UNIVERSITAT POLITÈCNICA DE CATALUNYA

Towards precise outdoor localisation based on image recognition

by

Tomasz Piotr Trzciński

Supervised by

Master Thesis Director: **Ferran Marqués**
Master Thesis Tutor: **Tomasz Adamek**

A thesis submitted in partial fulfillment for the
degree of Master of Science

in
Research on Information and Communication Technologies
The Signal Theory and Communication Department

July 2010

Declaration of Authorship

I, Tomasz Piotr Trzciński, declare that this thesis titled, 'Towards precise outdoor localisation based on image recognition' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: *Tomasz Trzciński*

Date: *July 2010*

“Information is the resolution of uncertainty.”

Claude Shannon

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Abstract

Research on Information and Communication Technologies
The Signal Theory and Communication Department

Master of Science

by Tomasz Piotr Trzciński

In recent years significant progress has been made in the field of visual search, mostly due to the introduction of powerful local image features. At the same time, a rapid development of mobile platforms enabled deployment of image retrieval systems on mobile devices. Various mobile applications of the visual search have been proposed, one of the most interesting being geo-localisation service based on image recognition and other positioning information. This thesis attempts to advance the existing visual search system developed at Telefónica I+D Barcelona so it could be used for high precision geo-localisation. In order to do so, this dissertation tackles two significant challenges of image retrieval: improvement in robustness of the detection of similarities between images and increase of discriminatory power.

The work advances the state-of-the-art with three main contributions. The first contribution consists of the development of an evaluation framework for visual search engine. Since the assessment of any complex system is crucial for its development and analysis, an exhaustive set of evaluation measures is selected from the relevant literature and implemented. Furthermore, several datasets along with the corresponding information about correct correspondences between the images have been gathered and unified. The second contribution considers the representation of image features describing salient regions and attempts to alleviate the quantisation effects introduced during its creation. A mechanism that in the literature is commonly referred to as *soft assignment* is adapted to a visual search engine of Telefónica I+D with several extensions. The third and final contribution consists of a post-processing stage that increases discriminative power by verification of local correspondences' spatial layout. The performance and generality of the proposed solutions has been analysed based on extensive evaluation using the framework proposed in this work.

Acknowledgements

I would like to express my appreciation to my tutor, Tomasz Adamek, for his time, understanding and patience. Thanks for the support and all the clarifications presented along with a deep insight in the world of research. Thanks to all the members of the Augmented Reality Initiative at Telefónica Investigación y Desarollo, Barcelona: Luis Javier Chico, Arturo Bonnin Llofriu and Roger Gimeno Hernandez. Special thanks to the leader of MobileAR Group, David Marimón Sanjuan for giving me the opportunity to be part of the research group.

My gratitude also goes to my Master Thesis Director, Ferran Marqus, there are not enough words to describe the inspiration and motivation I received from him. Thanks for being my mentor and for all your guidance.

I would like to thank my parents for giving me a lifetime opportunity to attend the universities I desired and for being my everlasting incentive to reach for the sky. Mum, Dad, *per aspera ad astra.*

The most special thanks goes to my best friend and true soul mate, my fiancée. For her unconditional support and love through all these years.

Contents

Dedicated to my fiancée.

Chapter 1

Introduction

1.1 Introduction to Visual Search

With the increasing popularity of capturing devices, such as cameras, voice and video recorders, an astonishing increase in the amount of digital content has been observed. Since the content of data that is being both produced and consumed on a daily basis is continuously growing, the importance of a reliable and computationally feasible mechanism to analyse that data increases accordingly. Nowadays, capacity of digital contents being generated everyday by hundreds of millions of cameras is counted in exabytes (1000^6 bytes). Therefore, there is an obvious need for computer-aided algorithms capable of grasping the information hidden within the stream of bytes. One of the most complex problems that the algorithms are facing is the analysis of visual content – photos, videos, snapshots, *etc.*

The advancements in computational power of modern CPUs along with the results of scientific work led way to the birth of a new scientific branch – computer vision – that addresses this problem. There exist many different aspects of computer vision, *e.g.* tracking, scene reconstruction and pose estimation. Among them, visual search, which can be defined as finding all the images from a larger collection that have specific content, quickly attracted a lot of researchers' attention. This is not only due to the fact that visual search, also referred to as content-based image retrieval, combines the knowledge and expertise from two fields – information retrieval and image processing. Furthermore, it can be employed in various applications: high precision geo-localisation [1], augmented reality or near-duplicate detection [2]. Indeed, the visual recognition technology is an intuitive extension of contextual services terminals that enables them to perceive the surrounding environment in a human alike way.

Retrieving an object from a database of images is now reaching some maturity. It is still a challenging problem because object's visual appearance may be very different due to viewpoint and lighting changes or partial occlusion, but successful methods now exist [3]. Nevertheless, the state of the art visual search systems still face several problems, such as scalability of the existing solutions, their reliability, robustness and computational cost.

Rapid development of mobile platforms has inspired researchers around the world to improve the performance of the existing visual search systems and deploy them on mobile devices. Since quick identification of objects in the scene provides a powerful new source of contextual information, portable appliances seem to provide perfect application platform for image retrieval systems. Equipped with efficient CPUs, high resolution cameras and additional sensors, they can be easily used in various everyday scenarios, *e.g.* while commuting or travelling.

Other possible mobile applications of image retrieval include high precision navigation based on image recognition, augmented reality services, content-aware photography, and various tourist or entertainment applications. Using the features of the portable devices (such as compass, GPS, *etc.*) it is becoming possible to combine the information about the contents of the picture with the additional data.

1.2 Motivation

As mentioned above, the main challenges of the reliable visual search include scalability, robustness and computational cost. Many techniques have been proposed in the last 20 years that were able to tackle some of those problems. However, most of them were tightly constrained to particular applications. The other methods suffer from performance drop with the increasing number of images in the dataset. An extensive overview of those solutions can be found in [4, 5].

One of the main breakthroughs in the field of visual search came with a more generic approach inspired by text retrieval techniques. This innovative solution, termed “Video Google”, was presented in [3] and [6]. In this approach, each salient region is represented by a feature descriptor that is quantised into a cluster, a so-called *visual word*. This descriptor is supposed to be invariant to illumination, rotation and scale changes. After descriptors' quantisation, an image is represented as a set of visual words – clusters that were selected by mapping of feature descriptors. The similarities between the images are found through comparison of their visual word representations. The “Video Google” approach was proved to be robust and relatively efficient in terms of computational power, however, it does not fully eliminate major problems of visual search.

This thesis focus on two significant challenges of visual content retrieval systems: improvement in robustness of the detection of similarities between images and increase of discriminatory power. Even though robustness of a system may partially depend on the descriptor's invariance to conditions' changes, it shall be also accounted for within the search engine. If it is not, the similarities between the objects might be lost and the performance of the system drops. However, improved robustness is prone to increase the number of similarities which may lead to discovery of false correspondences between the images. One of the possible solution of this problem is the increase of discriminative power. While improving system performance, we should bear in mind potential mobile application of such system. Hence, we should take into account the trade-off between performance and computational cost of implemented solution and adjust it to the application scenario.

In order to analyse the performance of any complex system, *e.g.* visual search engine, an exhaustive evaluation tool is needed. Meaningful assessment is a crucial part of development of algorithms within this work. It is also necessary to answer significant questions regarding the incorporation of the improvements proposed in this thesis. Thus, one of the challenges to be faced during development of this thesis is development of exhaustive evaluation framework.

In this work, we focus on the application of visual search to outdoor image recognition, so that high precision geo-localisation on mobile devices becomes feasible. Nevertheless, we will also present the results for other types of objects to confirm the generality of the proposed solutions.

One should know that the representation of image patches used for matching depends on the training data. In the case of the geo-localisation systems the assumption of using ideal training data is unrealistic, since the images describing outdoor scenes are not static. Thus, while developing this thesis, we analyse the impact of different collections of images used for training on the performance quality and attempt to reduce this influence.

1.3 Research Contribution

The research done while developing this Master thesis advances the state of the art visual search system developed at Telefónica I+D (discussed in chapter 3) with three main contributions:

- **Development of evaluation environment**
- **Soft assignment of visual words to key-points**
- **Spatial consistency verification of initial matches**

1.3.1 Evaluation Framework

As mentioned above, one of the main contributions of this thesis was development of exhaustive evaluation framework. We have used it extensively not only for system assessment, but also during development of extensions proposed in this thesis.

The subject of objective evaluation of visual search system performance has not been widely discussed in the literature, despite the number of publications on the topic of image retrieval. Most of the evaluation measures proposed in the literature were used to evaluate very specific collections. Thus, taking into consideration the subjective assessment of user and adapting existing methods of measurement from information retrieval [7], we propose a complete set of evaluation methods.

Furthermore, to obtain statistically significant results we gather and prepare several datasets of different types: generic and outdoor objects, consumer goods items. Since the developed evaluation tool is *ground truth* based, heterogeneous information about correct correspondences between the images is collected and unified.

1.3.2 Soft Assignment

The state of the art visual search engines are inspired by the concept of text retrieval. The overlying idea of this approach is classifying particular regions of images using descriptors and mapping them to *visual words* – a set of clusters in description space. However, since quantisation effect is introduced through this assignment, correspondence between two similar descriptors might be lost during this assignment.

In this thesis we try to adapt the methods of quantisation effects' reduction proposed by J. Philbin in [6] to the visual search system developed at Telefónica I+D. We attempt to explore techniques used to map the region descriptors into more than one visual word through a process called *soft assignment*. As a result, we obtain a new visual word representation of salient regions that increases the robustness of the search engine.

While developing the thesis, several variants of soft assignment are implemented and analysed. Different algorithms are tested, optimised and incorporated into the visual search engine architecture. Due to the system complexity, the analysis of the final results of soft assignment turns out not to be informative enough. Thus, we propose a set of novel experiments that allow us to scrutinise the influence of soft assignment excluding potential interferences with other components of search engine. Thanks to those experiments, constructive conclusions for integration of the soft assignment into the search engine mechanism are obtained.

Finally, the soft assignment mechanism is integrated with the search engine and, after an exhaustive evaluation, the most promising configuration is identified. The benefits of soft assignment are evaluated and discussed in details.

1.3.3 Spatial Consistency Verification

According to the "Video Google" approach, each salient region is represented with feature descriptor quantised to visual word. In order to retrieve images depicting the same object that appears in the query, comparison of images' visual word representation is performed. However, this may not be sufficient for precise image retrieval, since it is probable that there are several image patches in the scene that have identical descriptors, even though they do not represent corresponding patches.

One of the possible solutions of this probable is an additional post-processing stage that increases discriminative power by verification of local correspondences' spatial layout. In this thesis, we propose to implement this stage relying on algorithm that estimates transformation between query and reference images. This solution has been proven to be successfully implemented in various systems [3, 6, 8]. Here, we develop a version of these solutions that complements the initial spatial consistency verification stage that has been already implemented in the visual search engine developed at Telefónica I+D. We introduce extension to the solution described in literature that builds up on the weighting scheme implemented in the original version of visual search engine. This so-called *spatial consistency verification* module has been found useful when combined with soft assignment discussed earlier in this section.

During the development of this thesis we implement and evaluate the additional spatial verification stage. We fine tune parameters of this post-processing element. We also discuss in details the benefits of this approach. Finally, we propose future improvements and possible extensions of implemented solution.

1.4 Thesis Organisation

The remainder of this thesis is organised in the following way:

Chapter 2 introduces various basic ideas, methods and algorithms which are important throughout this work. Specifically, it briefly discusses local visual feature types and image matching algorithms with a special emphasis on visual vocabulary based approaches.

Chapter 3 briefly describes the visual search system that has been developed at Telefónica I+D Barcelona. This system provided the starting point for the developments proposed in this thesis. The main ideas behind the local feature descriptor DART are described and the visual search engine used in the experiments of this thesis is explained in details.

Chapter 4 discusses the importance of exhaustive evaluation in the development of visual search engine and difficulties that correspond to this evaluation. Then, evaluation measures that are commonly used to evaluate image recognition systems are described in detail. Their advantages, potential flaws and implementation difficulties are discussed. Finally, this chapter provides the description of all dataset gathered and adapted to be used within this thesis.

Chapter 5 describes the soft assignment mechanism proposed as a solution for alleviating quantization effects in visual search engine relying on visual word dictionaries. Specifically, it describes an adaptation of the soft assignment proposed in the literature to the system developed at Telefónica I+D. Since several extensions and configurations are implemented and evaluated, this chapter includes the implementation details and results of the experiments performed to optimize the mechanism. Finally, the influence of the soft assignment on system performance is presented and the final conclusions are drawn.

Chapter 6 introduces the spatial consistency verification mechanism proposed to increase discriminatory power of the visual search engine developed at Telefónica I+D. RANSAC inspired algorithm is used for similarity transformation model estimation and identification of local correspondences consistent with that model. Finally, the results of visual search engine evaluation are presented and the final conclusions are drawn.

Chapter 7 contains detailed results of the visual search engine with all the solutions proposed in this thesis. The performance of the system before and after introduction of extensions is compared and discussed in details.

Chapter 8 concludes the work done within this thesis and discusses further research directions and additional amendments.

Chapter 2

State of the Art in Visual Search

In this chapter, basic ideas that this thesis builds on are introduced. Furthermore, the current state of the art systems are described providing the benchmark for the research done within this thesis development.

2.1 Basic Ideas

The concept that is fundamental for most computer vision systems is the definition of image features. Depending on the application, features should have particular properties. In order to enable matching of images, they should provide certain discriminative power to the classification process, so reliable object recognition is possible. Furthermore, those properties shall be invariant to image scaling, rotation, illumination and viewpoint changes. However, full invariance is not achievable (due to clutter or noise) neither for humans nor for computers.

Since there exists a trade-off between discriminative power and invariance to condition changes, many differentiated feature models have been proposed. Extensive review of several features can be found in [9–11].

Features can be described using different properties, *e.g.* dominating colour, variation of global colour histograms, histogram or gradient of grey values, sum of all pixel values, orientation of a particular set of pixels, *etc.*

We can divide features used to describe the contents of an image into two different categories:

1. **Global**: features that are computed for entire image.
2. **Local**: features that describe characteristics of image patch.

Global features are computed for the whole image and describe it with a compact representation. Mostly, they analyse texture and shape of objects apparent in the image. Thus, their robustness to occlusions, rotation and different scene compositions is relatively low. Thus, in this thesis we will not focus on this type of features.

Local features, however, can be used for image matching in much more robust way – they decompose the image into localised image patch descriptors around interest points. That increases their reliability.

