$

As we can see, the detected points location are distinct, *i.e.,* $p = 0$. Therefore contribution of D_b on D_a and vice versa are calculated as

$$Cn_{D_b|D_a} = 1 \quad Cn_{D_a|D_b} = 1$$

The complementarity score can be calculated as,

$$Cn_{cs} = 1$$

The complementarity score is maximum, *i.e.,* 1, due to the non existence of common keypoints detected by the two detectors.

Scenario 2.

Let us consider detector D_a and D_c have detected $n = 6$ and $m = 4$ points respectively in an image (Im) as shown in below Table. The values in the parenthesis indicate coordinates of the detected points.

Detector	Points
D_a	$d_a^1(5,8), d_a^2(34,10), d_a^3(8,9), d_a^4(10,21), d_a^5(19,21), d_a^6(19,8)$
D_c	$d_c^1(8,17), d_c^2(34,10), d_c^3(21,11), d_c^4(11,13)$

As we observe, there is one similar detected point between D_a and D_c, *i.e.,* $p = 1$. Therefore contribution of D_b on D_a and vice versa are calculated as

$$Cn_{D_c|D_a} = 0.83 \quad Cn_{D_a|D_c} = 0.75$$

The complementarity score can be calculated as,

$Cn_{CS} = 0.75$

The overall complementarity score is reduced, compared to Scenario 1, as there is a common keypoint detected by the both detectors.

Scenario 3.

Let us consider, detector D_a and D_e have detected $n = 6$ and $m = 5$ points respectively in an image (Im) as shown in below Table. The values in the parenthesis indicate coordinates of the detected points.

Detector	Points
D_a	$d_a1(5,8), d_a2(34,10), d_a3(8,9), d_a4(10,21), d_a5(19,21), d_a6(19,8)$
D_e	$d_e(8,13), d_e2(34,10), d_e3(18,7), d_e4(10,21), d_e(19,8)$

As we can see, three detected point between D_a and D_e are the same, *i.e.,* $p = 3$. Therefore contribution of D_e on D_a and vice versa are calculated as

$Cn_{D_e|D_a} = 0.50 \quad Cn_{D_a|D_e} = 0.40$

The complementarity score can be calculated as,

$Cn_{cs} = 0.40$

The complementarity score is further reduced, compared to Scenario 1 and 2, with the increasing number of common keypoints detected by the detectors.

C.3 Example of cluster based mesurement

In this section, we discussed about the examples of cluster based complementarity measurement which is explained in the Sec. 4.2.3 of Chapter 4. Different scenarios are explained through examples.

Let us consider detector D_a and D_b have detected 6 and 5 points respectively in an image (I). The number of clusters generated from the points are 4.

Detector	Points
D_a	$d_a^1, d_a^2, d_a^3, d_a^4, d_a^5, d_a^6$
D_b	$d_b^1, d_b^2, d_b^3, d_b^4, d_b^5$

Cluster	
Cl	cl_1, cl_2, cl_3, cl_4

Scenario 1.

The clusters are represented by the points only from either D_a or D_b. For this scenario, we consider the following clustering organization as shown in the below Table.

Cluster	Shared points
c_1	$d_a^2, d_a^3 \| \| D_a = 2, D_b = 0$
c_2	$d_b^3, d_b^4, d_b^5 \| \| D_a = 0, D_b = 3$
c_3	$d_a^1, d_a^4, d_a^4, d_a^6 \| \| D_a = 4, D_b = 0$
c_4	$d_b^1, d_b^2 \| \| D_a = 0, D_b = 2$

Therefore, cluster complementarity score is calculated as,

$$Cl_{cs} = 1 - 2.\frac{1}{4}\sum_{c=1}^{4} min(p_{jD_a}, p_{jD_b}) = 1$$

We observed that, no cluster is shared between two detectors; which implies the points from D_a and D_b are distinct. Hence the complementarity score is maximum, *i.e.,* 1.

Scenario 2.

In this scenario, the clusters are equally shared by D_a and D_b and one cluster (c_4) is shared only by D_a.

Cluster	Shared points
c_1	$d_a^2, d_a^3, d_b^1, d_b^3 \| \| D_a = 2, D_b = 2$
c_2	$d_a^1, d_b^2 \| \| D_a = 1, D_b = 1$
c_3	$d_a^4, d_a^5, d_b^4, d_b^5 \| \| D_a = 2, D_b = 2$
c_4	$d_a^6 \| \| D_a = 1$

Therefore, cluster complementarity score is calculated as,

$$Cl_{cs} = 1 - 2.\frac{1}{4}\sum_{c=1}^{4} min(p_{jD_a}, p_{jD_b}) = 0.25$$

As the clusters are almost equally contributed from the both detectors; that implies the points from the D_1 and D_2 are similar. Hence the complementarity is low.

Scenario 3.

Now, we consider that the clusters are heavily shared by either points from D_a or D_b.

Cluster	Shared points
c_1	$d_a^2, d_a^3, d_b^1 \| \| D_a = 2, D_b = 1$
c_2	$d_a^1, d_b^2, d_b^4 \| \| D_a = 1, D_b = 2$
c_3	$d_a^4, d_a^5 \| \| D_a = 2, D_b = 0$
c_4	$d_a^6, d_b^5 \| \| D_a = 1, D_b = 1$

Therefore, clsuter complementarity score is calculated as,

$$Cl_{cs} = 1 - 2.\frac{1}{4}\sum_{c=1}^{4} min(p_{jD_a}, p_{jD_b}) = 0.42$$

As the clusters are dominated by the points from either D_a or D_b, therefore the complementarity score is better than scenario 2, but smaller from the Scenario 1.

Appendix D

List of publications

D.1 Journal

N. Bhowmik, V. Gouet-Brunet, G. Bloch, and S. Besson. Combination of image descriptors for the exploration of cultural photographic collections. In Journal of Electronic Imaging, 26(1):011019, 2016. doi: 10.1117/1.JEI.26.1.011019. URL: http://electronicimaging.spiedigitallibrary.org/article.aspx?articleid=2594509.

D.2 Conference

N. Bhowmik, Li Weng, V. Gouet-Brunet, and B. Soheilian. Cross-domain image localization by adaptive feature fusion. In 2017 Joint Urban Remote Sensing Event (JURSE), pages 1–4, March 2017. doi: 10.1109/JURSE.2017.7924572.

N. Bhowmik, V. Gouet-Brunet, L. Wei, and G. Bloch. Adaptive and Optimal Combination of Local Features for Image Retrieval, pages 76–88. Springer International Publishing, Cham, 2017. ISBN 978-3-319-51814-5. doi: 10.1007/978-3-319-51814-5_7. URL: https://doi.org/10.1007/978-3-319-51814-5_7.

N. Bhowmik, V. R. González, V. Gouet-Brunet, H. Pedrini, and G. Bloch. Efficient fusion of multidimensional descriptors for image retrieval. In 2014 IEEE International Conference on Image Processing (ICIP), pages 5766–5770, Oct 2014. doi: 10.1109/ICIP.2014.7026166.

D.3 Others

Vulgarization article: Content-based linking of geographic iconographic heritage collections, V. Gouet-Brunet and N. Bhowmik. In CIPA Newsletter 12, May 2017, http://cipa.

icomos.org/portfolio-item/newsletter_12.

Bibliography

A.E. Abdel-Hakim and A.A. Farag. CSIFT: A SIFT Descriptor with Color Invariant Characteristics. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 1978–1983, 2006. 19, 29

Motilal Agrawal, Kurt Konolige, and MortenRufus Blas. Censure: Center surround extremas for realtime feature detection and matching. In David Forsyth, Philip Torr, and Andrew Zisserman, editors, *Computer Vision – ECCV 2008*, volume 5305 of *Lecture Notes in Computer Science*, pages 102–115. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-88692-1. doi: 10.1007/978-3-540-88693-8_8. URL http://dx.doi.org/10.1007/978-3-540-88693-8_8. 26, 102, 117, 140

Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *Computer vision and pattern recognition (CVPR), 2012 IEEE conference on*, pages 510–517. IEEE, 2012. 19, 23, 30

David Aldavert, Arnau Ramisa, Ricardo Toledo, and Ramon Lopez de Mantaras. *Efficient Object Pixel-Level Categorization Using Bag of Features*, pages 44–54. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-10331-5. doi: 10.1007/978-3-642-10331-5_5. URL http://dx.doi.org/10.1007/978-3-642-10331-5_5. 46

A. Alzu'bi, A. Amira, and N. Ramzan. Compact root bilinear cnns for content-based image retrieval. In *2016 International Conference on Image, Vision and Computing (ICIVC)*, pages 41–45, Aug 2016. doi: 10.1109/ICIVC.2016.7571271. 34

Ahmad Alzu'bi, Abbes Amira, and Naeem Ramzan. Semantic content-based image retrieval: A comprehensive study. *Journal of Visual Communication and Image Representation*, 32:20 – 54, 2015. ISSN 1047-3203. doi: http://dx.doi.org/10.1016/j.jvcir.2015.07.012. URL //www.sciencedirect.com/science/article/pii/S1047320315001327. 6

A. Amato and V. Di Lecce. Edge detection techniques in image retrieval: the semantic meaning of edge. In *Proceedings EC-VIP-MC 2003. 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications (IEEE Cat. No.03EX667)*, volume 1, pages 143–148 vol.1, July 2003. doi: 10.1109/VIPMC.2003.1220453. 54

R. Arandjelovic and A. Zisserman. All about vlad. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1578–1585, June 2013. doi: 10.1109/CVPR.2013.207. 34

R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2911–2918, June 2012. doi: 10.1109/CVPR.2012.6248018. 49

Nafiz Arica and Fatos T. Yarman Vural. Bas: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters*, 24(9–10):1627 – 1639, 2003. ISSN 0167-8655. doi: http://doi.org/10.1016/S0167-8655(03)00002-3. URL http://www.sciencedirect.com/science/article/pii/S0167865503000023. 44

C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit. Instant outdoor localization and slam initialization from 2.5d maps. *IEEE Transactions on Visualization and Computer Graphics*, 21(11):1309–1318, Nov 2015. ISSN 1077-2626. doi: 10.1109/TVCG.2015.2459772. 162

Mathieu Aubry, Bryan C. Russell, and Josef Sivic. Painting-to-3d model alignment via discriminative visual elements. *ACM Trans. Graph.*, 33(2):14:1–14:14, April 2014. ISSN 0730-0301. doi: 10.1145/2591009. URL http://doi.acm.org/10.1145/2591009. 19, 138

S. Avila, N. Thome, M. Cord, E. Valle, and A. de A. Araújo. Bossa: Extended bow formalism for image classification. In *2011 18th IEEE International Conference on Image Processing*, pages 2909–2912, Sept 2011. doi: 10.1109/ICIP.2011.6116268. 36

Sandra Avila, Nicolas Thome, Matthieu Cord, Eduardo Valle, and Arnaldo de A. Araújo. Pooling in image representation: The visual codeword point of view. *Computer Vision and Image Understanding*, 117(5):453 – 465, 2013. ISSN 1077-3142. doi: http://dx.doi.org/10.1016/j.cviu.2012.09.007. URL //www.sciencedirect.com/science/article/pii/S1077314212001737. 18, 36, 37

Mohammad Awrangjeb and Guojun Lu. A robust corner matching technique. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 1483–1486. IEEE, 2007. 28

P. Azad, T. Asfour, and R. Dillmann. Combining harris interest points and the sift descriptor for fast scale-invariant object recognition. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4275–4280, Oct 2009. doi: 10.1109/IROS.2009.5354611. 61

A. Babenko and V. Lempitsky. The inverted multi-index. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3069–3076, 2012. doi: 10.1109/CVPR.2012.6248038. 11, 18, 67, 68, 69, 70, 88, 92

Artem Babenko, Anton Slesarev, Alexander Chigorin, and Victor S. Lempitsky. Neural codes for image retrieval. *CoRR*, abs/1404.1777, 2014. URL http://arxiv.org/abs/1404.1777. 137

Kobus Barnard, Pinar Duygulu, David Forsyth, Nando de Freitas, David M Blei, and Michael I Jordan. Matching words and pictures. *Journal of machine learning research*, 3(Feb):1107–1135, 2003. 5

Jorge Alberto Marcial Basilio, Gualberto Aguilar Torres, Gabriel Sánchez Pérez, L. Karina Toscano Medina, and Héctor M. Pérez Meana. Explicit image detection using ycbcr space color model as skin detection. In *Proceedings of the 2011 American Conference on Applied Mathematics and the 5th WSEAS International Conference on Computer Engineering and Applications*, AMERICAN-MATH'11/CEA'11, pages 123–128, Stevens Point, Wisconsin, USA, 2011. World Scientific and Engineering Academy and Society (WSEAS). ISBN 978-960-474-270-7. URL http://dl.acm.org/citation.cfm?id=1959666.1959689. 22

Youssef Bassil. Hybrid information retrieval model for web images. *CoRR*, abs/1204.0182, 2012. URL http://arxiv.org/abs/1204.0182. 5, 38

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 11(3):346–359, Jun 2008. ISSN 1077-3142. 19, 20, 25, 28, 46, 61, 87, 118, 140, 163

Richard Bellman. Adaptive control processes: a guided tour. 1961. 6

S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, Apr 2002. ISSN 0162-8828. doi: 10.1109/34.993558. 9, 20, 30, 87, 118, 140

Anna Bosch, Andrew Zisserman, and Xavier Muñoz. *Scene Classification Via pLSA*, pages 517–530. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-33839-0. doi: 10.1007/11744085_40. URL http://dx.doi.org/10.1007/11744085_40. 38

T. Botterill, S. Mills, and R. Green. Speeded-up bag-of-words algorithm for robot localisation through scene recognition. In *2008 23rd International Conference Image and Vision Computing New Zealand*, pages 1–6, Nov 2008. doi: 10.1109/IVCNZ.2008.4762067. 46

Nouha Bouteldja, Valerie Gouet-Brunet, and Michel Scholl. *The Many Facets of Progressive Retrieval for CBIR*, pages 611–624. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-89796-5. doi: 10.1007/978-3-540-89796-5_63. URL http://dx.doi.org/10.1007/978-3-540-89796-5_63. 39

R. Brunelli and O. Mich. Histograms analysis for image retrieval. *Pattern Recognition*, 34(8):1625 – 1637, 2001. ISSN 0031-3203. doi: https://doi.org/10.1016/

S0031-3203(00)00054-6. URL http://www.sciencedirect.com/science/article/pii/S0031320300000546. 53

Grigore C. Burdea. *Force and Touch Feedback for Virtual Reality*. John Wiley & Sons, Inc., New York, NY, USA, 1996. ISBN 0-471-02141-5. 156

Grigore C. Burdea. Haptic feedback for virtual reality, 1999. 156

Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010. 31

Yudong Cao, Honggang Zhang, Yanyan Gao, Xiaojun Xu, and Jun Guo. Matching Image with Multiple Local Features. In *Proceedings of 20th International Conference on Pattern Recognition*, pages 519–522, 2010. doi: 10.1109/ICPR.2010.132. 39, 58, 64

C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, Aug 2002. ISSN 0162-8828. doi: 10.1109/TPAMI.2002.1023800. 18

S. Chaabouni, F. Tison, J. Benois-Pineau, and C. Ben Amar. Prediction of visual attention with deep cnn for studies of neurodegenerative diseases. In *2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 1–6, June 2016. doi: 10.1109/CBMI.2016.7500243. 33

Vijay Chandrasekhar, Jie Lin, Olivier Morère, Hanlin Goh, and Antoine Veillard. A practical guide to cnns and fisher vectors for image instance retrieval. *CoRR*, abs/1508.02496, 2015. URL http://arxiv.org/abs/1508.02496. 49

Shih-Fu Chang, T. Sikora, and A. Purl. Overview of the mpeg-7 standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):688–695, Jun 2001. ISSN 1051-8215. doi: 10.1109/76.927421. 8, 22

Savvas A. Chatzichristofis and Yiannis S. Boutalis. *CEDD: Color and Edge Directivity Descriptor: A Compact Descriptor for Image Indexing and Retrieval*, pages 312–322. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008a. ISBN 978-3-540-79547-6. doi: 10.1007/978-3-540-79547-6_30. URL http://dx.doi.org/10.1007/978-3-540-79547-6_30. 52

Savvas A. Chatzichristofis and Yiannis S. Boutalis. Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval. In *Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services*, WIAMIS '08, pages 191–196, Washington, DC, USA, 2008b. IEEE Computer Society. ISBN 978-0-7695-3130-4. doi: 10.1109/WIAMIS.2008.24. URL http://dx.doi.org/10.1109/WIAMIS.2008.24. 52

Savvas A. Chatzichristofis and Yiannis S. Boutalis. Content based radiology image retrieval using a fuzzy rule based scalable composite descriptor. *Multimedia Tools and Applications*, 46(2):493–519, 2010. ISSN 1573-7721. doi: 10.1007/s11042-009-0349-x. URL http://dx.doi.org/10.1007/s11042-009-0349-x. 52

Savvas A. Chatzichristofis, Yiannis S. Boutalis, and Mathias Lux. Spcd - spatial color distribution descriptor - a fuzzy rule based compact composite descriptor appropriate for hand drawn color sketches retrieval. In *Proceedings of the 2nd International Conference on Agents and Artificial Intelligence - Volume 1: ICAART,*, pages 58–63, 2010a. ISBN 978-989-674-021-4. doi: 10.5220/0002725800580063. 52

Savvas A Chatzichristofis, YS Boutalis, and A Arampatzis. Investigating the behavior of compact composite descriptors in early fusion, late fusion and distributed image retrieval. *Radioengineering*, 2010b. xvi, 9, 51, 52, 63

D. M. Chen, G. Baatz, K. Köser, S. S. Tsai, R. Vedantham, T. Pylvänäinen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk. City-scale landmark identification on mobile devices. In *CVPR*, pages 737–744, June 2011. doi: 10.1109/CVPR.2011.5995610. 163

Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan. Deep domain adaptation for describing people based on fine-grained clothing attributes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5315–5324, June 2015. doi: 10.1109/CVPR.2015.7299169. 137

Y. H. Chen, T. Krishna, J. S. Emer, and V. Sze. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE Journal of Solid-State Circuits*, 52 (1):127–138, Jan 2017. ISSN 0018-9200. doi: 10.1109/JSSC.2016.2616357. 34

Y. T. Chen and C. S. Chen. Fast human detection using a novel boosted cascading structure with meta stages. *IEEE Transactions on Image Processing*, 17(8):1452–1464, Aug 2008. ISSN 1057-7149. doi: 10.1109/TIP.2008.926152. 50, 62

M. M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang, and S. M. Hu. Global contrast based salient region detection. In *CVPR 2011*, pages 409–416, June 2011. doi: 10.1109/CVPR.2011.5995344. 48

A. Cho, W. K. Yang, D. S. Jeong, and W. G. Oh. Bag-of-features signature using invariant region descriptor for object retrieval. In *2011 17th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV)*, pages 1–4, Feb 2011. doi: 10.1109/FCV.2011.5739754. xvi, 46, 62