Because of the above properties of local features, the most recent research clearly shows that in applications, such as visual search engine, they give much better results than global descriptors.

2.2 Local Features Detection and Description

Images that portrait similar scenes or objects are identified through comparison of the features. If the features from two images match each other, those images are believed to describe the same object. Hence, due to importance of local features in image matching, in this section we will thoroughly analyse the process of local features detection and description.

In most cases, describing an image with a set of local features can be divided in two parts:

1. **Feature detection**: The detection aims at finding the regions where key-points exist.
2. **Feature description**: The invariant descriptor of those points needs to be computed.

During the last decade, a lot of research on robust algorithms for detection of stable, invariant key-points was made, among which Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) algorithms are the most known. In the same time, researchers were focused on reducing computational power needed for reliable features description.

The regions of interest where most informative contents are localised can be found through feature detection algorithms or simply by uniform selection. Mostly, those regions are defined either by texture or edges. Therefore, we obtain a set of distinctive points that characterise a scene or an object. However, the distribution of those points is far from uniform – it is highly concentrated in the parts of image where the most relevant image contents are concentrated.

It is worth noticing that even though intuitively colour information might be believed to convey essential information about the contents of the image, only grey-scale images are used in most of the algorithms for feature extraction. Hence, all the algorithms and experiments described in this thesis will rely on grey-scale images, unless stated otherwise.

Feature detection

The main objective of the first phase of image description is localising points of interest which convey the most relevant information about the particular image. A plethora of different detection algorithms exists – a detailed description can be found in [12]. They all aim at detecting characteristic patches of different sizes and shapes – circular, square, elliptic, *etc.*

A good feature detector locates points that can be detected repeatedly and independently of the illumination, scale, angle and compression changes. The main criterion to judge the quality of detector is its invariance to those changes measured by the repeatability value [12]. It verifies if the same interest point can be reliably detected at the same position after image transformations have been performed.

Reliable identification of object in image recognition requires at least 3 features per object be correctly matched [13]. Intuitively, we would like to describe an image with high number of features in order to correctly identify all important objects present in the scene. Nevertheless, there exists a trade-off between the number of local features per image and the computational time that is required to process them: with increasing number of descriptors, we can observe proportional increment of computations needed. Moreover, additional features may introduce noise and redundancy. Thus, we are interested in defining the optimal number of descriptors per image so correct matching is possible within a reasonable time span.

Another aspect of the feature detection algorithm that has to be taken into account is a trade-off between quality of the detection (*e.g.* measured by repeatability value) and its cost (measured in time and power needed to perform this operation). The second factor is especially significant in computer vision applications involving real-time video processing, such as object tracking.

Feature description

The next step after the localisation of points of interest is to describe them. The region around the point of interest is encoded using a descriptor vector based on *e.g.* the histogram of gradients in its neighbourhood. The similarities between vectors will then be used to match two images and verify if they depict similar objects. When two alike descriptors are found in two different images, correspondences are identified and the degree of similarity between the images, which depends on the number of matches, is computed.

The descriptor should define characteristics of an image patch in a discriminative manner, so two distinct objects can be distinguished. At the same time, it shall not differ with the changes of size, scale, illumination or viewpoint. Moreover, the vector representation compactness and corresponding computational power needed for vector matching must be also accounted for.

A wide range of description methods has been proposed, one of them being a simple representation of an image with values of intensity of the pixels. Afterwards, *e.g.* in order to match two images, cross-correlation between those representations can be computed [14]. However, it can be easily proven that this descriptor is not invariant to the previously mentioned changes, such as illumination or scale changes. What is more, processing of high dimensional vectors, obtained this way, with accurate precision would imply computational cost and memory space that is not feasible for modern devices. Taken all the above into account, more complex algorithms of feature description have been proposed.

Below some of the most popular descriptors are briefly discussed. They include features which are invariant to scale changes (SIFT, SURF) as well as features invariant to affine changes (MSER). An exhaustive evaluation of various feature descriptors can be found in [10].

2.2.1 SIFT

Scale Invariant Feature Transform (SIFT) [13] defines both feature detector and descriptor. It is both scale (hence the name) and rotation invariant.

The SIFT detector is by all means the most popular detection algorithm used in state of the art computer vision systems. This partially is due to the fact that its repeatability value for scale, viewpoint and illumination changes is relatively high. Since SIFT detector builds on scale-space theory, key-points are well localised not only in frequency, but also in space, enabling correct detection of objects that are occluded, partially visible or appearing in the presence of noise.

The scale invariance is obtained through repeatable convolution of the image with a Gaussian at several scales. Outcome of this process is so-called scale space pyramid of convolved images. Only the points that are stable across scales are detected. Their stability is defined using an approximation of Laplacian – Difference-of-Gaussians (DoG) approach – where the subtraction of convolved images at subsequent scales is performed.

Let G be a Gaussian and L input image convolved with different Gaussians, Difference-of-Gaussians can be defined as follows:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma))I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{2.1}$$

After the computation of DoG, local maxima are identified. If they appear at a single pixel across scales, stability of the key-point is confirmed. Edge responses are eliminated through additional refinement steps. Finally, the dominant orientation of the point is found through the analysis of the radial histogram of gradients in a circular neighbourhood of the detected point. This enables descriptor's rotational invariance.

The region around the interest point is divided into orientation histograms on 4×4 pixel neighbourhoods. The histograms contain 8 bins each and are relative to the key-point dominant orientation. Each descriptor consists of 4×4 array of 16 histograms around the key-point. Hence, the final dimension of the SIFT feature vector equals to $4 \times 4 \times 8 = 128$. In the post-processing phase features vector are normalised to enhance invariance to illumination changes.

It should be noted, that the design of the SIFT descriptor was inspired by the perception of human vision system. Edelman, Intrator, and Poggio demonstrated in 1997 that complex neurons in primary visual cortex respond to a gradient at a particular orientation and spatial frequency [15]. However, location of the gradient on the retina may shift slightly without being noticed by human. Edelman et al. hypothesised about the function of these complex neurons and performed detailed experiment which showed that matching gradients while allowing for shifts in their position results in much better classification under 3D rotation. Even though SIFT descriptor was inspired by this idea, its original implementation allows for positional shift using a different computational mechanism.

2.2.2 SURF

Building on the advances presented by D. Lowe in the SIFT algorithm, a speeded-up method was proposed by H. Bay *et al.* in [16]. Speeded Up Robust Features (SURF) is particularly fast and compact. Furthermore, it is scale and rotation invariant.

As SIFT uses Differences-of-Gaussian for detection of points of interest, SURF is based on the Determinant-of-Hessian and employs an efficient approximation of the Hessian. Since Gaussian second order derivatives can be simplified to step functions, the Hessian detector can be approximated with simple box filters. This simplification allows using integral images which speeds up the computations even further.

Just like the SIFT descriptor, the SURF descriptor is computed for 4×4 square sub-regions. For each of the sub-regions Haar wavelet responses (horizontal d_x and vertical d_y) are used to encode characteristic features of image patch. Due to the fact that integral images are used, the computational power needed for calculating filter responses is relatively low. The wavelet responses in each orientation are summed up and so are the absolute values of the responses.

Hence, each sub-region has a four-dimensional description vector for its underlying intensity structure $\mathbf{v} = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$. As a result of all the above operations, a description vector of length 64 is obtained (4×4 sub-region vectors, 4 dimensions each). Invariance to rotation is obtained by expressing the gradients in relation to the dominant orientation and illumination invariance is achieved by the gradients themselves.

2.2.3 Hessian-Affine

In order to obtain higher robustness than the one provided by scale and rotation invariant detectors, the affine model was introduced. The Hessian-Affine detector belongs to a class of affine-covariant detectors that can cope with affine changes [12]. The features are localised using the determinant of Hessian that has been proven to provide accurate results. The main advance over the previously mentioned detectors is the introduction of adaptive elliptic shape used for the description of interest points. Instead of square or circular neighbourhood, Hessian-Affine uses ellipse determined with the second moment matrix of the intensity gradient.

However, since the Hessian-Affine detector does not come with its own descriptor, detected regions can be described using other descriptor types, *e.g.* SURF. For description purposes the region around the point of interest is normalised to circle or square so it complies with the descriptor requirements.

2.2.4 MSER

Maximally Stable Extrema Regions (MSER) is another affine-covariant detector [17]. It is based on connected components of an appropriately thresholded image, instead of the scale-space Gaussian methods. Extrema regions are defined as the regions which consist of pixels that have either higher (bright extrema regions) or lower (dark extrema regions) intensity than all the pixels on its outer boundary. Maximally stable regions are detected through a thresholding process. While thresholding an input image with various threshold values, the maximally stable regions' binarization does not change.

The region detected around the point of interest is an adaptive ellipse that is later fitted to the area of interest and normalised so the descriptor can be computed. Regions detected with MSER are invariant to illumination changes, because only the differences in the pixels' intensities are used, not their absolute values.

2.3 Visual Content Retrieval

The main principle behind visual content retrieval is delivery of the results in response to photos instead of words. Typically, in order to provide such results, image matching based on comparison of images' local features needs to be performed.

Comparison of two images using local features can be divided in the following way:

1. **Feature extraction**: localise points of interest and describe them appropriately.

2. **Feature matching**: for each feature from a query image find a corresponding feature from another image or a dataset of images. This often boils down to nearest neighbour search in n-dimensional feature space.

3. **Recognition**: judging by the number of correspondences found and their characteristics decide if both images present the same object or scene. This may involve additional post-processing.

The first part of the above process has already been discussed in the previous section. The third part will be explained briefly later in this chapter and discussed thoroughly in the next chapter where the visual search engine developed at Telefónica I+D will be presented. In this section we will focus on the second part – feature matching.

It is essential to note that, as far as the computational cost is concerned, feature matching presents a challenging problem of n-dimensional feature space search. It becomes particularly crucial for online systems working with datasets of reference images counted in thousands of pictures. Therefore, computational time of nearest neighbour search should be possibly short to enable matching within a reasonable time span. It becomes obvious that calculating the distances to all the features from all the images of such a dataset is a gargantuan and infeasible task. Thus, several different approaches to the problem of fast nearest neighbour search have been proposed, among them randomised k-d trees [13] being one example. Furthermore, it is not clear if the nearest neighbour search should be performed using Euclidean, Mahalanobis or another distance measure, which may also add complexity to the existing problem. Many of the proposed solutions perform relatively well only with low dimensional spaces. They also seem to suffer from yet another problem: finding the best match may not suffice for all possible applications. This is due to the boundaries of the search that depend on the characteristic of the matching that is performed, *e.g.* object class detection versus detection of specific objects.

In response to this demanding task, a few solutions have been proposed, the *visual words* approach being one of the most successful [3, 18]. In this solution, greatly inspired by text retrieval systems, descriptor space quantisation is performed, so every feature extracted from an image is mapped to a cluster – so-called *visual word*. The quantisation process relies on clustering structures: visual word vocabularies, also defined through a text retrieval analogy as *dictionaries*. Afterwards, matching of images represented by vectors with visual word IDs instead of key-points is performed through comparison of those vectors. This solution, often referred to as "Video Google" has been proven to provide an efficient and scalable recognition mechanism even for large collection of images. Detailed description of this approach can be found below.

2.3.1 Visual Words Representation

High-dimensional descriptors for "salient" image regions are quantised and clustered into a vocabulary of visual words. Each region descriptor can be assigned to the nearest visual word according to this clustering. Images are then represented through *bags of visual words* – groups of features describing the same visual primitive element. This representation is then used for indexing and retrieval in the next stages of processing. Typically, no spatial information about the visual words in the image is conveyed. Examples of visual words can be seen in Fig. 2.1.

There are numerous similarities between the discussed visual words approach and the representation of documents through bags of words used in text retrieval. Analogies between those two

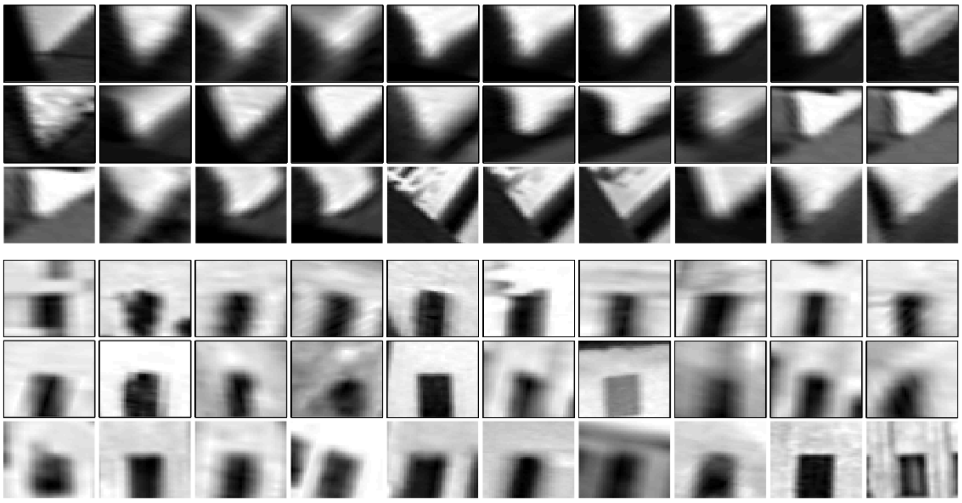


FIGURE 2.1: Two groups of image patches – each assigned to its visual word. *Source:* [3]

mechanisms have been widely analysed and many attempts have been made in order to put in use the knowledge of text retrieval systems in analogical visual search systems [3].

2.3.2 Dictionaries

Visual vocabularies – also called *dictionaries* through text analogy – that are used to represent descriptors in compact way are obtained by clustering a set of example feature descriptors in multi-dimensional vector space. Typically, the dataset that is used to create a dictionary is clustered into k representative clusters. From the algorithmic point of view, each of the clusters is treated as a visual word. The characteristics of the clusters vary widely depending on the features that were used for dictionary creation, clustering algorithm and algorithm's parameters. This can, however, work as an advantage, since we can adjust the quality of dictionary and its clusters to the application. Also the number of clusters k can be optimised for a particular task, *e.g.* a few hundreds suffice for object class recognition [19, 20], but for reliable retrieval of a specific object from a large dataset an increase of the number of clusters up to 1 million is needed [8]. The examples show how the variability within the cluster differs depending on the application: in object class recognition each cluster can be highly differentiated, whereas for retrieval of a specific object it should be much more homogeneous.

There are several clustering methods that can be used to create visual vocabularies. A summary of the most relevant ones can be found below.