Ayoung Cho, Won-Keun Yang, Weon-Geun Oh, and Dong-Seok Jeong. Concentric circle-based image signature for near-duplicate detection in large databases. *ETRI journal*, 32(6): 871–880, 2010. 46

Sumit Chopra, Suhrid Balakrishnan, and Raghuraman Gopalan. Dlid: Deep learning for domain adaptation by interpolating between domains. In *in ICML Workshop on Challenges in Representation Learning*, 2013. 137

R. Choudhary, N. Raina, N. Chaudhary, R. Chauhan, and R. H. Goudar. An integrated approach to content based image retrieval. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 2404–2410, Sept 2014. doi: 10.1109/ICACCI.2014.6968394. 9, 41, 62

Tommy W.S. Chow and M.K.M. Rahman. A new image classification technique using tree-structured regional features. *Neurocomputing*, 70(4–6):1040 – 1050, 2007. ISSN 0925-2312. doi: http://dx.doi.org/10.1016/j.neucom.2006.01.033. URL //www.sciencedirect.com/science/article/pii/S092523120600244X. Advanced Neurocomputing Theory and MethodologySelected papers from the International Conference on Intelligent Computing 2005 (ICIC 2005)International Conference on Intelligent Computing 2005. xvi, 47, 48, 62

Wei-Ta Chu and Ming-Hung Tsai. Visual pattern discovery for architecture image classification and product image search. In *Proceedings of the 2Nd ACM International Conference on Multimedia Retrieval*, ICMR '12, pages 27:1–27:8, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1329-2. doi: 10.1145/2324796.2324831. URL http://doi.acm.org/10.1145/2324796.2324831. 138

J.L. Crowley and Alice C. Parker. A representation for shape based on peaks and ridges in the difference of low-pass transform. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(2):156–170, 1984. ISSN 0162-8828. doi: 10.1109/TPAMI.1984.4767500. 25

M. Crucianu, J. Tarel, and M. Ferecatu. An Exploration of Diversified User Strategies for Image Retrieval with Relevance Feedback. *Visual Languages and Computing*, 19:629–636, 2008. doi: 10.1016/j.jvlc.2008.04.006. 5

Carolina Cruz-Neira, Daniel J. Sandin, and Thomas A. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the cave. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '93, pages 135–142, New York, NY, USA, 1993. ACM. ISBN 0-89791-601-8. doi: 10.1145/166117.166134. URL http://doi.acm.org/10.1145/166117.166134. 156

Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. 35

R. da S. Torres, A.X. Falcão, and L. da F. Costa. A graph-based approach for multiscale shape analysis. *Pattern Recognition*, 37(6):1163 – 1174, 2004. ISSN 0031-3203. doi:

http://doi.org/10.1016/j.patcog.2003.10.007. URL http://www.sciencedirect.com/science/article/pii/S0031320303004011. 44

Ricardo da S. Torres, Alexandre X. Falcão, Marcos A. Gonçalves, João P. Papa, Baoping Zhang, Weiguo Fan, and Edward A. Fox. A genetic programming framework for content-based image retrieval. *Pattern Recognition*, 42(2):283 – 292, 2009. ISSN 0031-3203. doi: http://dx.doi.org/10.1016/j.patcog.2008.04.010. URL http://www.sciencedirect.com/science/article/pii/S0031320308001623. Learning Semantics from Multimedia Content. xv, 18, 44, 62

Ricardo da Silva Torres and Alexandre X. Falcão. Content-based image retrieval: Theory and applications. *RITA*, 13(2):161–185, 2006. 8

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, 2005. doi: 10.1109/CVPR.2005.177. 9, 19, 29, 41

R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: ideas, influences, and trends of the new age. *ACM COMPUTING SURVEYS*, 40(2):55:1–5:60, May 2008. 6, 20, 66

John G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20(10):847–856, 1980. ISSN 0042-6989. doi: http://dx.doi.org/10.1016/0042-6989(80)90065-6. URL http://www.sciencedirect.com/science/article/pii/0042698980900656. 21, 28

Philippe Pérez de San Roman, Jenny Benois-Pineau, Jean-Philippe Domenger, Aymar de Rugy, Florent Paclet, and Daniel Cataert. Saliency driven object recognition in egocentric videos with deep cnn: toward application in assistance to neuroprostheses. *Computer Vision and Image Understanding*, pages –, 2017. ISSN 1077-3142. doi: https://doi.org/10.1016/j.cviu.2017.03.001. URL http://www.sciencedirect.com/science/article/pii/S1077314217300462. 33

Yining Deng, B. S. Manjunath, and H. Shin. Color image segmentation. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 2, page 451 Vol. 2, 1999. doi: 10.1109/CVPR.1999.784719. 47

Yining Deng, B. S. Manjunath, C. Kenney, M. S. Moore, and H. Shin. An efficient color representation for image retrieval. *IEEE Transactions on Image Processing*, 10(1):140–147, Jan 2001. ISSN 1057-7149. doi: 10.1109/83.892450. 22

Adrien Depeursinge and Henning Müller. *ImageCLEF: Experimental Evaluation in Visual Information Retrieval*, chapter Fusion Techniques for Combining Textual and Visual Information Retrieval, pages 95–114. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-15181-1. doi: 10.1007/978-3-642-15181-1_6. URL http://dx.doi.org/10.1007/978-3-642-15181-1_6. 51

Timo Dickscheid and Wolfgang Förstner. Evaluating the suitability of feature detectors for automatic image orientation systems. In *Computer Vision Systems, 7th International Conference on Computer Vision Systems, ICVS 2009, Liège, Belgium, October 13-15, 2009, Proceedings*, pages 305–314, 2009. doi: 10.1007/978-3-642-04667-4_31. URL http://dx.doi.org/10.1007/978-3-642-04667-4_31. 105

Timo Dickscheid, Falko Schindler, and Wolfgang Förstner. Coding images with local features. *Int. J. Comput. Vision*, 94(2):154–174, September 2011. ISSN 0920-5691. doi: 10.1007/s11263-010-0340-z. URL http://dx.doi.org/10.1007/s11263-010-0340-z. 105

B Dinakaran, J Annapurna, and Ch Aswani Kumar. Interactive image retrieval using text and image content. *Cybern Inf Tech*, 10:20–30, 2010. 39

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *CoRR*, abs/1310.1531, 2013. URL http://arxiv.org/abs/1310.1531. 18, 21, 32, 33

Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pages 19:1–19:8, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-480-5. doi: 10.1145/1646396.1646421. URL http://doi.acm.org/10.1145/1646396.1646421. 35

Rajshree S Dubey, Rajnish Choubey, and Joy Bhattacharjee. Multi feature content based image retrieval. *International Journal on Computer Science and Engineering*, 2(6):2145–2149, 2010. 39, 54, 63

S. Ehsan, N. Kanwal, A.F. Clark, and K.D. McDonald-Maier. Improved repeatability measures for evaluating performance of feature detectors. *Electronics Letters*, 46(14):998–1000, July 2010. ISSN 0013-5194. doi: 10.1049/el.2010.1442. 104

Shoaib Ehsan, Adrian F. Clark, and Klaus D. McDonald-Maier. Rapid online analysis of local feature detectors and their complementarity. *Sensors*, 13(8):10876, 2013. ISSN 1424-8220. doi: 10.3390/s130810876. URL http://www.mdpi.com/1424-8220/13/8/10876. 12, 107

HA Elnemr. Combining surf and mser along with color features for image retrieval system based on bag of visual words. *Journal of Computer Science*, 12(4):213–222, 2016. 61, 64

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 33

Weiguo Fan, Edward A. Fox, Praveen Pathak, and Harris Wu. The effects of fitness functions on genetic programming-based ranking discovery for web search. *Journal of the American*

Society for Information Science and Technology, 55(7):628–636, 2004. ISSN 1532-2890. doi: 10.1002/asi.20009. URL http://dx.doi.org/10.1002/asi.20009. 44

L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 524–531 vol. 2, June 2005. doi: 10.1109/CVPR.2005.16. 48

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding*, 106(1):59–70, 2007. 32

Marin Ferecatu, Nozha Boujemaa, and Michel Crucianu. Semantic interactive image retrieval combining visual and conceptual content description. *Multimedia Systems*, 13(5):309–322, 2008. ISSN 1432-1882. doi: 10.1007/s00530-007-0094-9. URL http://dx.doi.org/10.1007/s00530-007-0094-9. 38

C.D. Ferreira, J.A. Santos, R. da S. Torres, M.A. Gonçalves, R.C. Rezende, and Weiguo Fan. Relevance feedback based on genetic programming for image retrieval. *Pattern Recognition Letters*, 32(1):27 – 37, 2011. ISSN 0167-8655. doi: http://dx.doi.org/10.1016/j.patrec.2010.05.015. URL http://www.sciencedirect.com/science/article/pii/S0167865510001558. Image Processing, Computer Vision and Pattern Recognition in Latin America. 19

D. Filliat. A visual bag of words method for interactive qualitative localization and mapping. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3921–3926, April 2007. doi: 10.1109/ROBOT.2007.364080. 46

Philipp Fischer, Alexey Dosovitskiy, and Thomas Brox. Descriptor matching with convolutional neural networks: a comparison to SIFT. *CoRR*, abs/1405.5769, 2014. URL http://arxiv.org/abs/1405.5769. 34

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24 (6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL http://doi.acm.org/10.1145/358669.358692. 163

Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, and Marc Pollefeys. *Building Rome on a Cloudless Day*, pages 368–381. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-15561-1. doi: 10.1007/978-3-642-15561-1_27. URL http://dx.doi.org/10.1007/978-3-642-15561-1_27. 162, 167

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139,

1997. ISSN 0022-0000. doi: http://dx.doi.org/10.1006/jcss.1997.1504. URL http://www.sciencedirect.com/science/article/pii/S002200009791504X. 138

Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Academic press, 2013. 38

Guillaume Gales, Alain Crouzil, and Sylvie Chambon. Complementarity of feature point detectors. In Paul Richard and José Braz, editors, *VISAPP (1)*, pages 334–339. INSTICC Press, 2010. ISBN 978-989-674-028-3. URL http://dblp.uni-trier.de/db/conf/visapp/visapp2010-1.html#GalesCC10. 12, 104, 109

Lianli Gao, Jingkuan Song, Fuhao Zou, Dongxiang Zhang, and Jie Shao. Scalable multimedia retrieval by deep learning hashing with relative similarity learning. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, pages 903–906, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3459-4. doi: 10.1145/2733373.2806360. URL http://doi.acm.org/10.1145/2733373.2806360. 34

Michael Gashler, Dan Ventura, and Tony Martinez. Iterative non-linear dimensionality reduction with manifold sculpting. In J.c. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 513–520. MIT Press, Cambridge, MA, 2007. URL http://books.nips.cc/papers/files/nips20/NIPS2007_0690.pdf. 18

Steffen Gauglitz, Tobias Höllerer, and Matthew Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision*, 94(3):335, 2011. ISSN 1573-1405. doi: 10.1007/s11263-011-0431-5. URL http://dx.doi.org/10.1007/s11263-011-0431-5. 8

P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *2009 IEEE 12th International Conference on Computer Vision*, pages 221–228, Sept 2009. doi: 10.1109/ICCV.2009.5459169. 58, 59

T. Gevers and H. Stokman. Robust histogram construction from color invariants for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):113–118, Jan 2004. ISSN 0162-8828. doi: 10.1109/TPAMI.2004.1261083. 8

Th Gevers, Arnold WM Smeulders, et al. Content-based image retrieval: An overview. 2004. 8

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, June 2014. doi: 10.1109/CVPR.2014.81. 32, 33

J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 487–493 vol.1, Oct 2003. doi: 10.1109/ICCV.2003.1238387. 50

Simon Goodall, Paul H. Lewis, Kirk Martinez, Patrick A. S. Sinclair, Fabrizio Giorgini, Matthew J. Addis, Mike J. Boniface, Christian Lahanier, and James Stevenson. *SCULPTEUR: Multimedia Retrieval for Museums*, pages 638–646. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-27814-6. doi: 10.1007/978-3-540-27814-6_74. URL http://dx.doi.org/10.1007/978-3-540-27814-6_74. 138

R. Gopalan, Ruonan Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *2011 International Conference on Computer Vision*, pages 999–1006, Nov 2011. doi: 10.1109/ICCV.2011.6126344. 137

R.M. Gray and D.L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6): 2325–2383, Oct 1998. ISSN 0018-9448. doi: 10.1109/18.720541. 68

V. N. Gudivada and V. V. Raghavan. Content based image retrieval systems. *Computer*, 28(9): 18–22, Sep 1995. ISSN 0018-9162. doi: 10.1109/2.410145. 8

A. Haja, B. Jahne, and S. Abraham. Localization accuracy of region detectors. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008. doi: 10.1109/CVPR.2008.4587829. 104, 105

Ju Han and Kai-Kuang Ma. Fuzzy color histogram and its use in color image retrieval. *IEEE Transactions on Image Processing*, 11(8):944–952, Aug 2002. ISSN 1057-7149. doi: 10.1109/TIP.2002.801585. 54

C. Harris and M. Stephens. A combined corner and edge detector. In *In Proc. of Fourth Alvey Vision Conference*, pages 147–151, 1988. 23

Zellig S Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954. 35

G. Heidemann. Focus-of-attention from local color symmetries. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(7):817–830, July 2004. ISSN 0162-8828. doi: 10.1109/TPAMI.2004.29. 25, 117, 140

Mahmoud R Hejazi and Yo-Sung Ho. An efficient approach to texture-based image retrieval. *International journal of imaging systems and technology*, 17(5):295–302, 2007. 54

Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Comput.*, 18(7):1527–1554, July 2006. ISSN 0899-7667. doi: 10.1162/neco.2006.18.7.1527. URL http://dx.doi.org/10.1162/neco.2006.18.7.1527. 32

P.S. Hiremath and J. Pujari. Content based image retrieval using color, texture and shape features. In *Advanced Computing and Communications, 2007. ADCOM 2007. International Conference on*, pages 780–784, 2007. 9

Judy Hoffman, Eric Tzeng, Jeff Donahue, Yangqing Jia, Kate Saenko, and Trevor Darrell. One-shot adaptation of supervised deep convolutional models. *CoRR*, abs/1312.6204, 2013. URL http://arxiv.org/abs/1312.6204. 137

W. Hu, N. Xie, L. Li, X. Zeng, and S. Maybank. A survey on visual content-based video indexing and retrieval. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6):797–819, Nov 2011. ISSN 1094-6977. doi: 10.1109/TSMCC.2011.2109710. 6, 8

Junshi Huang, Rogério Feris, Qiang Chen, and Shuicheng Yan. Cross-domain image retrieval with a dual attribute-aware ranking. In *ICCV*, pages 1062–1070, 2015a. 137

Min Huang, Huazhong Shu, Yaqiong Ma, and Qiuping Gong. Content-based image retrieval technology using multi-feature fusion. *Optik - International Journal for Light and Electron Optics*, 126(19):2144 – 2148, 2015b. ISSN 0030-4026. doi: https://doi.org/10.1016/j.ijleo.2015.05.095. URL http://www.sciencedirect.com/science/article/pii/S0030402615004088. 40, 54, 63

Z. C. Huang, P. P. K. Chan, W. W. Y. Ng, and D. S. Yeung. Content-based image retrieval using color moment and gabor texture feature. In *2010 International Conference on Machine Learning and Cybernetics*, volume 2, pages 719–724, July 2010. doi: 10.1109/ICMLC.2010.5580566. xvi, 18, 53, 54, 63

A. Irschara, C. Zach, J. M. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. In *CVPR*, pages 2599–2606, June 2009. doi: 10.1109/CVPR.2009.5206587. 162, 163

H. Jegou, M. Douze, and C. Schmid. Product Quantization for Nearest Neighbor Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011. ISSN 0162-8828. doi: 10.1109/TPAMI.2010.57. 68

Zhong Ji, Jing Wang, Yuting Su, Zhanjie Song, and Shikai Xing. Balance between object and background: Object-enhanced features for scene image classification. *Neurocomputing*, 120:15 – 23, 2013. ISSN 0925-2312. doi: http://dx.doi.org/10.1016/j.neucom.2012.02.054. URL //www.sciencedirect.com/science/article/pii/S0925231213002804. Image Feature Detection and Description. xvi, 19, 39, 47, 48, 62

Shu-Qiang Jiang, Jun Du, Qing-Ming Huang, Tie-Jun Huang, and Wen Gao. Visual ontology construction for digitized art image retrieval. *Journal of Computer Science and Technology*, 20(6):855–860, 2005. ISSN 1860-4749. doi: 10.1007/s11390-005-0855-x. URL http://dx.doi.org/10.1007/s11390-005-0855-x. 138

Su Jie and Wang Bingqin1 Guo Li. Textural feature extraction and classification study research of digital image. *Electronic Measurement Technology*, 5:016, 2008. 54

Feng Jing, Mingjing Li, Hong-Jiang Zhang, and Bo Zhang. An efficient and effective region-based image retrieval framework. *IEEE Transactions on Image Processing*, 13(5):699–709, May 2004. ISSN 1057-7149. doi: 10.1109/TIP.2004.826125. 5

F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 604–610 Vol. 1, Oct 2005. doi: 10.1109/ICCV.2005.66. 35, 37

H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3304–3311, June 2010. doi: 10.1109/CVPR.2010.5540039. 6

S. Kashioka, M. Ejiri, and Y. Sakamoto. A transistor wire-bonding system utilizing multiple local pattern matching techniques. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(8):562–570, Aug 1976. ISSN 0018-9472. doi: 10.1109/TSMC.1976.4309551. 5

Yan Ke and R. Sukthankar. Pca-sift: a more distinctive representation for local image descriptors. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–506–II–513 Vol.2, June 2004. doi: 10.1109/CVPR.2004.1315206. 61

Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, pages 2938–2946, 2015. ISBN 978-1-4673-8391-2. doi: 10.1109/ICCV.2015.336. 163

John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992. ISBN 0-262-11170-5. 44

S. Krishnamachari and R. Chellappa. Multiresolution gauss-markov random field models for texture segmentation. *IEEE Transactions on Image Processing*, 6(2):251–267, Feb 1997. ISSN 1057-7149. doi: 10.1109/83.551696. 8

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. xv, 32

Jérome Landré, Frédéric Truchetet, Sophie Montuire, and Bruno David. Automatic building of a visual interface for content-based multiresolution retrieval of paleontology images. *Journal of Electronic Imaging*, 10(4):957–965, 2001. doi: 10.1117/1.1406505. URL http://dx.doi.org/10.1117/1.1406505. 64

S. Lazebnik and M. Raginsky. Supervised learning of quantizer codebooks by information loss minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(7): 1294–1309, July 2009. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.138. 37

S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using affine-invariant regions. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–319–II–324 vol.2, June 2003. doi: 10.1109/CVPR.2003.1211486. 95

S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1265–1278, 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.151. 28