Clustering

The main objective of data clustering is to partition data into a set of groups called *clusters*. A clustering algorithm should take into account the similarities between the samples of data so the final clusters will consist of similar items. The type of method used depends highly on the dataset features: size, homogeneity, type of data it stores, *etc.* Since within the frame of this thesis we will be using clustering for creation of visual word dictionaries of multiple dimensions, main emphasis will be put on algorithms that handle such multi-dimensional vectors. Generally, clustering algorithms identify areas of high density in vector space where a significant subset of data points is located and try to build the cluster around those areas.

A clustering algorithm that is one of the most commonly used in the field of image matching is *k-means*. Due to its great importance for the creation of dictionaries used in this thesis, it will be thoroughly described below. It shall be noted, however, that other methods (K-medoids, histogram binning, *etc.*) can be used as well for visual vocabularies creation.

The main aim of the k-means clustering is to partition the observations so each one of them will belong to a cluster with the closest mean. More precisely, for n observations and $k \leq n$ clusters, the algorithm tries to minimise total intra-cluster variance

$$V_{\text{total}} = \sum_{i=1}^{k} \sum_{x_j \in c_i} (x_j - \mu_i)^2 \tag{2.2}$$

where c_i stands for i-th cluster for $i = 1, \ldots, k$ and μ_i is the mean of all points $x_j \in c_i$.

One of the most common iterative refinement heuristic that is used for k-means clustering is Lloyd's algorithm. It is both simple to implement and execute. The algorithm needs to be initialised with k centroids as representatives of the clusters. Afterwards, data points are assigned to the closest centroid according to the distance measure computed. Then, a new mean being the centre of the centroid is calculated. The iterations are repeated until no point changes its cluster or until the number of iterations reaches a predefined threshold.

However elegant the simplicity of this algorithm might appear, it does not come without its cost. First of all, the selection of parameter k is not trivial. Since it affects the outcome of the algorithm significantly, the possible solution is to try out a few values of k and choose the optimal one. Nevertheless, the time complexity of this solution has to be taken into account. For n data points of d dimensions and l iterations, the complexity of k-means algorithm is $O(N \cdot d \cdot k \cdot l)$. Secondly, the initialisation of the algorithm is also a challenging task, since for different initialisations different outcomes shall be expected. Because of the algorithm's propensity to lock itself in local minima, most common work around this problem is to try different initialisations and choose the best one or pick random data points and initialise clustering with them.

In this thesis, k-means clustering will be used to create visual word vocabularies. Extracted feature descriptors will be clustered into visual words used later on for image matching.

As far as the measuring of cluster quality is concerned, there are two different approaches that can be used: general statistical analysis or, if the clustered structures are used as a module in a pipeline processing, analysis of the output of a system. The latter one is more practical, since the statistical analysis of the clusters might not reveal its final influence on the system performance. However, the outcome of this analysis will be specific for that system and many time consuming experiments shall be done to define the quality of clustering.

In this thesis both approaches will be used. An example of a statistic measure that will be employed is the mean square error (MSE). For a visual words' dictionary with k clusters and n data points used in clustering, the mean square error equals to:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{k} \sum_{x_j \in c_i} (x_j - \mu_i)^2 \tag{2.3}$$

where μ_i stands for the mean of all points $x_j \in c_i$.

As it has been observed in [8], the performance quality of the visual search system may depend on the training data, *i.e.* on the images used for dictionary creation. In general, we can state that the quality of the dictionary for a given evaluated collection is relatively high if it has been created from the evaluated reference images. It is due to the fact that visual word representation of reference images will be then most accurate. Another quality category is obtained by creating dictionaries from images depicting scenes from the same application scenarios as the evaluated collection. Finally, in the most challenging case, the dictionaries are created using images depicting general scenes, unrelated to the images from the evaluated dataset. It shall be reminded that one of the objectives of this thesis is to develop extensions of the visual search engine that minimise the dependency of search engine performance on the quality of the used dictionary.

2.3.3 Nearest Neighbour Search

The most computationally expensive part of the image matching process consists of searching for the closest matches to high-dimensional region descriptors. In many applications, the computational cost of linear search, which provides exact solution, renders this method of search infeasible. Thus, an interest in fast approximate nearest neighbour search algorithms arouse. Such algorithms, though orders of magnitude faster than exhaustive search, still provide near-optimal accuracy.

There have been hundreds of methods proposed to solve the problem of rapid approximate nearest neighbour search. Below, algorithms that became most commonly used in the context of visual search are presented. For detailed analysis and comparison of various search algorithms the reader is referred to [21].

1. **Flat k-means algorithm**: typically, the k-means algorithm is used to cluster previously extracted region descriptors into visual words. This way, flat visual word dictionary is obtained. When it is used for classifying image features, exhaustive linear search of all clusters is performed and an exact solution is obtained. Nevertheless, the computational time increases linearly with the increase of dictionary size.

   Moreover, on top of k-means algorithm imperfections that are discussed in the previous section, additional problem appears for clustering thousands of images – computational memory requirement. According to the author's experience, clustering of 3,500 images (Oxford dataset) into $k = 100,000$ visual words lasts approximately 1 month. Furthermore, the memory needed for storing 30,000 images exceeds 12 GB ($30,000 \approx 100,000,000$ SIFT descriptors $\approx$ 12GB).

2. **Hierarchical k-means tree algorithm**: having observed the problems mentioned above, Nistér and Stewénius proposed a hierarchical structure that is created through hierarchical k-means clustering scheme, also called tree structured vector quantisation [18]. Basically, on the first level of a tree, all region descriptors are clustered using k-means algorithm into a small number of groups (typical branching factor $k = 10$). On the next levels, k-means is applied iteratively with the same branching factor to each of the previously partitioned clusters. The final outcome is a hierarchical tree vocabulary with k^L visual words (leaf nodes) at the bottom of the tree, where L equals to number of tree levels. An example of hierarchical tree structure can be seen in Fig. 2.2.

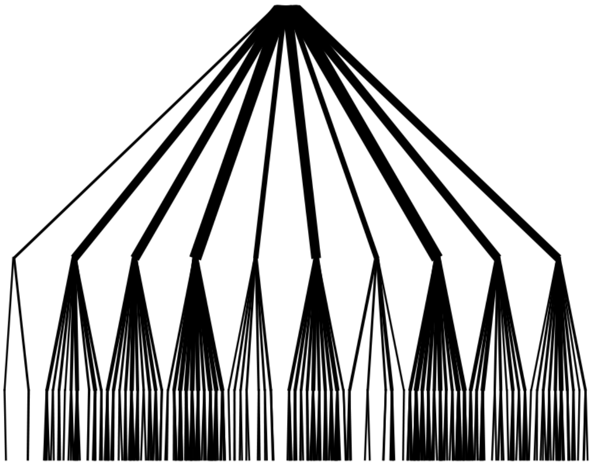


Figure 2.2: Three levels of a vocabulary tree with branching factor $k = 10$ populated to represent an image with 400 features. *Source:* [18]

When a query image is analysed, descriptors can be either compared with all the leaf nodes (full search) or, using approximate search algorithms, assigned to the nearest visual word by descending the tree. In the approximate search algorithm proposed in [6] each data point is not only assigned to a single leaf at the bottom of the tree, but additionally to some internal nodes which are passed through while descending the tree. This can help mitigate the effects of quantisation error for cases when a point lies close to the Voronoi region's[1] boundary of a cluster centre. In the extension of this solution, search is performed according to a set of priority queues of unexplored branches created during initial tree traversal [21]. The accuracy of search is then specified by limiting the number of nodes to be examined.

3. **Randomised kd-tree algorithm**: the alternative to the visual words vocabulary approach is kd-tree algorithm. Original algorithm proposed by Freidman et al. splits the data in half at each level on the dimension for which the data exhibits the greatest variance [22].

   Even though the original version performs efficiently in low dimensions, with higher dimension its performance drops rapidly. For instance, no speed-up over exhaustive search for more than 10 dimensions can be observed. Because of that limitation, essential improvement to the original algorithm was proposed by Lowe [13]. Essentially, he suggests using approximate search algorithm called Best-Bin-First. The Best-Bin-First algorithm uses a modified search ordering for the kd-tree algorithm so that bins in feature space are searched in the order of their closest distance from the query location.

   Other amendments were introduced by Silpa-Anan and Hartley [23]. They proposed a set of randomised kd trees that are built by choosing the split dimension randomly from the first D dimensions on which data has the greatest variance. In their implementation, a single priority queue is maintained across all randomised trees so that search can be ordered by increasing distance to each bin boundary. As in the case of fast search in hierarchical tree, the accuracy of search is defined through predefined number of visited nodes.

[1] Voronoi diagram is a geometrical method to partition a 2D plane with n sites into n areas such that each point in the area is closer to the areas site than to any other site. Voronoi region is the area that belongs to a particular site after partitioning.

Most recent proposal of kd-tree algorithm improvement has been done by Philbin and combines the benefits of both visual words and randomised kd-tree approaches [6]. Approximate k-means (AKM) reduces the computational time by using approximate nearest neighbour algorithm with a forest of 8 randomised k-d trees built over the cluster centres. Since in the randomised kd-tree algorithm dimensions in which the splitting is done are chosen at random, the conjunction of the trees creates an overlapping region of feature space and mitigates the effects of quantisation error. This can help to avoid situations where features fall close to cluster boundary and are classified incorrectly. It is especially significant in the high-dimensional spaces where more data points are bound to fall close to the cluster limits. Furthermore, the benefits of approximate nearest neighbour algorithm with priority queues are exploited. Initial priority queue created while traversing all the trees for the first time is used afterwards to widen area of search and choose most promising branches to search through.

2.3.4 Retrieval

After extraction and clustering of high-dimensional region descriptors, similarities between the images can be found. According to the visual word representation, image can be indexed and encoded as a set of visual word IDs or a histogram over the visual words that appear in the image representation. The complete process of encoding can be seen in Fig. 2.3.

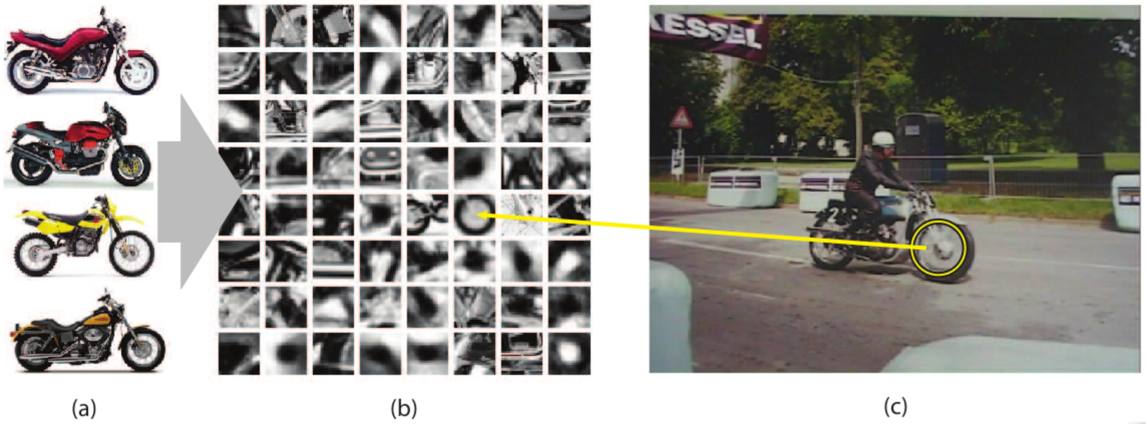


Figure 2.3: Image indexing. Features clustering (a). Each cluster represented by visual word ID (b). Image features represented by visual word IDs. *Source:* [24]

Comparing the images is done through the comparison of their visual word IDs or, alternatively, by comparing their histograms. When we want to retrieve the information about the query from the large dataset of reference images, matching consists of finding the closest visual word for each feature, instead of finding the nearest neighbour from the whole dataset. This is more efficient since the number of visual words is typically much smaller than number of features extracted from reference images.

As far as reference images ranking is concerned, most of the current solutions relay on *tf-idf* scoring model based on vector representation of images [3, 8, 18]. Typically, image is represented by a vector of visual word frequencies and the similarities between the images are verified through computation of distances between their vectors.

The weighting scheme known as "term frequency – inverse document frequency" (*tf-idf*) is computed as follows. Let k be a number of words in dictionary. Hence, every image is represented by a vector $V_d = (t_1, t_2, \ldots, t_k)^T$ of length k, where t_i stands for word frequency computed as:

$$t_i = \frac{n_{id}}{n_d} \log \frac{N}{n_i} \tag{2.4}$$

where n_{id} is the number of occurrences of word i in document d, n_d is the total number of words in the document d, n_i is the number of occurrences of term i in the whole dataset and N is the number of all documents in the dataset. Hence, two terms can be distinguished: *term frequency* $\frac{n_{id}}{n_d}$ and *inverse document frequency* $\log \frac{N}{n_i}$. First term normalised the number of word occurrences against the number of all words in a document, second reduces the weight of words that often appear in the dataset. As an output of the search, the reference images are ranked according to the score they obtain from normalised scalar product between query vector V_q and reference vector V_r.

Adapting text retrieval research contributions into the area of visual search, several additional mechanisms can be applied, *stop list mechanism* being one of them. The mechanism which eliminates key-points mapped to the most common visual words has been proposed in [3]. In general, it is an analogy of the text retrieval mechanism that discriminates the most frequent, and hence not informative, words such as *a* or *the* in English. When using a stop list in visual search engine, number of mis-matches is reduced and system reliability improves, since most common visual words that are not discriminating for a particular document are suppressed.

It will be thoroughly discussed in the next chapter where visual search system at Telefónica I+D is presented. Nevertheless, it must be emphasised that various analogies between text and video retrieval have been successfully exploited and there is certainly much more improvements that can be developed for the benefit of both fields.

2.3.5 Applications

There is a myriad of possible applications of image patch features and visual search mechanisms. In general, the feature extraction and description algorithms allow extracting the contents of an image and exploring them by machines in an intelligent way.

Below, some examples of the fields that may benefit from the methods described in this chapter are listed. It is, however, not a closed list and new applications are expected to appear in the future.

- 3D Reconstruction and Motion Tracking
- Image Segmentation
- Object and Object Class Recognition
- Image and Video Retrieval

Chapter 3

Visual Search Engine Technology in Telefónica I+D

This chapter describes the system of visual search that has been developed at Telefónica Invectigación y Desarollo Barcelona. First of all, the proprietary method of feature detection and description – DART – is analysed. Afterwards, the visual retrieval mechanism of search engine is presented and discussed in details. In other words, this chapter describes the core configuration of the system that provides a starting point for improvements proposed in this thesis.

3.1 Local Features: DART

As explained in the section 2.2, local features describing salient image patches are one of the most important elements of image matching. Points of interest (key-points) are bounded to be detected and described with high repeatability across different view conditions. However, the trade-off between the accuracy of descriptors under different conditions and their computational time exists. For the practical application, the balance between those two elements has to be found. Recently, much research in the area of local feature extraction has been focused on reducing computational power and time of existing algorithms. For instance, one of the proposed solution to speed up the computations, SURF detector and descriptor, takes advantage of integral images representation and simplicity of approximate Gaussian filters, box filters [16].

Following this research path, Telefónica I+D developed an efficient method for extracting viewpoint and illumination invariant key-points in scale-space with a variation of DAISY descriptor [25]. The algorithm, provisionally named DART, is designed towards low overall computational complexity. The reduction of time and power needed for calculating appropriate features is obtained through three main improvements:

- Piece-wise triangular filters used to approximate the determinant of the Hessian. Thanks to this contribution DART key-points are extracted faster and more accurately than in the original integral image-based implementation of SURF.
- A key-point orientation assignment that is faster then the one in SURF.
- Reusing computations done for key-point extraction in description phase and optimising the sampling space.

The overall speed-up factor obtained by the above improvements equals to 4 in respect to SURF. Furthermore, *precision-recall* curve for DART indicates that it outperforms both SIFT and SURF.

The overall DART algorithm can be divided into following phases:

1. Image filtering at different scales using triangle kernel. The outcome of this phase is reused in the next parts of the algorithm.
2. Hessian computation on each of the scales.
3. Detection of extrema.
4. Computation of dominant orientation. using the gradient information extracted in the first phase.
5. Oriented gradient calculation and DAISY-like descriptor computation.

Fig. 3.1 shows a block diagram illustrating the main phases of DART extraction.

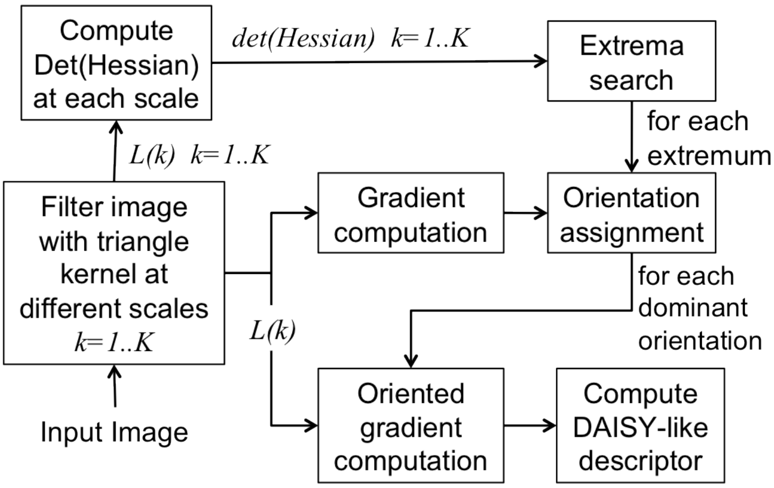

Compute Det(Hessian) at each scale
det(Hessian) k=1..K
Extrema search
for each extremum
L(k) k=1..K
Filter image with triangle kernel at different scales k=1..K
Gradient computation
Orientation assignment
for each dominant orientation
L(k)
Input Image
Oriented gradient computation
Compute DAISY-like descriptor

Figure 3.1: Block diagram of DART key-point extraction method. *Source:* [25]

The detailed description of the algorithm can be found below.

3.1.1 Extraction

As shown by Lindeberg [26] one of the possible feature detector can relay on extracting maxima of determinant of Hessian. It is computed through the following formula:

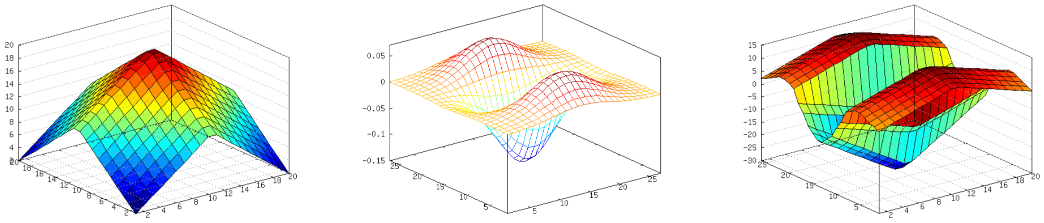
$$H_{xy} = |\partial_{xx}(i,j)\partial_{yy}(i,j) - \partial_{xy}(i,j)^2| \tag{3.1}$$

where ∂_{xx} is the second horizontal derivative of Gaussian over an image, ∂_{yy} is vertical, and ∂_{xy} is cross derivative.

The second derivatives, that are used in this matrix, give strong responses on blobs and ridges, however the functions that are based on Hessian matrix penalise very long structures for which the second derivative in one particular orientation is very small. A local maximum of the determinant indicates the presence of a blob structure. Thus, the key-point is considered localised.

In order to ease on computations, determinant of Hessian can be approximated. This idea has been successfully adopted in SURF (see section 2.2.2) where Hessian detector is approximated with simple box filters.

In case of DART, proposed extraction method begins with computation of approximated determinant of Hessian using simplified filters. Instead of Gaussian filters, image filtering is done with 2D triangle-shaped filters. The shape of this filter can be seen in Fig. 3.2(a). Using this filter, shape of the second derivative of Gaussian (see Fig. 3.2(b)) can be approximated with translated and weighted triangle-shaped responses (see Fig. 3.2(c)).



(a) 2D triangle-shaped kernel.

(b) Second derivative of Gaussian.

(c) Second derivative of Gaussian approximated using triangle responses.

FIGURE 3.2: Plots of the filter shapes used to compute the determinant of Hessian. *Source:* [25]

Let $L(k,i,j)$ be the output of triangle-shaped filter at scale k. Then, the approximations of the derivatives can be written in the following way:

$$\begin{aligned}
\partial^k_{xx} &= L(k,i-d_1,j) - 2\cdot L(k,i,j) + L(k,i+d_1,j)\\
\partial^k_{yy} &= L(k,i,j-d_1) - 2\cdot L(k,i,j) + L(k,i,j+d_1)\\
\partial^k_{xy} &= L(k,i-d_2,j-d_2) - L(k,i+d_2,j-d_2) - L(k,i-d_2,j+d_2) + L(k,i+d_2,j+d_2)
\end{aligned} \tag{3.2}$$

where $d_1 = (2\lceil 3\sigma\rceil + 1)/3$ and $d_2 = d_1/2$ are chosen empirically to best approximate the kernel of the second derivative of Gaussian generated with the corresponding σ.

The solution presented above reduces the number of accesses to $L(k,i,j)$ to 9, comparing with 32 accesses to the integral image in the case of box-shaped approximation in SURF. Furthermore, the creation of scale space is different than the one of SIFT where a pyramid of filtered versions is created. Bearing in mind the concept of reusing once computed filter responses, the scale space is created as a stack of filtered versions of $L(i,j)$ with no sub-sampling. The computational complexity of such solution does not increase dramatically, since the outputs of the previous filtering are taken as an input of the following ones.

3.1.2 Description

Similarly to SIFT and SURF descriptors, the dominant image patch orientation must be defined before extracting descriptor of the image patch. DART benefits from both approaches:

1. **SIFT approach**: dominant orientations (one or more) are found through analysis of the histogram of gradient orientations.
2. **SURF approach**: derivatives at sampled points in circular neighbourhood are obtained with Haar-wavelets. Afterwards, their magnitude is used to create vertical and horizontal derivatives space, which is then scanned and the dominant orientation is found as the largest sum of magnitude values.

In DART the sampling is done in a circular neighbourhood (as in SURF), however the gradients are computed faster simply by accessing $L(i, j)$. Then, histogram of orientations is created and multiple dominant orientations are found (as in SIFT).

After the extrema have been localised, variation of DAISY descriptor is used in order to obtain the vector defining image patch. Since DAISY descriptor [27] has been proven to outperform other widely known descriptors [28], it has been adapted to DART. The main improvement of DAISY descriptor over SIFT is a circular sampling layout that has been adapted to DART. Using this layout, derivatives oriented with respect to dominant orientation of the key-point are obtained for every sample. The oriented derivatives are used to compute a vector of four values: $\{|\partial_x| - \partial_x; |\partial_x| + \partial_x; |\partial_y| - \partial_y; |\partial_y| + \partial_y\}$. Since the optimal sampling layout consists of 8 segments and 2 rings, resulting descriptor contains $(1 + 2 \times 8) \times 4 = 68$ values for each extracted feature.

The additional speed-up is obtained through the optimisation of the sampling space. It has been shown [25] that the ring segments used in description algorithm highly overlap which leads to redundancy (see Fig. 3.3). Thus, by optimising the size of the overlap, the reduction of redundancy and time of computation can be achieved, with no significant loss of performance.

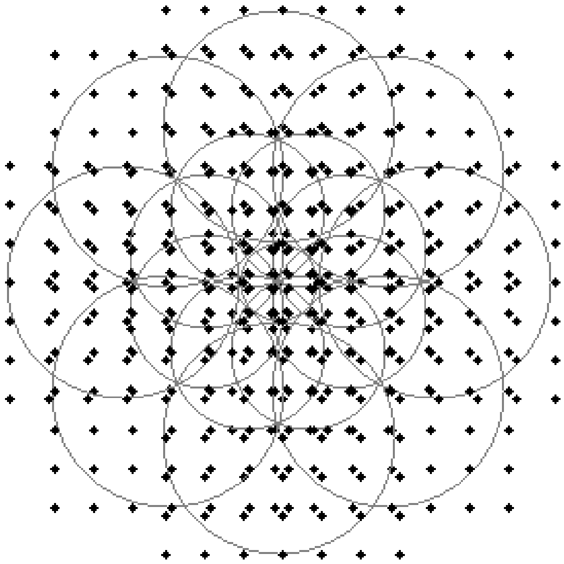


FIGURE 3.3: DART layout - adapted from DAISY. *Source:* [25]

3.2 Visual Search Engine

The visual search system developed at Telefónica I+D Barcelona addresses the problem of image retrieval. The main objective of the system is to retrieve a ranked list of images from the dataset that contain the objects present in the query image. Typically, the query image can contain various number of objects placed within a complex scene. However, it is assumed that reference images depict only one well-posed reference object on rather simple background.

The basic configuration of search engine is capable of rapid identification of objects present in the query image. The recognition process relies on comparison of local features from the query image with features from collection of reference images containing known objects. The overlaying idea behind this approach is performing local feature matching using visual word dictionary and rough spatial consistency verification in one single step.

First of all, the assumption of the well-posed single objects in reference images is exploited through a mechanism of *region of interest* (ROI) detection. In the reference images preprocessing stage the distribution of key-points is analysed and used to define the part of an image where the most relevant items are displayed – so-called *region of interest*. Obtained region borders are then used to normalise scales of the key-points lying inside ROIs and to discard key-points located outside. Thanks to that mechanism, computational time and power are saved and due to the rejection of insignificant key-points the level of background clutter is reduced.

The commonly used *tf-idf* scoring scheme, described in section 2.3.4, implies comparing multidimensional vectors that represent images. Dimensionality of the vectors are defined by number of visual words. Thus number of vector components can be as high as one million. Moreover, *tf-idf* scoring scheme does not include any spatial information about matching local features. Hence, in order to verify spatial consistency of corresponding key-points additional postprocessing stage needs to be implemented. This may introduce additional problems, such as definition of number of images to be analysed in spacial consistency verification stage.

The solution implemented in visual search engine of Telefónica I+D eliminates imperfections mentioned above by performing single step recognition. It relies on voting mechanism that provides rough spatial consistency verification of matching features. Using extension of inverted file structure proposed in [3], matches are clustered in the pose space through an operation that resembles Hough transform. The inverted file structure stores lists of hits – occurrences of every visual word in all reference images (see Fig. 3.4).

However, single hit contains not only information about the reference image ID, but also scale, orientation and *vote strength* of the key-point. Those parameters are then used in the voting process, when the best matching reference image for a particular query is found. The voting mechanism relies on a set of pose accumulators – each corresponding to a single reference image. They enable voting in the limited pose space that includes only scale and orientation information. Accumulated votes are then used to create final relevancy ranking. It has been found that the inclusion of even such a rudimentary spatial verification mechanism in the very initial stage of recognition is efficient and robust.

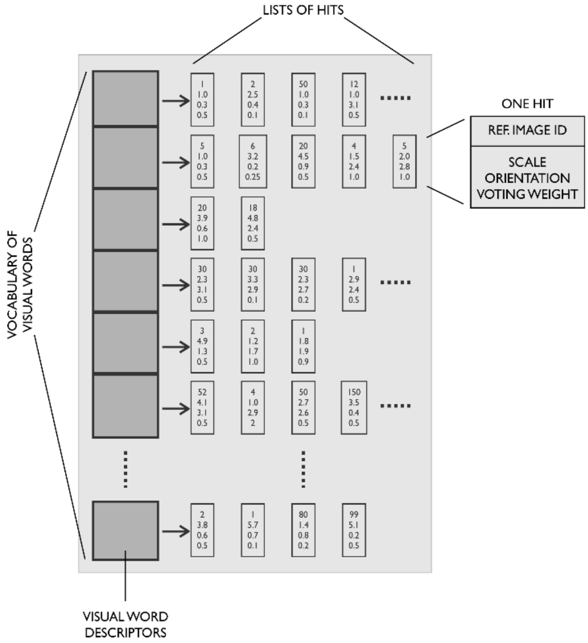


Figure 3.4: Inverted file structure that has a hit for every reference image.

3.2.1 System Overview

Benchmarking results providing information about the performance quality of the visual search engine can be found in chapter 7 where they are compared with results obtained after introduction of solutions proposed within this thesis.

Below a brief description of the whole visual search engine can be found. Fig. 3.5 shows block diagram demonstrating the dependencies between the components of the system.

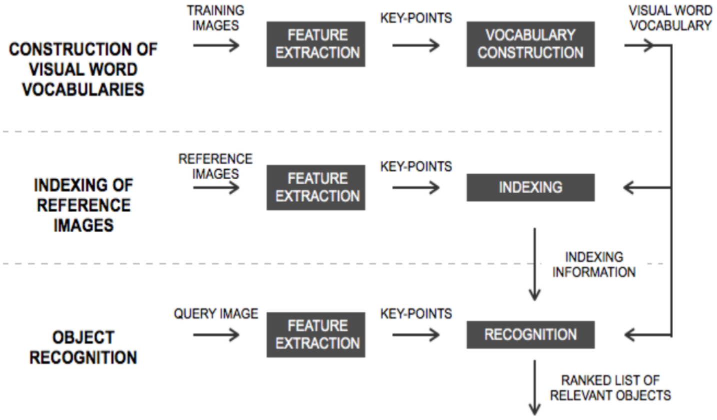


Figure 3.5: Overview of the visual search engine with the relations between the components.