S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178, 2006. doi: 10.1109/CVPR.2006.68. 59

S. Leutenegger, M. Chli, and R.Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2548–2555, Nov 2011. doi: 10.1109/ICCV.2011.6126542. 19, 25, 28, 31, 102, 117, 148

Michael S. Lew, Nicu Sebe, Chabane Djeraba, and Ramesh Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19, February 2006. ISSN 1551-6857. doi: 10.1145/1126004.1126005. URL http://doi.acm.org/10.1145/1126004.1126005. 20

Jing Li and Nigel M. Allinson. A comprehensive review of current local features for computer vision. *Neurocomputing*, 71(10–12):1771 – 1787, 2008. ISSN 0925-2312. doi: http://dx.doi.org/10.1016/j.neucom.2007.11.032. URL http://www.sciencedirect.com/science/article/pii/S0925231208001124. Neurocomputing for Vision ResearchAdvances in Blind Signal Processing. 8

P. Li, H. Yan, G. Cui, and Y. Du. Image local invariant features matching using global information. In *2012 IEEE International Conference on Information Science and Technology*, pages 627–633, March 2012a. doi: 10.1109/ICIST.2012.6221721. 60, 64

Xinchao Li, M. Larson, and A. Hanjalic. Pairwise geometric matching for large-scale object retrieval. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5153–5161, June 2015. doi: 10.1109/CVPR.2015.7299151. 127, 133, 172

Xiuqi Li, Shu-Ching Chen, Mei-Ling Shyu, and Borko Furht. An efficient multi-filter retrieval framework for large image databases. *SIMULATION SERIES*, 34(2):81–86, 2002. 61, 64

Yunpeng Li, Noah Snavely, Dan Huttenlocher, and Pascal Fua. Worldwide pose estimation using 3D point clouds. In *ECCV*, pages 15–29, 2012b. ISBN 978-3-642-33717-8. doi: 10.1007/978-3-642-33718-5_2. 162, 163

H. Lim, S. N. Sinha, M. F. Cohen, and M. Uyttendaele. Real-time image-based 6-DOF localization in large-scale environments. In *CVPR*, pages 1043–1050, June 2012. doi: 10.1109/CVPR.2012.6247782. 163

D. A. Lisin, M. A. Mattar, M. B. Blaschko, E. G. Learned-Miller, and M. C. Benfield. Combining local and global image features for object class recognition. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, pages 47–47, June 2005. doi: 10.1109/CVPR.2005.433. 60, 64

Haihui Liu, Wan-Lei Zhao, Hanzi Wang, Kyungmo Koo, and Sangwhan Moon. A comparative study on features for similar image search. In *Proceedings of the International Conference on Internet Multimedia Computing and Service*, ICIMCS'16, pages 349–353, New York, NY, USA, 2016a. ACM. ISBN 978-1-4503-4850-8. doi: 10.1145/3007669.3008269. URL http://doi.acm.org/10.1145/3007669.3008269. 34

Ningning Liu, Emmanuel Dellandréa, Bruno Tellez, and Liming Chen. *A Selective Weighted Late Fusion for Visual Concept Recognition*, pages 1–28. Springer International Publishing, Cham, 2014. ISBN 978-3-319-05696-8. doi: 10.1007/978-3-319-05696-8_1. URL http://dx.doi.org/10.1007/978-3-319-05696-8_1. xvi, 57, 63

Wei Liu, Weidong Xu, Lihua Li, and Weiwei Wang. Applying visual attention computational model and latent semantic indexing to image retrieval. In *2009 4th IEEE Conference on Industrial Electronics and Applications*, pages 2667–2671, May 2009. doi: 10.1109/ICIEA.2009.5138691. 18

Yu Liu, Yanming Guo, Song Wu, and Michael S. Lew. Deepindex for accurate and efficient image retrieval. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ICMR '15, pages 43–50, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3274-3. doi: 10.1145/2671188.2749300. URL http://doi.acm.org/10.1145/2671188.2749300. 33

Z. Liu, L. Y. Duan, J. Chen, and T. Huang. Depth-based local feature selection for mobile visual search. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 276–280, Sept 2016b. doi: 10.1109/ICIP.2016.7532362. 163

DavidG. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. ISSN 0920-5691. doi: 10.1023/B:VISI.0000029664.99615.94. URL http://dx.doi.org/10.1023/B%3AVISI.0000029664.99615.94. 9, 18, 21, 23, 25, 28, 34, 41, 48, 60, 61, 87, 118, 140, 163

Bo Luo, Xiaogang Wang, and Xiaoou Tang. World wide web based image search engine using text and image content features. In *Electronic Imaging 2003*, pages 123–130. International Society for Optics and Photonics, 2003. 39

Z. M. Ma, Gang Zhang, and Li Yan. Shape feature descriptor using modified zernike moments. *Pattern Analysis and Applications*, 14(1):9–22, 2011. ISSN 1433-755X. doi: 10.1007/s10044-009-0171-0. URL http://dx.doi.org/10.1007/s10044-009-0171-0. 54

J. Maatta, A. Hadid, and M. Pietikainen. Face spoofing detection from single images using texture and local shape analysis. *IET Biometrics*, 1(1):3–10, March 2012. ISSN 2047-4938. doi: 10.1049/iet-bmt.2011.0009. 58, 63

Telu Venkata Madhusudhanarao, Sanaboina Pallam Setty, and Yarramalle Srinivas. Model based approach for content based image retrievals based on fusion and relevancy methodology. *International Arab Journal of Information Technology (IAJIT)*, 12(6), 2015. 39, 50, 62

Fatin Abbas Mahdi and Abdulkareem Ibadi. Mirs: Museum image retrieval system using most appropriate low-level feature descriptors. *International Journal of Computer Science Issues (IJCSI)*, 11(5):1, 2014. 138

J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of the British Machine Vision Conference*, pages 36.1–36.10, 2002. 21, 23, 24, 58, 102, 117, 140

Geoffrey McLachlan. *Discriminant analysis and statistical pattern recognition*, volume 544. John Wiley & Sons, 2004. 6

K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630, 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.188. 21, 28

K. Mikolajczyk, B. Leibe, and B. Schiele. Local features for object class recognition. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1792–1799 Vol. 2, Oct 2005. doi: 10.1109/ICCV.2005.146. 12, 104, 110

Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *Int. J. Comput. Vision*, 60(1):63–86, October 2004. ISSN 0920-5691. doi: 10.1023/B:VISI.0000027790.02288.f2. URL http://dx.doi.org/10.1023/B:VISI.0000027790.02288.f2. 21, 23, 24, 87, 102, 104, 117, 140

M. J. Milford and G. F. Wyeth. Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *2012 IEEE International Conference on Robotics and Automation*, pages 1643–1649, May 2012. doi: 10.1109/ICRA.2012.6224623. 162

P. Montesinos, V. Gouet, and R. Deriche. Differential Invariants for Color Images. In *Proceedings of* 14^{th} *International Conference on Pattern Recognition*, pages 838–840, Brisbane, Australia, 1998. 23

Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*, volume 821. John Wiley & Sons, 2012. 112

Frank Moosmann, Eric Nowak, and Frederic Jurie. Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1632–1646, 2008. 37

H.P. Moravec. Towards automatic visual obstacle avoidance. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, page 584, August 1977. 23

Pierre Moreels and Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects. *Int. J. Comput. Vision*, 73(3):263–284, July 2007. ISSN 0920-5691. doi: 10.1007/s11263-006-9967-1. URL http://dx.doi.org/10.1007/s11263-006-9967-1. 104

J. M. Morel and G. Yu. ASIFT: A New Framework for Fully Affine Invariant Image Comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009. 18, 29, 58

Dibyendu Mukherjee, QM Jonathan Wu, and Guanghui Wang. A comparative experimental study of image feature detectors and descriptors. *Machine Vision and Applications*, 26(4): 443–466, 2015. ISSN 1432-1769. doi: 10.1007/s00138-015-0679-9. URL http://dx.doi.org/10.1007/s00138-015-0679-9. 8, 20

Hiroshi Murase and Shree K. Nayar. Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14(1):5–24, 1995. ISSN 1573-1405. doi: 10.1007/BF01421486. URL http://dx.doi.org/10.1007/BF01421486. 22

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 of Medical Informatics*, 73(1):1 – 23, 2004. ISSN 1386-5056. doi: http://dx.doi.org/10.1016/j.ijmedinf.2003.11.024. URL http://www.sciencedirect.com/science/article/pii/S1386505603002119. 6, 8

K. Nallaperumal, M. S. Banu, and C. C. Christiyana. Content based image indexing and retrieval using color descriptor in wavelet domain. In *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007)*, volume 3, pages 185–189, Dec 2007. doi: 10.1109/ICCIMA.2007.72. 22

V. E. Neagoe and A. D. Ropot. Concurrent self-organizing maps for pattern classification. In *Proceedings First IEEE International Conference on Cognitive Informatics*, pages 304–312, 2002. doi: 10.1109/COGINF.2002.1039311. 47

NikolayN. Neshov. Comparison on Late Fusion Methods of Low Level Features for Content Based Image Retrieval. In Valeri Mladenov, Petia Koprinkova-Hristova, Günther

Palm, Alessandro E.P. Villa, Bruno Appollini, and Nikola Kasabov, editors, *Artificial Neural Networks and Machine Learning*, volume 8131 of *Lecture Notes in Computer Science*, pages 619–627. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-40727-7. doi: 10.1007/978-3-642-40728-4_77. xxii, 39, 40, 51, 63, 89, 90, 91, 100, 122, 123, 141, 142