The visual search engine system developed at Telefónica I+D can be divided into four distinctive modules (stages):

1. **Feature Extraction**: this phase consists of detection and description of image patch features. Typically, Telefónica proprietary solution discussed in the previous section –

DART – is used. However, the system works also for other descriptors, such as SIFT or SURF. Having extracted key-points for reference and query images, key-point post-processing can be done. This involves, for instance, detection of region of interest and rejection of less relevant key-points.

2. **Visual Word Dictionary Creation**: at this stage the clustering of feature descriptors into visual words is performed. It shall be noted that this is an off-line process which can be done absolutely independently of the system's work. Its final outcome is a visual word vocabulary that allows quantising the descriptor space (see 2.3.2). After the creation of a dictionary, the reference and query image key-points can be classified and the visual word representation is generated.

3. **Indexing of Reference Images**: in this phase feature descriptors are quantised using visual word dictionary and organised in a structure that enables fast matching. After features are extracted and their post-processing is done, the visual word representation is created through assignment of the key-points to visual words. Then, the inverted structure of hits storing reference image ID, scale and orientation data is built (see Fig. 3.4). Additionally, every hit has an associated weight with which it supports the presence of the corresponding reference object.

4. **Recognition**: at this point the final reference image scoring is performed and the visual contents of query image are retrieved. The main component of this process is voting mechanism that uses accumulators of aggregated correspondences between the query and reference image key-points. After assigning the query image features to visual words, voting weights for each key-point are computed. Those weights are then combined with the weights of reference images stored in the inverted file structure and result with the total scores of the correspondences: query image – reference image. At the end, the final scores are ordered and the irrelevant images are rejected by dynamic thresholding proposed in [29].

Below a detailed description of the above components is presented.

3.2.2 Feature Extraction

The visual search engine was designated to work with local features, *e.g.* SIFT or SURF. Detailed description along with properties of various local features can be found in section 2.2. However, in contract to other standard solutions, image retrieval system developed at Telefónica I+D performs additional post-processing of extracted features.

In principal, the system was designed to work with reference images depicting one object. Furthermore, during the development of the system it was observed that in cases of high-resolution images many of the key-points detected at the lowest scales did not represent any discriminatory patterns, but rather noise and artefacts.

The post-processing stage exploits those facts. It is aimed at selection of the most useful key-points from an image through elimination of unstable features that are likely to represent noise, *e.g.* features detected at small scales in high resolution images. Moreover, a significant reduction of computation cost can be obtained by selecting only a predefined number of the most useful key-points. Last but not least, extraction of too many features may not always be beneficial due to the noise introduced by non-discriminative descriptors.

Thus, basic idea behind post-processing of the extracted key-points is to detect those of higher scales, since they are much more discriminative than ones detected at lower scales. What is more, we locate the region where most of the useful key-points are located – so-called *region of interest* (ROI). This is done in order to exploit the assumption about the presence of one well-posed object in reference image. Key-points localised outside of ROI are discarded and those within its limits have their scales normalised with respect to the size of ROI.

Then, the key-points are sorted according to their normalised scales, key-points with the scales below predefined threshold are rejected and predefined number of key-points with the largest scales is retained. Typically, around 800 features are sufficient for reliable recognition.

It must be noted, however, that in case of query images we do not detect ROI – it is set to cover entire image. Thus, the normalisation of scales is not needed.

3.2.3 Visual Word Dictionary Creation

In order to match query image with the appropriate reference image, multi-dimensional descriptor vectors of both images are compared. If the dataset of reference images consists of many images, as in some practical application where the number of images can reach up to a few thousands, the exhaustive search is not feasible due to the enormous computational cost. As discussed in section 2.3, one of the most popular solution of this problem is quantising feature descriptors into clusters called visual words. This operation is performed using dictionaries and, as a result, every image is represented with a set of visual words. Finally, every key-point from the query image is assigned to a list of key-points from reference images that were clustered to the same visual word.

The visual word dictionaries, often called vocabularies, are created through an off-line clustering process using k-means algorithms, discussed in 2.3.2. Since the performance of the system highly depends on the quality of such dictionary, it is truly important to create them properly and using as many images as possible. This reduces search time and improves the results, because the quantisation cells are minimised. However, with the increased discriminative power the repeatability in the presence of noise may decrease.

In the solution described in this chapter we create hierarchical dictionaries proposed in [18]. In order to perform fast visual words assignment, we implemented approximate nearest neighbour search algorithm inspired by [21] and discussed in section 2.3.3.

3.2.4 Indexing of Reference Image

The overview of the process can be seen in Fig. 3.6. As it was mentioned before, the main aim of indexing of reference image is a creation of a structure that enables fast matching of a reference image with a query. The inverted file structure (see Fig. 3.4) organises key-points from reference images into lists of hits (occurrences) of every visual word. However, in our case, every hit stores also additional information about key-points scale, orientation and voting strength. That information is used afterwards in voting process. In order to build such a structure, the feature key-points need to be extracted and post-processed as it has been discussed in 3.2.2.

After post-processing, extracted key-points are mapped to visual words using visual vocabulary. This is done in order to obtain compact and efficient representation of feature descriptors.

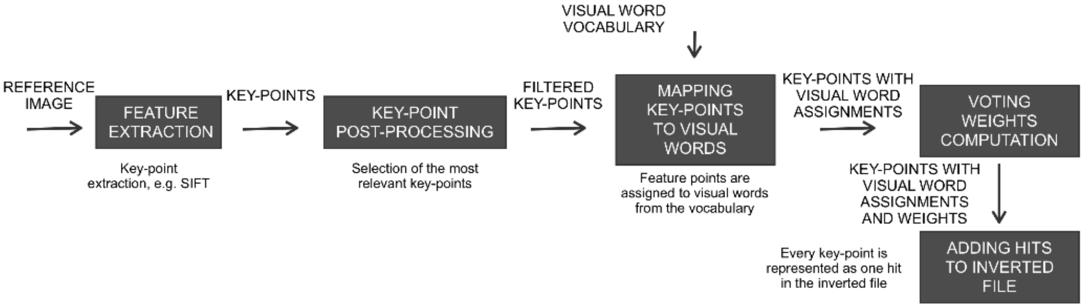

VISUAL WORD VOCABULARY
REFERENCE IMAGE
FEATURE EXTRACTION
Key-point extraction, e.g. SIFT
KEY-POINTS
KEY-POINT POST-PROCESSING
Selection of the most relevant key-points
FILTERED KEY-POINTS
MAPPING KEY-POINTS TO VISUAL WORDS
Feature points are assigned to visual words from the vocabulary
KEY-POINTS WITH VISUAL WORD ASSIGNMENTS
VOTING WEIGHTS COMPUTATION
KEY-POINTS WITH VISUAL WORD ASSIGNMENTS AND WEIGHTS
ADDING HITS TO INVERTED FILE
Every key-point is represented as one hit in the inverted file

Figure 3.6: Overview of the indexing process.

In the next step, individual weights of key-points are computed. Their value is based on two factors: the scale at which the key-point was detected and number of key-points detected in the image that are mapped to the same visual word and have the same scale and orientation.

It has been proven empirically that the key-points detected at higher scales are more relevant and provide more information that those detected at lower scales. This is rather intuitive, since the small-scale key-points describe rather irrelevant objects that are present in many different scenes, hence their discriminatory power is lower. On the other hand, key-points from higher scales correspond to larger parts of the image, hence being more informative. The above observations are reflected in the formula that is used to compute the weight factor w_S^i for key-point i detected at normalised scale s_i:

$$w_S^i = \min(s_i, T_S) \tag{3.3}$$

where T_S is an empirically chosen threshold that limits the influence of the key-points at very high scales and enables more stable recognition.

The second weighting factor reduces the influence of the numerous key-points from one image that have the same visual word, scale and orientation. Although those cases are not very common, this factor plays an important role in the voting scheme that is described in 3.2.5. Thus, the weighting factor w_M^i for i-th key-point is computed as follows:

$$w_M^i = \frac{1}{N_S^i} \tag{3.4}$$

where N_S^i stands for the number of key-points that are assigned to the same visual word as i, were detected at the same scale and have the same orientation.

The ultimate voting weight of a key-point from reference image is computed as a product of both factors: $w^i = w_S^i \cdot w_M^i$.

Finally, the inverted file structure enabling fast image matching is created. It shall be noted that the concept of inverted file structure was inspired by text retrieval systems and successfully employed in visual search [3]. However, the solution proposed at Telefónica I+D differs from the one proposed by Sivic and Zisserman. In their implementation, the inverted file structure stores a hit list for each visual word where the occurrences of that word in all reference images are stored. In the solution implemented at Telefónica I+D the hits are enriched with the information about the scale, orientation and voting strength of the key-point mapped to the particular visual word. This enables incorporating a simplified Hough transform in the voting process, that is presented below in details.

3.2.5 Recognition

The overview of the recognition can be seen in Fig. 3.7. In order to identify images relevant to the query image, the corresponding reference images need to be found. This is done through comparison of key-points clustered to visual words, as it was described in section 2.3, and a voting process in reduced pose space.

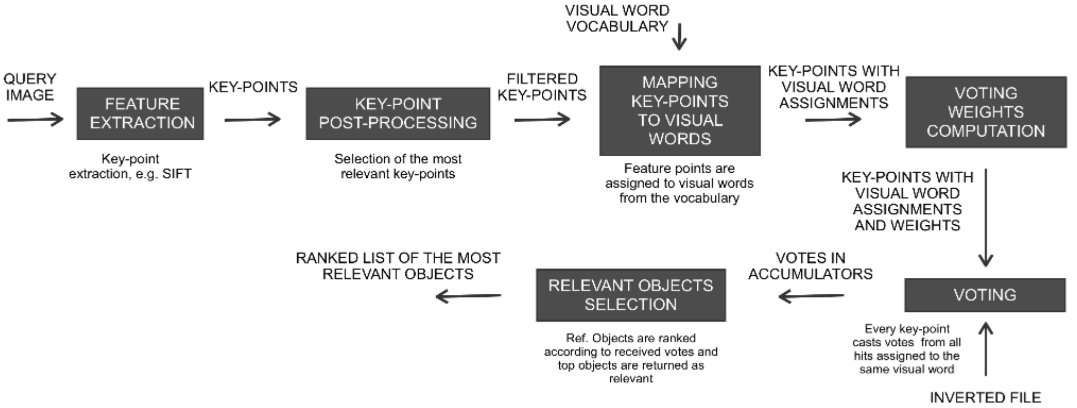

VISUAL WORD VOCABULARY
QUERY IMAGE
FEATURE EXTRACTION
Key-point extraction, e.g. SIFT
KEY-POINTS
KEY-POINT POST-PROCESSING
Selection of the most relevant key-points
FILTERED KEY-POINTS
MAPPING KEY-POINTS TO VISUAL WORDS
Feature points are assigned to visual words from the vocabulary
KEY-POINTS WITH VISUAL WORD ASSIGNMENTS
VOTING WEIGHTS COMPUTATION
KEY-POINTS WITH VISUAL WORD ASSIGNMENTS AND WEIGHTS
VOTING
Every key-point casts votes from all hits assigned to the same visual word
INVERTED FILE
VOTES IN ACCUMULATORS
RELEVANT OBJECTS SELECTION
Ref. Objects are ranked according to received votes and top objects are returned as relevant
RANKED LIST OF THE MOST RELEVANT OBJECTS

Figure 3.7: Overview of the recognition process.

The first phase of the recognition is identical to the one described above for the reference images – key-points are extracted and post-processing is done. Afterwards, they are mapped to visual words using appropriate vocabulary and the corresponding voting weights are computed. Once key-points are assigned to visual words, they are associated with the list of hits corresponding to each visual word. The weight of the query key-point is computed according to the Eq. 3.4. It shall be noted that the scale is not used for weight computation (as it is done for reference images through Eq. 3.3), since the goal of the search engine is to recognise the objects present in the query image independently of their size.

As it was proposed in [3], the mechanism of eliminating key-points mapped to the most common visual words was implemented. In general, the *stop list* mechanism, discussed in section 2.3.4, discriminates the most frequent visual words. In the visual search engine implementation, the frequencies of visual words appearance are computed in the reference image indexing phase. Afterwards, the query key-points mapped to the subset of the most frequent visual words (typically 1% of the visual word vocabulary) are disregarded in the recognition phase.

The next stage of recognition is votes aggregation. Votes for each reference image are stored in the specific structure called accumulator. Since the votes coming to the accumulators are cast according to the scale and orientation of the “voting” key-points, the whole process can be seen as a simplified version of Hough transform. In other words, every pair (key-point and hit describing the occurrence of a corresponding key-point in reference image) votes for a presence of a particular reference object with a specified scale and rotation. The strength of the vote is computed as a product of voting strength of reference and query image key-points. As soon as the voting process ends, the pose accumulators are scanned in order to identify bins with the highest number of votes. Votes from those bins are then taken as a final score and used to compose the relevancy list of reference images. Finally, the most relevant images are selected through dynamic thresholding described in [29].

3.3 Limitations

The technology of visual search developed at Telefónica I+D provides efficient and rapid image content retrieval. However, a certain limitations of the current system set-up have to mentioned.

Firstly, it has been reported that the visual search engine is sensitive to changes of the quality of visual vocabulary used. This comes as no surprise, since similar behaviour of image retrieval systems can be found in literature [8, 18]. In general, system performs well with dictionaries built from reference images and when number of visual words in vocabulary is similar to number of key-points used for its creation. Nonetheless, with a change of dictionary quality performance can drop severely. Thus, in this thesis we will address the problem of system robustness under changing conditions of visual vocabulary quality.

Moreover, it shall be noted that the current configuration of search engine performs only partial and simplified spatial consistency verification. Since voting is done in the limited pose space, additional spatial information, such as key-point coordinates, are not accounted for in recognition process. This may be particularly significant in outdoor recognition where repetitive patterns are present and increase of discriminatory power is highly desired. Hence, in this thesis we will attempt to fully exploit spatial information using re-ranking mechanism. That shall lead to increase of engine's discriminatory power and overall performance improvement.

3.4 Applications

In general, the visual search engine designed and implemented at Telefónica I+D provides a fast and reliable recognition tool to deliver results in response to photos instead of text. Although this is enabling technology and could be employed in various applications, special care has been taken to tackle several most promising application areas.

Outdoor Augmented Reality

The visual search engine combined with GPS and compass device can be easily deployed in any mobile terminal as a high-precision geo-location service. By comparing photos captured by camera with geo-tagged reference data, it would enable user not only to find his position, but also obtain information about the surrounding real world attractions, landmarks and other places. User would take a picture, holding up his mobile device, and the search engine would easily retrieve the information about the place of interest. Then, it would display the directions on the screen, leading a user towards a desired location.

One of the main aims of this thesis aims is improving the state of the art visual search engine technology developed at Telefónica I+D so it can be applied in geo-localisation services. Additional incentive to adapt this technology to outdoor localisation comes from collaboration within MobiAR project[1] funded by Spanish Government. Main objective of this project is development of augmented reality based touristic services that will be available for mobile devices.

[1] http://mobiar.tid.es/

Visual Purchasing

Since the proposed solution provides enabling technology for various visual recognition system, possible applications may include buying audiovisual contents by taking pictures of their ads in magazines, or purchasing tickets for a music concert by simply taking a photo of a city poster.

Advertisement monitoring

The solution described above could also find a potential application in automated monitoring of commercial campaigns across various types of media (television, Internet, newspapers, *etc.*). For instance, visual search engine implemented in a monitoring tool could automatically search through Internet and analyse all the occurrences of trademarks or logos of a particular company. That could be used to value the services of a marketing company hired to prepare the campaign.

Chapter 4

Evaluation of System Performance

This chapter discusses the importance of exhaustive evaluation in the development of visual search engine and difficulties that correspond to this evaluation. Then, evaluation measures that are commonly used to evaluate image recognition systems are described in detail. Their advantages, potential flaws and implementation difficulties are discussed. Finally, this chapter provides the description of all dataset gathered and adapted to be used within this thesis.

4.1 Problem Statement

Evaluation framework is essential to analyse performance of any complex system, visual search engine being one example. Meaningful and precise assessment is crucial for system development and optimisation. However, exhaustive evaluation of complex systems is not a trivial task.

In order to evaluate any information retrieval system several problems have to be accounted for. First of all, due to system complexity, it is difficult to analyse the behaviour of system modules separately. Furthermore, assessment of user experience quality is not easy, since perception of retrieved results may differ highly not only amongst application scenarios but also amongst different people. Moreover, meaningful and statistically relevant results require using large collections. Currently, there is only a few data sets proposed in the literature and even fewer are publicly available. Therefore, gathering essential test collections becomes a challenging task. Furthermore, in order to analyse the performance of visual retrieval system, the images from the collections must be accompanied with the information about the correspondences between query and reference images – the so-called *ground truth.* Typically, every collection comes with ground truth structured and formulated differently. Last but not least, until now every visual recognition system has been evaluated using different collection and different evaluation measures. The main reason for this situation is the cumbersome adaptation of each evaluation procedure to a very specific application scenario.

4.2 Proposed Solution

Because of the problems mentioned above, we have decided to implement an assessment tool capable of computing a complete set of evaluation measures. We incorporate most of those that were proposed in the literature. The final set of the measures contains those approved by the information retrieval community [7] and also measures specific to the visual recognition field, *e.g.* Top-X curve [18].

Furthermore, we develop a ground truth representation format that is able to unify all different ground truth formats. The ground truth information that comes with the various collections is represented in this common format. One of the challenges that we face when designing unified ground truth format is is the fact that for the images of customer goods, *e.g.* Coca–Cola can, concept of relevant reference image can be extended to all the objects with Coca–Cola logotype. Thus, in proposed ground truth representation, we introduce the concept of *Perfect Match* reference image. A reference image can be defined as Perfect Match for query image if it depicts *exactly* the same object as the query. Hence, a set of the Perfect Match reference images is a group of the images that depict exactly the same object and it is a subset of all relevant images which may include objects that are *e.g.* branded with the same logotype. Following the Coca–Cola can example, a Perfect Match for the query image depicting Coca–Cola can is a reference image that shows the same can. The relevant images for this query can include Coca–Cola bottle or Coca–Cola Light can.

4.3 Implementation

In this section, we will discuss in detail the evaluation measures that have been implemented. Once the evaluation framework was implemented, various stress and functional tests have been carried out in order to verify the correctness of the proposed solution. The references to the definitions of the presented measures will be done frequently throughout this thesis.

4.3.1 Evaluation Measures

Below, description of the implemented measures can be found.

Precision and Recall

Precision and *recall* are widely used statistical classifications. Precision can be described as fidelity of measures, and *recall* as a measure of completeness.

$$\text{Precision} = \frac{\text{correct matches}}{\text{correct matches} + \text{false matches}} \quad \text{Recall} = \frac{\text{correct matches}}{\text{correspondences}}$$

Values of precision and recall vary from 0 (no correct matches retrieved) to 1 (precision: only correct matches retrieved, recall: all relevant matches retrieved).

Mean Average Precision (MAP)

Mean Average Precision is measure proposed within the information retrieval community and one of the most commonly used in the literature [6, 8]. It provides a single-figure measure of quality across recall levels. Among evaluation measures, MAP has been shown to have especially good discrimination and stability.

For a single information need, Average Precision is the average of the precision value obtained for the set of top k documents existing after each relevant document is retrieved, and this value is then averaged over information needs [7]. That is, if the set of relevant documents for an information need $q_j \in Q$ is $\{d_1, \ldots d_{m_j}\}$ and R_{jk} is the set of ranked retrieval results from the top result until you get to document d_k, then

$$\text{MAP}(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} \text{Precision}(R_{jk})$$

If the relevant document (or image) is not retrieved at all the precision value in the above equation equals 0. For a single information need, the average precision approximates the area under the interpolated precision-recall curve, and so the MAP is roughly the average area under the precision-recall curve for a set of queries. Because of the above set-up, while using MAP, there is no need to approximate recall levels. This becomes significant with the collections where the number of relevant images per query image differs.

When working with test collections, Mean Average Precision is the arithmetic mean of average precision values for all queries. However, it must be noted that since the precision depends highly on the difficulty of the query image, diverse and representative selection of queries shall be chosen to assess system effectiveness.

Measures regarding Perfect Match

In order to analyse the distribution of reference images defined as Perfect Match in the list of retrieved reference images, several measures have been implemented. 1^s *Position Perfect Match* describes the number of queries (and the corresponding percentage) for which any Perfect Match reference image was retrieved at the first position on the retrieved list. *Top-5 Perfect Match* describes the number of queries (and the corresponding percentage) for which any Perfect Match reference image was retrieved within first five top-ranked images. *Out of Top-5 Perfect Match* defines number of queries (and the corresponding percentage) for which no Perfect Match reference images were retrieved within five top-ranked images. The percentages obtained for *Top-5 Perfect Match* and *Out of Top-5 Perfect Match* always add up to 100%. The ideal system obtains 100% for 1^s *Position Perfect Match* and *Top-5 Perfect Match* which means that for all the queries in the evaluated dataset the image retrieved on the first position was the Perfect Match reference image.

Additional measure describing the characteristics of the Perfect Match reference image is average 1^{st} *Perfect Match to First irrelevant image scores' ratio.* It is computed as an average of the ratios between the final score of the top ranked Perfect match reference image and the score of the top ranked irrelevant image. If the score of the irrelevant image is equal to zero, in order to avoid division by zero, we approximate that score with 0.001. This measure tries to capture the discriminatory power of the system. Hence, ideally, the value of this measure should be as high as possible. More precisely, the higher the average score ratio, the easier it would be to distinguish the relevant images from the irrelevant ones easily.

Another implemented measure that takes into account the distribution of reference images defined as Perfect Matches is *Average position of* 1^st *Perfect Match.* It is computed as the average value of the positions of top ranked Perfect Match reference images on the retrieved images' list for all the queries. In the case of ideal system, it should be equal to 1, meaning that for all the queries the reference image retrieved on the first position was the Perfect Match.

The abovementioned measures are intuitive, but due to the specific definition of Perfect Match reference image they may lead to confusion. Furthermore, they do not take into account the relevant images that are not the Perfect Match, but should be displayed at the top of the retried images' list due to the relevancy for the query.

Percentage of queries where irrelevant image was returned before first relevant

It is a measure describing distribution of false positives. It is essential addition to the previous measures, since ideally information retrieval system shall never position irrelevant image before relevant one.

Average number of reference images in the Top-X

This measure defines how many images out of X relevant ones are retrieved within X top ranked reference images. This measure is especially valuable for the collections with the constant number of relevant images per query (*e.g.* UK-Bench dataset [18]). It describes how many reference images makes it to the top of the retrieved reference images. For instance, if there are 4 relevant images for the particular query and the average number of reference images in Top-4 equals to 3, it means that on average out of four first images returned by the visual search engine, 3 are the relevant ones. However, when the queries in a dataset do not have the same number of relevant images, this measure might not be the most representative quantity.

Precision – recall curve

Precision – recall curve is based on the number of correct matches and the number of false matches obtained for an image pair. For the given recall that is computed as it was explained above we calculate the precision. We use the values obtained this way as the x and y coordinates displayed on the graph where in x-axis we have the recall and in y-axis – the precision. An ideal precision – recall curve has precision 1 over all recall levels, which corresponds to an Average Precision of 1. More precisely, the ideal system obtains the precision that equals to 1 for the recall equal to 1. This means that all relevant images are retrieved with no irrelevant images before them.

The example of a precision – recall curve can be seen in Fig. 4.1. For instance, let $n = 5$ be the number of relevant images in a query and assume that there is only one query in our data set (so no averaging over queries is needed). If for given recall 0.2 we obtain precision 1 that means that one relevant image out of 5 ($recall = 1/5 = 0.2$) was retrieved on first position: $precision = 1$ correct match/(1 correct match + 0 incorrect matches) = 1. Moreover, if for given recall 0.8 we obtain precision 0.5 that means that 4 relevant images out of 5 ($recall = 4/5 = 0.8$) were obtained within 8 first positions: $precision =$ (4 correct matches/(4 correct matches + 4 incorrect matches) = 4 correct matches/8 positions = 0.5.

Since while drawing *precision – recall curve* we assume fixed recall levels for all queries, for the collections with different number of relevant images per query quantisation of the recall values is needed.

It shall be noted that the precision – recall curve is commonly used in literature, *e.g.* in [8].

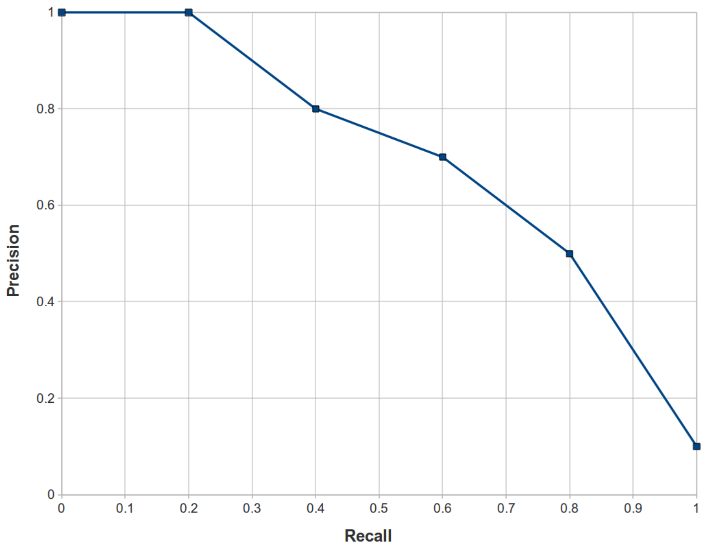


FIGURE 4.1: Precision-Recall curve

Top-X curve

Top-X curve shows percentage (y-axis) of the ground truth query images whose Perfect Match reference image makes it into the top X percent (x-axis) frames. If we assume that an ideal system would retrieve all the Perfect Match reference images on the first position, the Top-X curve would obtain value of 100% for all values of x of the x-axis. The example can be seen in Fig. 4.2. This curve was introduced in [18] and is a complimentary measure for the Perfect Match numerical statistics described above.

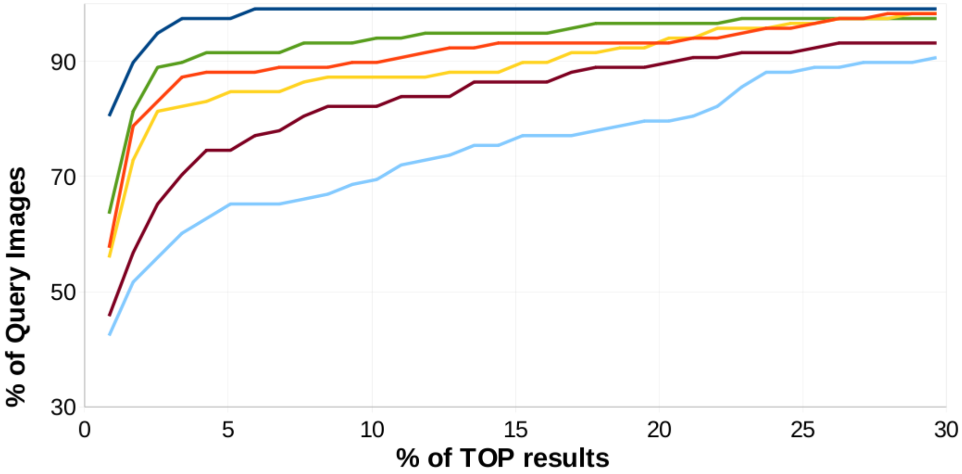


FIGURE 4.2: Top-X curve

4.4 Datasets

In order to perform the exhaustive evaluation of any visual search system a set of tagged images must be used. Furthermore, since all the measures discussed in the previous section are based on the *a priori* knowledge of the correspondences between query and reference images, an appropriately formatted information about those correspondences – the *ground truth* – must be provided. Because of the fact that in most cases the format of provided ground truth differed significantly from the format accepted by the evaluation tool, a set of tools has been developed to unify all those formats.

As mentioned in chapter 1, this work is mainly focused on the outdoor image recognition and visual search application for high precision geo-localisation services. Thus, we have gathered several datasets representing this particular type of scenes. Nevertheless, we also wanted to verify if the extensions introduced in this work are specific only for the outdoor collections or if they are more generic. Brief description of all the data sets used for the sake of this thesis can be found below. Sample images from each of the data sets can be found in appendix A.

The Oxford Dataset

The Oxford Buildings Dataset[1] consists of 5062 outdoor landmark images collected from Flickr by searching for particular Oxford landmarks [6]. The resolution of the images is 1024 × 768. The collection has been manually annotated to generate a comprehensive ground truth for 11 different landmarks, each represented by 5 possible queries. This gives a set of 55 queries over which an object retrieval system can be evaluated. Queries are created through cropping reference images according to the instructions provided by authors.