J. Y. H. Ng, F. Yang, and L. S. Davis. Exploiting local features from deep networks for image retrieval. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 53–61, June 2015. doi: 10.1109/CVPRW.2015.7301272. 34

Q. D. Nguyen, A. Devaux, M. Brédif, and N. Paparoditis. A 3d heterogeneous interactive web mapping application. In IEEE *Virtual Reality Conference*, pages 838–840, Arles, France, March 2015. 162

David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2161–2168, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2597-0. doi: 10.1109/CVPR.2006.264. URL http://dx.doi.org/10.1109/CVPR.2006.264. 35, 37, 46, 163

Abraham Montoya Obeso, Jenny Benois-Pineau, Alejandro Alvaro Ramírez-Acosta, and Mireya S. García-Vázquez. Architectural style classification of mexican historical buildings using deep convolutional neural networks and sparse features. *J. Electronic Imaging*, 26 (1):11016, 2017. doi: 10.1117/1.JEI.26.1.011016. URL https://doi.org/10.1117/1.JEI.26.1.011016. 138

Stephen O'Hara and Bruce A. Draper. Introduction to the bag of features paradigm for image classification and retrieval. *CoRR*, abs/1101.3354, 2011. URL http://arxiv.org/abs/1101.3354. 35

T. Ojala, M. Pietikäinen, and T. Maenpää. Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. 41, 42, 60

Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51 – 59, 1996. ISSN 0031-3203. doi: http://dx.doi.org/10.1016/0031-3203(95)00067-4. URL http://www.sciencedirect.com/science/article/pii/0031320395000674. 9, 30, 57

Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001. ISSN 1573-1405. doi: 10.1023/A:1011139631724. URL http://dx.doi.org/10.1023/A:1011139631724. 22

Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the 2014*

IEEE Conference on Computer Vision and Pattern Recognition, CVPR '14, pages 1717–1724, Washington, DC, USA, 2014. IEEE Computer Society. ISBN 978-1-4799-5118-5. doi: 10.1109/CVPR.2014.222. URL http://dx.doi.org/10.1109/CVPR.2014.222. 8, 33

N. Paparoditis, J.-P. Papelard, B. Cannelle, A. Devaux, B. Soheilian, N. David, and E. Houzay. Stereopolis II: A multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology. *Revue Française de Photogrammétrie et de Télédétection*, 200: 69–79, October 2012. 162, 164

G. Paschos, I. Radev, and N. Prabakar. Image content-based retrieval using chromaticity moments. *IEEE Transactions on Knowledge and Data Engineering*, 15(5):1069–1072, Sept 2003. ISSN 1041-4347. doi: 10.1109/TKDE.2003.1232264. 22

Otávio A.B. Penatti, Eduardo Valle, and Ricardo da S. Torres. Comparative study of global color and texture descriptors for web image retrieval. *Journal of Visual Communication and Image Representation*, 23(2):359 – 380, 2012. ISSN 1047-3203. doi: https://doi.org/10.1016/j.jvcir.2011.11.002. URL http://www.sciencedirect.com/science/article/pii/S1047320311001465. 22

Florent Perronnin, Christopher Dance, Gabriela Csurka, and Marco Bressan. Adapted vocabularies for generic visual categorization. In *European Conference on Computer Vision*, pages 464–475. Springer, 2006. 37

J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007. doi: 10.1109/CVPR.2007.383172. 35

D. Picard, N. Thome, and M. Cord. An efficient system for combining complementary kernels in complex visual categorization tasks. In *2010 IEEE International Conference on Image Processing*, pages 3877–3880, Sept 2010. doi: 10.1109/ICIP.2010.5651051. xvi, 59, 64

C. Poglitsch, C. Arth, D. Schmalstieg, and J. Ventura. [poster] a particle filter approach to outdoor localization using image-based rendering. In *2015 IEEE International Symposium on Mixed and Augmented Reality*, pages 132–135, Sept 2015. doi: 10.1109/ISMAR.2015.39. 162

Mary C. Potter, Brad Wyble, Carl Erick Hagmann, and Emily S. McCourt. Detecting meaning in rsvp at 13 ms per picture. *Attention, Perception, & Psychophysics*, 76(2):270–279, 2014. ISSN 1943-393X. doi: 10.3758/s13414-013-0605-z. URL http://dx.doi.org/10.3758/s13414-013-0605-z. 2

Y. Rahulamathavan, R. C. W. Phan, J. A. Chambers, and D. J. Parish. Facial expression recognition in the encrypted domain based on local fisher discriminant analysis. *IEEE*

Transactions on Affective Computing, 4(1):83–92, Jan 2013. ISSN 1949-3045. doi: 10.1109/T-AFFC.2012.33. 18

N. Raina, N. Roshi, R. Chauhan, and R. H. Goudar. A fused features approach on content-based image retrieval based on fuzzy rule-set. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 2436–2442, Sept 2014. doi: 10.1109/ICACCI.2014.6968407. 58, 63

Alain Rakotomamonjy, Francis R Bach, Stéphane Canu, and Yves Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9(Nov):2491–2521, 2008. 60

Esmat Rashedi, Hossein Nezamabadi-pour, and Saeid Saryazdi. A simultaneous feature adaptation and feature selection method for content-based image retrieval systems. *Knowledge-Based Systems*, 39:85 – 94, 2013. ISSN 0950-7051. doi: http://dx.doi.org/10.1016/j.knosys.2012.10.011. URL http://www.sciencedirect.com/science/article/pii/S0950705112002924. 105

Srinivas S Ravela. *On multi-scale differential features and their representations for image retrieval and recognition*. PhD thesis, University of Massachusetts Amherst, 2003. 60

A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 512–519, June 2014. doi: 10.1109/CVPRW.2014.131. 20

Daniel Reisfeld, Haim Wolfson, and Yehezkel Yeshurun. Context-free attentional operators: The generalized symmetry transform. *International Journal of Computer Vision*, 14(2):119–130, 1995. ISSN 1573-1405. doi: 10.1007/BF01418978. URL http://dx.doi.org/10.1007/BF01418978. 25

Y. Ren, A. Bugeau, and J. Benois-Pineau. Bag-of-bags of words irregular graph pyramids vs spatial pyramid matching for image retrieval. In *2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6, Oct 2014. doi: 10.1109/IPTA.2014.7001967. 18

V. Risojevic and Z. Babic. Fusion of Global and Local Descriptors for Remote Sensing Image Classification. *IEEE Geoscience and Remote Sensing Letters*, 10(4):836–840, 2013. ISSN 1545-598X. doi: 10.1109/LGRS.2012.2225596. 58, 63

Roman Rosipal and Nicole Krämer. Overview and Recent Advances in Partial Least Squares. In Craig Saunders, Marko Grobelnik, Steve Gunn, and John Shawe-Taylor, editors, *Subspace, Latent Structure and Feature Selection*, volume 3940 of *Lecture Notes in Computer Science*, pages 34–51. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-34137-6. doi: 10.1007/11752790_2. 92

Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006. 25

E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564–2571, Nov 2011. doi: 10.1109/ICCV.2011.6126544. 18, 25, 31, 102, 118, 140

Stevan Rudinac, Jan Zahálka, and Marcel Worring. *Discovering Geographic Regions in the City Using Social Multimedia and Open Data*, pages 148–159. Springer International Publishing, Cham, 2017. ISBN 978-3-319-51814-5. doi: 10.1007/978-3-319-51814-5_13. URL http://dx.doi.org/10.1007/978-3-319-51814-5_13. 3

Yong Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):644–655, Sep 1998. ISSN 1051-8215. doi: 10.1109/76.718510. 8

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. ISSN 1047-3203. doi: http://dx.doi.org/10.1006/jvci.1999.0413. URL http://www.sciencedirect.com/science/article/pii/S1047320399904133. 6, 8

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y. 32

B. C. Russell, J. Sivic, J. Ponce, and H. Dessales. Automatic alignment of paintings and photographs depicting a 3d scene. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 545–552, Nov 2011. doi: 10.1109/ICCVW.2011.6130291. 137

Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986. ISBN 0070544840. 36

A. Salvador, D. Manchón-Vizuete, A. Calafell, X. Giró i Nieto, and M. Zeppelzauer. Cultural event recognition with visual convnets and temporal models. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 36–44, June 2015. doi: 10.1109/CVPRW.2015.7301334. 138

Koen Sande, Theo Gevers, and Cees Snoek. Evaluating Color Descriptors for Object and Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, September 2010. ISSN 0162-8828. 19, 57, 66

T. Sattler, B. Leibe, and L. Kobbelt. Fast image-based localization using direct 2D-to-3D matching. In *ICCV*, pages 667–674, Nov 2011. doi: 10.1109/ICCV.2011.6126302. 162, 163, 167

T. Sattler, M. Havlena, K. Schindler, and M. Pollefeys. Large-scale location recognition and the geometric burstiness problem. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1582–1590, June 2016. doi: 10.1109/CVPR.2016.175. 106

Torsten Sattler, Bastian Leibe, and Leif Kobbelt. *Improving Image-Based Localization by Active Correspondence Search*, pages 752–765. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-33718-5. doi: 10.1007/978-3-642-33718-5_54. URL http://dx.doi.org/10.1007/978-3-642-33718-5_54. 163

Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. In *CVPR*, pages 1–7, June 2007. doi: 10.1109/CVPR.2007.383150. 163

C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, May 1997. URL http://www.inrialpes.fr/movi/people/Schmid/pub97_1.html. 21, 23, 117, 140

Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *Int. J. Comput. Vision*, 37(2):151–172, June 2000. ISSN 0920-5691. doi: 10.1023/A:1008199403446. URL http://dx.doi.org/10.1023/A:1008199403446. 104

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.*, 10(5):1299–1319, July 1998. ISSN 0899-7667. doi: 10.1162/089976698300017467. URL http://dx.doi.org/10.1162/089976698300017467. 6

Johannes L Schönberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 34

William Robson Schwartz, Aniruddha Kembhavi, David Harwood, and Larry S Davis. Human detection using partial least squares analysis. In *Computer vision, 2009 IEEE 12th international conference on*, pages 24–31. IEEE, 2009. 50, 62

Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *CoRR*, abs/1312.6229, 2013. URL http://arxiv.org/abs/1312.6229. 8, 18, 21, 33

Gayane Shalunts, Yll Haxhimusa, and Robert Sablatnig. *Architectural Style Classification of Building Facade Windows*, pages 280–289. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-24031-7. doi: 10.1007/978-3-642-24031-7_28. URL https://doi.org/10.1007/978-3-642-24031-7_28. 138

Jian Shao, Fei Wu, Chuanfei Ouyang, and Xiao Zhang. Sparse spectral hashing. *Pattern Recogn. Lett.*, 33(3):271–277, February 2012. ISSN 0167-8655. doi: 10.1016/j.patrec.2011.10.018. URL http://dx.doi.org/10.1016/j.patrec.2011.10.018. 18

Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, Aug 2000. ISSN 0162-8828. doi: 10.1109/34.868688. 54

J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, June 2013. doi: 10.1109/CVPR.2013.377. 163

Abhinav Shrivastava, Tomasz Malisiewicz, Abhinav Gupta, and Alexei A. Efros. Data-driven visual similarity for cross-domain image matching. *ACM Trans. Graph.*, 30(6):10, December 2011. ISSN 0730-0301. doi: 10.1145/2070781.2024188. 137, 163

C. Silpa-Anan and R. Hartley. Optimised kd-trees for fast image descriptor matching. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008. doi: 10.1109/CVPR.2008.4587638. 6

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. URL http://arxiv.org/abs/1409.1556. 32, 34

J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proceedings of Ninth IEEE International Conference on Computer Vision*, pages 1470–1477 vol.2, 2003. doi: 10.1109/ICCV.2003.1238663. 6, 11, 18, 35, 67, 68, 88

A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, Dec 2000. ISSN 0162-8828. doi: 10.1109/34.895972. 6, 8

Stephen M. Smith and J. Michael Brady. Susan—a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997. ISSN 1573-1405. doi: 10.1023/A:1007963824710. URL http://dx.doi.org/10.1023/A:1007963824710. 21, 23

Y. Song, X. Chen, X. Wang, Y. Zhang, and J. Li. 6-DOF image localization from massive geo-tagged reference images. *IEEE Transactions on Multimedia*, 18(8):1542–1554, Aug 2016. ISSN 1520-9210. doi: 10.1109/TMM.2016.2568743. 163, 167, 168

Renato O. Stehling, Mario A. Nascimento, and Alexandre X. Falcão. Cell histograms versus color histograms for image representation and retrieval. *Knowledge and Information Systems*, 5(3):315–336, 2003. ISSN 0219-1377. doi: 10.1007/s10115-003-0084-y. URL http://dx.doi.org/10.1007/s10115-003-0084-y. 22

Markus A Stricker and Markus Orengo. Similarity of color images. In *IS&T/SPIE's Symposium on Electronic Imaging: Science & Technology*, pages 381–392. International Society for Optics and Photonics, 1995. 22, 54

Jichao Sun. Local selection of features for image search and annotation. In *ACMMM*, MM '14, pages 655–658, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3063-3. doi: 10.1145/2647868.2654863. URL http://doi.acm.org/10.1145/2647868.2654863. 105

L. Svärm, O. Enqvist, M. Oskarsson, and F. Kahl. Accurate localization and pose estimation for large 3d models. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 532–539, June 2014. doi: 10.1109/CVPR.2014.75. 162

L. Svärm, O. Enqvist, F. Kahl, and M. Oskarsson. City-scale localization for cameras with known vertical direction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(7):1455–1461, July 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2598331. 162

MichaelJ. Swain and DanaH. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. ISSN 0920-5691. doi: 10.1007/BF00130487. URL http://dx.doi.org/10.1007/BF00130487. 21, 22

C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, June 2015. doi: 10.1109/CVPR.2015.7298594. 34, 49

L. Tao, X. Jing, S. Sun, H. Huang, N. Chen, and Y. Lu. Combining surf with mser for image matching. In *2013 IEEE International Conference on Granular Computing (GrC)*, pages 286–290, Dec 2013. doi: 10.1109/GrC.2013.6740423. 61

Xinmei Tian and Yijuan Lu. Discriminative codebook learning for web image search. *Signal Processing*, 93(8):2284 – 2292, 2013. ISSN 0165-1684. doi: http://dx.doi.org/10.1016/j.sigpro.2012.04.018. URL //www.sciencedirect.com/science/article/pii/S0165168412001405. Indexing of Large-Scale Multimedia Signals. 37

E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5):815–830, May 2010. ISSN 0162-8828. doi: 10.1109/TPAMI.2009.77. 57

Chih-Fong Tsai. A review of image retrieval methods for digital cultural heritage resources. *Online Information Review*, 31(2):185–198, 2007. 138

Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991. 18, 21, 22

Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(3):177–280, July 2008. ISSN 1572-2740. doi: 10.1561/0600000017. URL http://dx.doi.org/10.1561/0600000017. 8, 19, 20, 23, 66

Tinne Tuytelaars and Luc Van Gool. Matching widely separated views based on affine invariant regions. *International journal of computer vision*, 59(1):61–85, 2004. 21, 23

O. Tuzel, F. Porikli, and P. Meer. Human detection via classification on riemannian manifolds. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007. doi: 10.1109/CVPR.2007.383197. 50, 62

A. Utenpattanant, O. Chitsobhuk, and A. Khawne. Color descriptor for image retrieval in wavelet domain. In *2006 8th International Conference Advanced Communication Technology*, volume 1, pages 4 pp.–821, Feb 2006. doi: 10.1109/ICACT.2006.206089. 22

J. Valentin, M. Nießner, J. Shotton, A. Fitzgibbon, S. Izadi, and P. Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4400–4408, June 2015. doi: 10.1109/CVPR.2015.7299069. 163

R. E. G. Valenzuela, W. R. Schwartz, and H. Pedrini. Dimensionality Reduction Through PCA over SIFT and SURF Descriptors. In *Proceedings of IEEE Conference on Cybernetics Intelligent Systems*, pages 1–6, 2012. 92

K. van de Sande, T. Gevers, and C. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, Sept 2010. ISSN 0162-8828. doi: 10.1109/TPAMI.2009.154. 59

Joost van de Weijer and Cordelia Schmid. *Coloring Local Feature Extraction*, pages 334–348. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-33835-2. doi: 10.1007/11744047_26. URL https://doi.org/10.1007/11744047_26. 21, 28

Manik Varma and Andrew Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1-2):61–81, 2005. ISSN 0920-5691. doi: 10.1007/s11263-005-4635-4. URL http://dx.doi.org/10.1007/s11263-005-4635-4. 21, 28

A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *2009 IEEE 12th International Conference on Computer Vision*, pages 606–613, Sept 2009. doi: 10.1109/ICCV.2009.5459183. 59

Rémi Vieux, Jenny Benois-Pineau, and Jean-Philippe Domenger. *Content Based Image Retrieval Using Bag-Of-Regions*, pages 507–517. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-27355-1. doi: 10.1007/978-3-642-27355-1_47. URL http://dx.doi.org/10.1007/978-3-642-27355-1_47. 35

Stefanos Vrochidis, Charalambos Doulaverakis, Anastasios Gounaris, Evangelia Nidelkou, Lambros Makris, and Ioannis Kompatsiaris. *A Hybrid Ontology and Visual-based Retrieval Model for Cultural Heritage Multimedia Collections*, pages 1–10. Springer US, Boston, MA, 2009. ISBN 978-0-387-77745-0. doi: 10.1007/978-0-387-77745-0_1. URL http://dx.doi.org/10.1007/978-0-387-77745-0_1. 138

Ji Wan, Dayong Wang, Steven Chu Hong Hoi, Pengcheng Wu, Jianke Zhu, Yongdong Zhang, and Jintao Li. Deep learning for content-based image retrieval: A comprehensive study. In *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, pages 157–166, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3063-3. doi: 10.1145/2647868.2654948. URL http://doi.acm.org/10.1145/2647868.2654948. xv, 8, 33, 34, 137

X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *2009 IEEE 12th International Conference on Computer Vision*, pages 32–39, Sept 2009. doi: 10.1109/ICCV.2009.5459207. 50, 62

Xinggang Wang, Xiong Duan, and Xiang Bai. Deep sketch feature for cross-domain image retrieval. *Neurocomputing*, 207:387 – 397, 2016. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2016.04.046. URL http://www.sciencedirect.com/science/article/pii/S0925231216303198. 137

Adam Williams and Peter Yoon. Content-based image retrieval using joint correlograms. *Multimedia Tools Appl.*, 34(2):239–248, August 2007. ISSN 1380-7501. doi: 10.1007/s11042-006-0087-2. URL http://dx.doi.org/10.1007/s11042-006-0087-2. 22

S. A. J. Winder and M. Brown. Learning local image descriptors. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007. doi: 10.1109/CVPR.2007.382971. 38

Bo Wu and R. Nevatia. Optimizing discrimination-efficiency tradeoff in integrating heterogeneous local features for object detection. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008. doi: 10.1109/CVPR.2008.4587749. 50, 62