Since the data set provides several queries of variable difficulty level, this set is frequently used within the development of this thesis. The ground truth provided with the dataset is unified in the following way: images labelled as *good* (clear picture of the landmark) and *OK* (more than 25% of the object is clearly visible) are defined as the Perfect Match reference images for the landmark. Sample query and reference images are shown in Fig. A.1.

The ZuBuD Dataset

The ZuBuD Image Dataset[2] consists of 1005 outdoor images of 201 Zurich buildings [30]. Set of 115 query images along with ground truth documentation is provided with the collection. The resolution of the reference images is 640 × 480 and the resolution of the query images is 320 × 240.

It should be noted that the relative difficulty of the data set is rather low – query images are well centred and in focus. Furthermore, they depict entire building with no occlusions nor clutter. Sample query and reference images are shown in Fig. A.2.

[1] http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/index.html

[2] http://www.vision.ee.ethz.ch/showroom/zubud/index.en.html

The Nokia Dataset

The Nokia Dataset contains over 2500 geo-tagged outdoor images from Stanford University and its neighbourhoods [1]. It has been acquired over multiple shooting sessions by several individuals which guarantees a diversity of view, lighting conditions, occlusions and camera characteristics. Several types of scenes can be seen in the pictures: shopping malls, street views, university building, *etc.* Furthermore, reference images have been captured by high resolution cameras (resolution: 2048 × 1536) and query images by mobile phone cameras (resolution: 640 × 480). Many of the query images are out of focus and taken in poor light conditions.

While developing this thesis, the Nokia Dataset is used without the geo-location information. Therefore, the resulting evaluation conditions are much more challenging than those presented in [1]. Thus, the difficulty level of the Nokia Dataset is very high. Sample query and reference images are shown in Fig. A.3.

The Paris Dataset

The Paris Dataset [3] contains 6412 high resolution (1024 × 768) images obtained from Flickr by querying the associated text tags for famous Paris landmarks such as “Paris Eiffel Tower” or “Paris Triomphe” [6, 8]. In this thesis Paris collection is used only for dictionary creation. Sample query and reference images are shown in Fig. A.4.

The Iris Dataset

The Iris Dataset consists of 53 reference images of various resolutions and 117 query images. They depict various consumer good products such as Coca-Cola cans, bottles of water, chocolate bars, *etc.* It has been gathered to be used for development of mobile visual search applications that offer services related to the consumer good products. The images have been acquired under varying lighting conditions with many images containing more than one object. Manually defined ground truth is provided with the dataset.

Additional set of 206 images with indoor objects of the same type has been gathered in order to provide data set representing scenes typical for the same application scenario as the Iris Dataset. In the remainder of this thesis we will call it *Iris Others* data set and it will be used for creation of dictionaries only.

Iris Dataset is a rather challenging collection – some of the query images are out of focus and not well centred. However, it is relatively small and thus, presents rather medium level of difficulty, when measuring the discriminative power of the retrieval solutions. Sample query and reference images are shown in Fig. A.5.

The University of Kentucky Dataset

The University of Kentucky Dataset[4] (also referred to as *UK-Bench*) contains over 10 000 images of resolution 640 × 480 [18]. All the images have been captured indoors under good lighting

[3] http://www.robots.ox.ac.uk/~vgg/data/parisbuildings/

[4] http://vis.uky.edu/~stewe/ukbench/

conditions. All of them are well centred and correctly focused. Every object is represented by four pictures.

Since the number of pictures in the collection is immense, the difficulty level between queries varies significantly, *e.g.* CD covers are much easier than flowers. Sample query and reference images are shown in Fig. A.6.

The Caltech 101 Dataset

The Caltech 101 Dataset[5] contains 9146 images, split between 101 distinct object categories (such as faces, planes, ants, *etc.*) and one background category. Number of images varies from 40 to 800 per category. The resolution of images is roughly 300×200. Since it has been gathered to facilitate computer vision research and techniques for object class recognition, it cannot be used for visual retrieval of images representing specific objects. Thus, within this thesis, it will be only used to create generic dictionaries. Sample images are shown in Fig. A.7.

4.5 Dictionaries

As discussed in section 2.3.2, the quality of system performance depends on the visual word vocabulary. More precisely, it depends on the training data. In the case of the visual dictionaries, the training data consists of the images that are used to create the vocabulary. One of the objectives of this thesis is to develop extensions of the visual search engine that minimise the dependency of performance on the quality of the used dictionary. Thus, within the development of this thesis we evaluate several different configurations of datasets and employed dictionaries.

Dictionaries used in this thesis are created from reference images of collections described above. In the remainder of this thesis we will name the dictionaries created from reference images of particular data set after this data set. For instance, dictionary created from reference images of Iris collection will be called Iris dictionary, dictionary created from reference images of Oxford data set will be called Oxford dictionary, *etc.*

Furthermore, in order to compare results obtained with different dictionaries, all the created vocabularies will contain 100 000 visual words.

[5] http://www.vision.caltech.edu/Image_Datasets/Caltech101/

Chapter 5

Soft Assignment

This chapter discusses the soft assignment mechanism proposed as a solution for alleviating quantization effects in visual search engine relying on visual word dictionaries. Specifically, it describes an adaptation of the soft assignment proposed in the literature for tf-idf *approaches to the image retrieval system developed at Telefónica I+D. Since several extensions and configurations are implemented and evaluated, this chapter includes the implementation details and results of the experiments performed to optimize the mechanism. Finally, the influence of the soft assignment on the system performance is presented and the final conclusions are drawn.*

5.1 Problem Statement

Representing images with sets of key-points mapped to *visual words*, as it is described in section 2.3, permits rapid computation of similarity between two images through comparison of their visual word representations. Since similarity between key-points is found through comparison of visual words' co-occurrences, two key-points are considered identical if they are assigned to the same visual word. The degree of resemblance between the images is then computed taking into account the number of such pairs of identical key-points.

However, one should note that if two key-points are not assigned to the same visual word, they are considered completely different. In other words, the distance between them is infinite, since the mapping of key-points to visual words in only a coarse approximation of their location in the descriptor space.

The process of assigning one key-point to a single visual word is often called *hard assignment*. It provides a compact representation of an image patch feature. Since it replaces descriptor components with visual word identifiers, simultaneously, a significant amount of quantisation noise may be introduced into a recognition system.

As it was discussed in section 2.2, important property of local feature descriptor is its invariance to various conditions' changes: illumination, rotation, scale, *etc.* Nonetheless, it is common that the descriptor components can be affected to a certain degree by conditions' variations. Furthermore, the additional fluctuations of descriptors may arise from image noise, variability of detection process and difficult pose changes. Therefore, we can easily imagine situations where two descriptors representing the same image patch will be assigned to two different visual words in the process of hard assignment. Despite of the fact that the distance between them might be relatively small, those two visual words will be considered different.

Furthermore, since key-point descriptors rarely lie close to visual words, the task of capturing similarity between key-points using their vicinity to visual words is indeed very challenging. In other words, reliable and consistent capturing of similarities between key-points represented with visual words is not a trivial problem.

The above considerations clearly show that the hard assignment can easily lead to severe performance loss. Since often the correct matches between query and reference image are not found, the number of retrieved relevant images decreases, and so does the recall. The performance suffers even more when dictionaries of low quality (see section 2.3.2) are employed. In this case, quantisation effects become more apparent and even the lowest descriptor fluctuations can result in mis-match.

5.2 Proposed Solution

It has been shown in [8] that many limitations of hard assignment can be solved through weighted combination of visual words that represents an image patch. The mechanism that assigns a set of visual words to single image patch is commonly referred to as *soft assignment* (in contrast to hard assignment described in the previous section). It originates from histogram comparison technique that is used to “smooth” a histogram through spreading the count of one bin among neighbouring ones [31].

In general, the main objective of soft assignment is to capture similarities between the descriptors in a more reliable way, independently from the external image distortions and quantisation effects.

The benefits of soft assignment can be seen in the examples shown in Fig. 5.1 that were originally presented in [8]. As we can see, there are more meaningful matches found when using soft assignment.

As discussed in [8], there are two main approaches to the problem of soft assignment in visual search. In the first approach, defined as *descriptor-space soft assignment*, the descriptor is extracted from an image patch and then assigned to a set of closest visual words. According to the second approach, several deformations are applied on the image patch, generating set of synthesised patches. Then, one key-point is extracted from each of them. Afterwards, obtained key-points are assigned to one visual word each. In this thesis, first approach will be analysed, since its computational cost is relatively lower and it has been shown in [8] that it performs significantly better.

The starting point for the work presented in this thesis is the solution proposed by J. Philbin in [8]. Since it has been successfully implemented for standard *tf-idf* approach, we will adapt it to the needs of visual search engine developed at Telefónica I+D. Although both systems systems rely on the visual word representation, the structure of the visual content retrieval system of Telefónica I+D is radically different from the one used in [8]. Thus, the adaptation is not a trivial task and will involve designing additional elements, such as weights fusion scheme. Furthermore, several novel experimental set-ups will be introduced and implemented to analyse the mechanism of the soft assignment. Finally, several variants and configurations of soft assignment will be thoroughly tested and discussed.

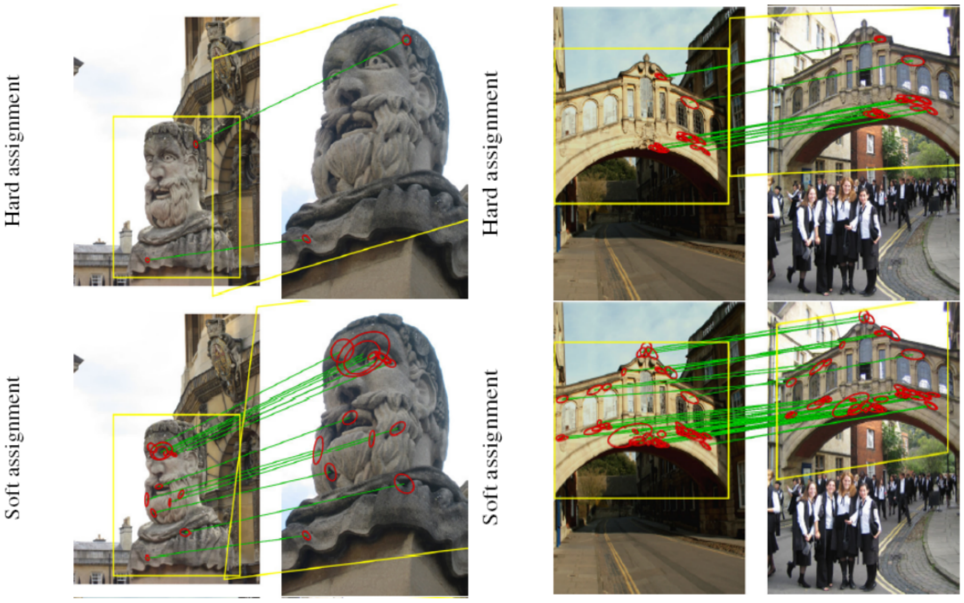


Figure 5.1: Benefits of soft assignment – comparison of identified correspondences between the images for hard and soft assignment. *Source:* [8]

Nevertheless, it must be noted that soft assignment implies higher computational complexity than hard assignment, since the search for nearest neighbour is extended to multiple nearest neighbours. What is more, the further stages of search engine will have to tackle the problem of higher number of tentative correspondences between two images which may increase computational cost even further. Thus, we propose and test additional thresholding mechanisms that aim at reduction of computational cost.

5.2.1 Weights

In [8], the weights used to create a combined visual word representation of image patch are computed according to the distance between the descriptor and the centre of the neighbouring clusters. The objective is to preserve similarities between the key-points even after quantisation.

The benefits of such an approach can be seen in Fig. 5.2, where points A-E represent the centres of clusters and 1-4 – the descriptors. In case of hard assignment, descriptors 1 and 2 will be considered identical and equally distant from A, whereas 3 and 4 will be assigned to two different visual words and treated as different. On the other hand, soft assignment provides a way to represent the notion of distances through weights – higher weight is associated with the closer visual word, lower with the further. Thus, descriptors 3 and 4 will be assigned to A, B and C with certain weights, preserving the information about the relative distances. This way, they can be matched in the further stages of the recognition. It should be noted that this mechanism also allows to capture difference in distance to A between 1 and 2. In general, the soft assignment provides a mechanism to preserve the information that would be lost in case of hard assignment.

Typically in soft assignment, *e.g.* in estimating Gaussian Mixture Models [31], weight assigned to a visual word is an exponential function of the distance to the cluster centre. In [8] associated weight is computed according to the following formula:

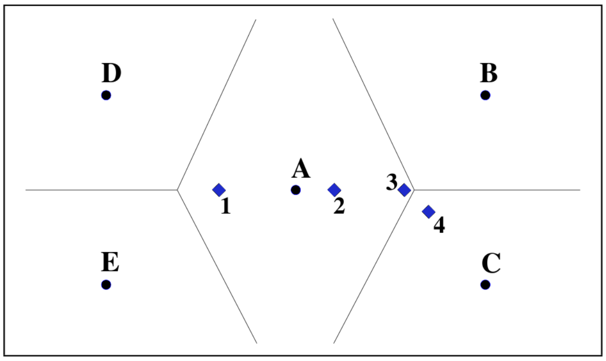


FIGURE 5.2: Quantisation effects for hard assignment. *Source:* [8]

$$w_k^i = \exp\left(-\frac{d_k^2}{2\sigma^2}\right) \tag{5.1}$$

where d_k is the distance from the i-th descriptor to the centre of cluster k, and σ is a parameter of exponential function that defines weight assigned to visual words. The distances d_k are computed as square Euclidean distances.

In the soft assignment for r nearest visual words considered, the descriptor can be represented with corresponding r-dimensional vector of weights. It shall be noted that typically after computing the values of the vector components, L_1 normalisation is performed, so the sum of the weights describing one key-point equals to one [8]. By doing so, potential problems, that may arise from differences in distribution of cluster centres in descriptor space, can be solved. In other words, this solution guarantees that the total weight of visual words associated with one descriptor will equal to 1. This also guarantees that every key-point has the same contribution to the recognition process, independently on the density of visual words in the descriptor space region where it falls.

The adaptation of the soft assignment to *tf-idf* scoring scheme proposed by [8] is relatively simple. Since this standard weighting scheme is generally applied to integer counts of visual words in images, some modification have been proposed to handle r-dimensional descriptor representation. Philbin proposes to use normalised weights' values in term frequency and integer counts of occurrences of non-zero weighted visual words in inverse document measure.