Changchang Wu. Towards linear-time incremental structure from motion. In *Proc. of International Conference on 3D Vision*, pages 127–134, June 2013. doi: 10.1109/3DV.2013.25. 162

L. Wu, Y. Hu, M. Li, N. Yu, and X. S. Hua. Scale-invariant visual language modeling for object categorization. *IEEE Transactions on Multimedia*, 11(2):286–294, Feb 2009. ISSN 1520-9210. doi: 10.1109/TMM.2008.2009692. 46

Y. Xia, K. He, F. Wen, and J. Sun. Joint inverted indexing. In *2013 IEEE International Conference on Computer Vision*, pages 3416–3423, Dec 2013. doi: 10.1109/ICCV.2013.424. 6, 18

S Hadi Yaghoubyan, Mohd Aizaini Maarof, Anazida Zainal, and Mahdi Maktabdar Oghaz. A survey of feature extraction techniques in content-based illicit image detection. *Journal of Theoretical and Applied Information Technology*, 87(1):110, 2016. 23

F. Yan, K. Mikolajczyk, J. Kittler, and M. Tahir. A comparison of l_1 norm and l_2 norm multiple kernel svms in image and video classification. In *2009 Seventh International Workshop on Content-Based Multimedia Indexing*, pages 7–12, June 2009. doi: 10.1109/CBMI.2009.44. 59

Ke Yan, Yaowei Wang, Dawei Liang, Tiejun Huang, and Yonghong Tian. Cnn vs. sift for image retrieval: Alternative or complementary? In *Proceedings of the 2016 ACM on Multimedia Conference*, MM '16, pages 407–411, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3603-1. doi: 10.1145/2964284.2967252. URL http://doi.acm.org/10.1145/2964284.2967252. xvi, 34, 49, 50, 62

Huei-Fang Yang, Kevin Lin, and Chu-Song Chen. Cross-batch reference learning for deep classification and retrieval. In *Proceedings of the 2016 ACM on Multimedia Conference*, MM '16, pages 1237–1246, New York, NY, USA, 2016a. ACM. ISBN 978-1-4503-3603-1. doi: 10.1145/2964284.2964324. URL http://doi.acm.org/10.1145/2964284.2964324. 34

Jianchao Yang, Kai Yu, Yihong Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1794–1801, June 2009. doi: 10.1109/CVPR.2009.5206757. 38

Jianwei Yang, Lifeng Liu, Tianzi Jiang, and Yong Fan. A modified gabor filter design method for fingerprint image enhancement. *Pattern Recognition Letters*, 24(12):1805 – 1817, 2003. ISSN 0167-8655. doi: https://doi.org/10.1016/S0167-8655(03)00005-9. URL http://www.sciencedirect.com/science/article/pii/S0167865503000059. 53

S. Yang and C. Zhao. A fusing algorithm of bag-of-features model and fisher linear discriminative analysis in image classification. In *2012 IEEE International Conference on Information Science and Technology*, pages 380–383, March 2012. doi: 10.1109/ICIST.2012.6221672. 38

Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze. Designing energy-efficient convolutional neural networks using energy-aware pruning. *CoRR*, abs/1611.05128, 2016b. URL http://arxiv.org/abs/1611.05128. 34

S. H. Yen, M. H. Hsieh, C. J. Wang, and H. J. Lin. A content-based painting image retrieval system based on adaboost algorithm. In *2006 IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 2407–2412, Oct 2006. doi: 10.1109/ICSMC.2006.385224. 138

Hui Yu, Mingjing Li, Hong-Jiang Zhang, and Jufu Feng. Color texture moments for content-based image retrieval. In *Proceedings. International Conference on Image Processing*, volume 3, pages 929–932 vol.3, June 2002. doi: 10.1109/ICIP.2002.1039125. 41, 42

Jing Yu, Zengchang Qin, Tao Wan, and Xi Zhang. Feature Integration Analysis of Bag-of-Features Model for Image Retrieval. *Neurocomputing*, 120:355 – 364, 2013. xv, 39, 41, 42, 62, 89, 90, 91, 100, 122, 123

Jun Yue, Zhenbo Li, Lu Liu, and Zetian Fu. Content-based Image Retrieval using Color and Texture Fused Features. *Mathematical and Computer Modelling*, 54(3–4):1121–1127, 2011. ISSN 0895-7177. Mathematical and Computer Modeling in Agriculture. 39, 43, 62

Amir Roshan Zamir and Mubarak Shah. Accurate image localization based on google maps street view. In *ECCV*, pages 255–268, 2010. ISBN 3-642-15560-X, 978-3-642-15560-4. 163

Matthew D. Zeiler and Rob Fergus. *Visualizing and Understanding Convolutional Networks*, pages 818–833. Springer International Publishing, Cham, 2014. ISBN 978-3-319-10590-1. doi: 10.1007/978-3-319-10590-1_53. URL http://dx.doi.org/10.1007/978-3-319-10590-1_53. 18

Bernhard Zeisl, Pierre Fite Georgel, Florian Schweiger, Eckehard Steinbach, and Nassir Navab. Estimation of location uncertainty for scale invariant feature points. In *Proceedings of the British Machine Vision Conference*, pages 57.1–57.12. BMVA Press, 2009. ISBN 1-901725-39-1. doi:10.5244/C.23.57. 104, 105

Jerry Zhang, Aaron Hallquist, Eric Liang, and Avideh Zakhor. Location-based image retrieval for urban environments. In *ICIP*, pages 3677–3680, 2011a. 163

L. Zhang, L. Wang, and W. Lin. Semisupervised biased maximum margin analysis for interactive image retrieval. *IEEE Transactions on Image Processing*, 21(4):2294–2308, April 2012a. ISSN 1057-7149. doi: 10.1109/TIP.2011.2177846. 19

Shaoting Zhang, Ming Yang, Timothee Cour, Kai Yu, and Dimitris N. Metaxas. *Computer Vision – ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part II*, chapter Query Specific Fusion for Image Retrieval, pages 660–673. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012b. ISBN 978-3-642-33709-3. doi: 10.1007/978-3-642-33709-3_47. URL http://dx.doi.org/10.1007/978-3-642-33709-3_47. 9, 58, 63

Weifeng Zhang, Zengchang Qin, and Tao Wan. Image Scene Categorization using Multi-Bag-of-Features. In *Proceedings of International Conference on Machine Learning and Cybernetics*, volume 4, pages 1804–1808, 2011b. doi: 10.1109/ICMLC.2011.6017012. xvi, 5, 19, 54, 55, 63

Yu Zhang, Stephane Bres, and Liming Chen. *Visual Concept Detection and Annotation via Multiple Kernel Learning of Multiple Models*, pages 581–590. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-41184-7. doi: 10.1007/978-3-642-41184-7_59. URL http://dx.doi.org/10.1007/978-3-642-41184-7_59. 39

Liang Zheng, Shengjin Wang, Jingdong Wang, and Qi Tian. Accurate image search with multi-scale contextual evidences. *International Journal of Computer Vision*, 120(1):1–13, 2016. ISSN 1573-1405. doi: 10.1007/s11263-016-0889-2. URL http://dx.doi.org/10.1007/s11263-016-0889-2. 49

D. Zhou, X. Li, and Y. J. Zhang. A novel cnn-based match kernel for image retrieval. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2445–2449, Sept 2016. doi: 10.1109/ICIP.2016.7532798. 34

Li Zhou, Zongtan Zhou, and Dewen Hu. Scene classification using a multi-resolution bag-of-features model. *Pattern Recognition*, 46(1):424 – 433, 2013. ISSN 0031-3203. doi: http://dx.doi.org/10.1016/j.patcog.2012.07.017. URL //www.sciencedirect.com/science/article/pii/S0031320312003330. xvi, 54, 55, 56

Xiang Sean Zhou and T. S. Huang. Unifying keywords and visual contents in image retrieval. *IEEE MultiMedia*, 9(2):23–33, Apr 2002. ISSN 1070-986X. doi: 10.1109/93.998050. 38

Yang Zhou, Dan Zeng, Shiliang Zhang, and Qi Tian. Augmented feature fusion for image retrieval system. In *ICMR*, ICMR '15, pages 447–450, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3274-3. doi: 10.1145/2671188.2749288. URL http://doi.acm.org/10.1145/2671188.2749288. 105

C. Zhu, C. E. Bichot, and L. Chen. Multi-scale color local binary patterns for visual object classes recognition. In *2010 20th International Conference on Pattern Recognition*, pages 3065–3068, Aug 2010. doi: 10.1109/ICPR.2010.751. 57

Chao Zhu, Charles-Edmond Bichot, and Liming Chen. Image region description using orthogonal combination of local binary patterns enhanced with color information. *Pattern Recognition*, 46(7):1949 – 1963, 2013. ISSN 0031-3203. doi: http://dx.doi.org/10.1016/j.patcog.2013.01.003. URL http://www.sciencedirect.com/science/article/pii/S0031320313000228. 30

Li Zhuo, Jing Zhang, Yingdi Zhao, and Shiwei Zhao. Compressed domain based pornographic image recognition using multi-cost sensitive decision trees. *Signal Processing*, 93(8):2126 – 2139, 2013. ISSN 0165-1684. doi: http://dx.doi.org/10.1016/j.sigpro.2012.07.003. URL //www.sciencedirect.com/science/article/pii/S0165168412002319. Indexing of Large-Scale Multimedia Signals. 18, 22

Larry Zitnick and Piotr Dollar. Edge boxes: Locating object proposals from edges. In *ECCV*. European Conference on Computer Vision, September

2014. URL https://www.microsoft.com/en-us/research/publication/edge-boxes-locating-object-proposals-from-edges/. 49