In the case of Telefónica I+D visual search engine, the fusion of weights is much more complex problem due to the employed voting scheme, where the voting strengths of matches are combined in spatial pose accumulators (see section 3.2.5). Thus, we propose to fuse the weights of the query image key-point and the corresponding reference key-point as follows. Let i, j be key-points indices, θ_k^i the identification index of k-th visual word in r_i-dimensional vector representation of key-point i and w_k^i is a weight assigned to this visual word. Then, the similarity measure of the pair of key-points i and j can be computed as a sum of squared dot products of corresponding visual words' weights according to the following formula:

$$s(i,j) = \sum_{k=1}^{r_i} \sum_{l=1}^{r_j} \pi_{kl}^{ij} \quad \text{where} \quad \pi_{kl}^{ij} = \begin{cases} \sqrt{w_k^i \cdot w_l^j} & \text{if } \theta_k^i = \theta_l^j \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

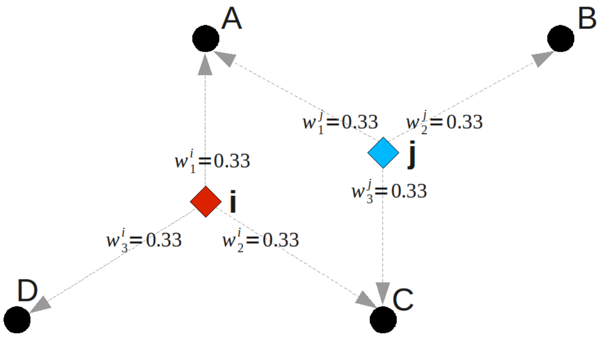


FIGURE 5.3: Example of Similarity Measure Computation

In general, we expect that the similarity values of descriptors that are close to each other to be higher than those of descriptors that lie further away. For instance, if two key-points are represented with an identical single visual word, then the weights assigned to this visual word equal to 1 for each key-point. Thus, similarity measure between those key-points is equal to $\sqrt{1 \cdot 1} = 1$ – they are identical. If each key-point is represented with single visual word, but the identification indices of those words do not match, then the similarity measure equals to 0, meaning they are different. An example of more complex case is shown in Fig. 5.3. We compute similarity measure between key-points i and j as follows. Let $\theta^{\mathbf{i}} = (A \quad C \quad D)$ be the visual word representation of key-point i with corresponding weights $\mathbf{w^i} = (0.33 \quad 0.33 \quad 0.33)$. By analogy, for key-point j we have $\theta^{\mathbf{j}} = (A \quad B \quad C)$ and $\mathbf{w^j} = (0.33 \quad 0.33 \quad 0.33)$.

Hence, we compute $s(i,j)$ in the following way:

$$\begin{aligned} s(i,j) &= \sum_{k=1}^{r_i}\sum_{l=1}^{r_j}\pi^{ij}_{kl} = \sum_{k=1}^{r_i}\left(\pi^{ij}_{k1}+\pi^{ij}_{k2}+\pi^{ij}_{k3}\right) = \left(\pi^{ij}_{11}+\pi^{ij}_{12}+\pi^{ij}_{13}\right)+\left(\pi^{ij}_{21}+\pi^{ij}_{22}+\pi^{ij}_{23}\right)+ \\ &+ \left(\pi^{ij}_{31}+\pi^{ij}_{32}+\pi^{ij}_{33}\right) = \sqrt{w^i_1 \cdot w^j_1}+\sqrt{w^i_2 \cdot w^j_3} = \sqrt{0.33 \cdot 0.33}+\sqrt{0.33 \cdot 0.33} = 0.66 \end{aligned} \tag{5.3}$$

5.2.2 Additional Configurations

Since the solution proposed in [8] was only a starting point for this thesis, we have designed, implemented and tested additional mechanisms and techniques in order to improve the soft assignment mechanism.

As mentioned before, the soft assignment implies higher computational complexity than hard assignment. This is due to the increase of potential correspondences between the images. Furthermore, with increasing number of matches additional noise is introduced to the system. Thus, we propose to implement thresholding mechanisms aimed at rejection of the visual words that do not provide relevant information since they are too distant from the key-point (distance based thresholding) or have too small weight (weight based thresholding). Moreover, an alternative method of weight computation will be discussed.

The overview of the adapted solution with additional mechanisms can be seen on the block diagram in Fig. 5.4.

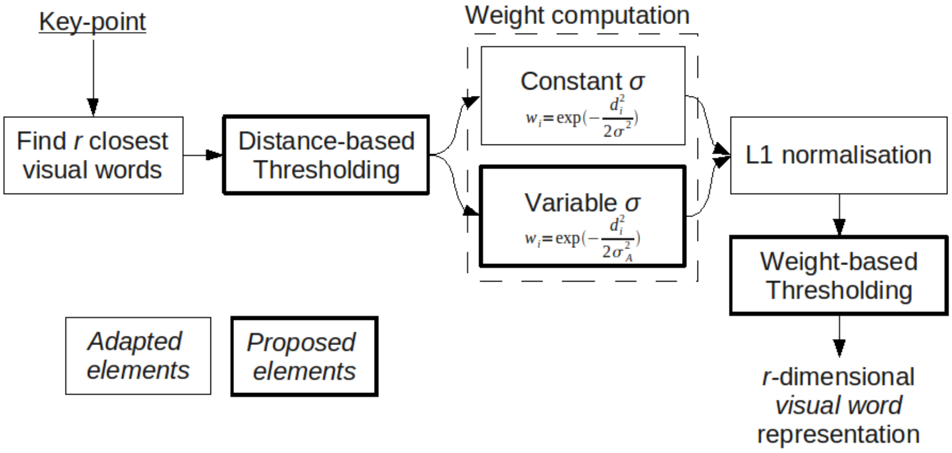

Key-point
Find *r* closest visual words
Distance-based Thresholding
Weight computation
Constant σ
$w_i = \exp(-\frac{d_i^2}{2\sigma^2})$
Variable σ
$w_i = \exp(-\frac{d_i^2}{2\sigma_A^2})$
L1 normalisation
Weight-based Thresholding
r-dimensional *visual word* representation
Adapted elements
Proposed elements

FIGURE 5.4: Block diagram describing soft assignment mechanism with additional configurations included.

First of all, in order to facilitate the soft assignment process, we propose a pre-processing stage – distance based thresholding – which aims to reject the correspondences to the visual words that fall beyond a specified threshold (*i.e.* the distance d_k between a key-point and the k-th visual word is greater than the threshold t_d). The threshold might be defined as a global constant ($t_d = t_g = const$) or local value t_d. The value of local threshold may vary depending on the characteristics of particular patch of descriptor space. We propose to compute the local threshold as a multiple of estimated cluster size. Possible method of estimating the size of a cluster relies on covariance matrix of distances distribution for the key-points that were used to create the cluster. Nevertheless, due to the computational complexity of calculating such matrices for all the visual words, we have decided to use a simpler estimator. We propose to approximate cluster size by the distance from visual word to the nearest neighbouring cluster centre (see Fig. 5.5). We consider this parameter as a relatively simple and representative description of local feature space characteristics. Hence, $t_d = t_l = n \cdot d_A$, where n will be chosen empirically.

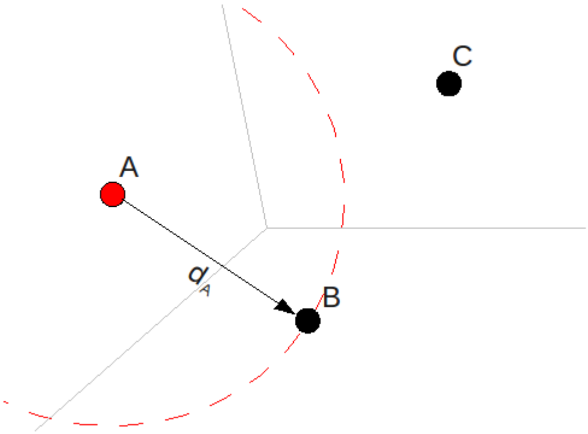


FIGURE 5.5: Distance to nearest neighbour used to estimate cluster size.

Secondly, we propose a post-processing stage that discards the visual words that are associated with relatively low L_1 normalised weights (below the threshold t_w). Though it shall not influence the overall performance, since such low weighs will not play a significant role at the next stages of image retrieval, it may significantly reduce the computational cost through the reduction of tentative correspondences between the images.

Last but not least, we have observed that visual words are not distributed uniformly in the descriptor space (see Fig. 5.6). This is a consequence of k-means clustering algorithm described in section 2.3.2. Thus, we have analysed alternative weight computation method based on estimation of cluster sizes. According to that method the weight assigned to a visual word shall be an exponential with variable σ parameter that depends on the local characteristics of descriptor space.

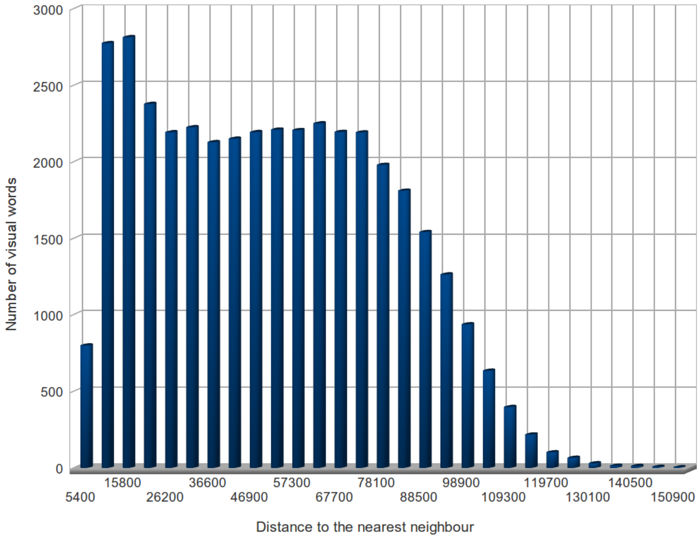


Figure 5.6: Distribution of nearest neighbour distances for visual words.

Hence, the weight assigned to a visual word is computed according to the following formula:

$$w_k^i = \exp\left(-\frac{d_k^2}{2\sigma_A^2}\right) \tag{5.4}$$

where d_k is a square Euclidean distance from i-th descriptor to the centre of cluster k, $\sigma_A = n{\cdot}d_A$, d_A is a distance from A-th visual word centre to its nearest neighbouring cluster centre and n is a parameter to be optimised.

5.3 Initial Experiments

Analysis and evaluation of the soft assignment is not a trivial problem, since the soft assignment mechanism needs to work in a complex system of visual search engine. Thus, in order to investigate various aspects of integrating soft assignment and detect potential irregularities, we have decided to isolate the problem and implement standalone evaluation tools that would enable us to analyse the influence of different configurations and parameters. The author is aware that there is no guarantee that the observations from these experiments easily translate to the results obtained when the soft assignment module is included in the visual search engine. However, they should show us in more details several important aspects of the soft assignment. We have developed two experimental set-ups that aim at:

1. **Analysis of the distribution of similarity measures depending on the distances** between descriptors.
2. **Analysis of the similarity between descriptors for image patches with known correspondence**.

5.3.1 Distribution of Similarity Measures Depending on the Distances

As it has been already explained, the main goal of the soft assignment is to capture similarities between descriptors in a reliable and consistent way through assignment of multiple visual words per key-point. Hence, we have decided to implement an evaluation tool that allows analysis of different soft assignment configurations in terms of similarity measure computed for pairs of descriptors with given distance between them. Since the distances between the descriptors can have wide range of values, we compute the statistical measure for a set of key-point pairs whose distance falls into a narrow predefined range of distances. We expect that the similarity values of key-point descriptors that are close to each other are higher than those of descriptors representing completely different image patch (*i.e.* those that are lying further away). Moreover, weight assignment shall be done in a consistent way, which means that the standard deviation of the similarity measure distribution for a given distance shall be lower for the soft assignment than for the hard assignment.

The evaluation data will consist of key-points extracted randomly from a set of Coca-Cola can images (Iris Dataset, see section 4.4) and from a set of All Souls church images (Oxford Dataset, see section 4.4). After extraction, distances between the key-point descriptors have been computed and corresponding similarity values have been calculated according to the Eq. 5.2.

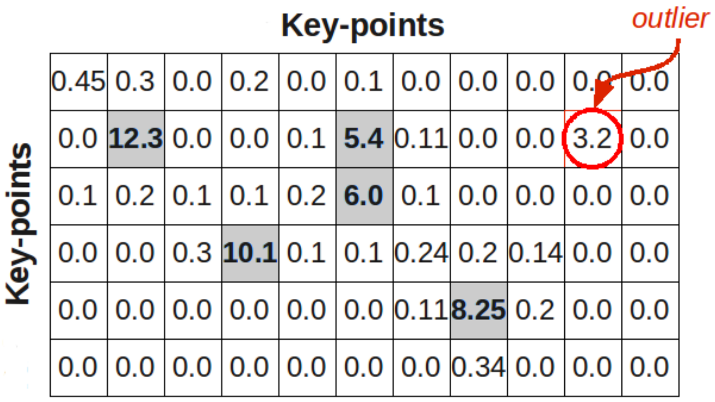
5.3.2 Analysis of the Similarity Between Descriptors for Image Patches with Known Correspondence

In order to better understand the influence of soft assignment configurations on distribution of similarity measure between key-points, we propose an evaluation tool which is based on the homography between two images. This experiment aims at showing how the soft assignment captures the similarity between descriptors that represent the same scene element in different images. We can understand the impact of parameters of soft assignment on matching similar image patches by analysing the similarity values that are associated to the correct, according to the homography, pairs of key-points.

The experiment will be performed using images from the data set presented in [10] (selected images are shown in Fig. A.8). In [10] they were used to evaluate different properties of local descriptors. Here, the data set will be used to analyse how soft assignment captures similarities between key-points describing the same part of the scene in different images. This will help us to verify which soft assignment configuration provides consistent and reliable way of capturing those similarities.

As a *ground truth* for the experiments, accurate homographies with root-mean square error of less than 1 pixel for every image pair are used. The homographies are provided with the data set and they are used to identify descriptors that are associated with the same part of a scene in several images.

Fig. 5.7 describes the main idea behind the homography-based evaluation experiment. We are using two sets of key-points and we compute similarity measure between them. Thanks to the accurate homographies between the images, we know the correct matches between key-points (drawn in grey), *i.e.* matches the reader should be advised that in this section we have discussed only the results that provided us with valuable and interesting observations of the soft assignment mechanisms. Extensive evaluation of all the configurations, including the ones that have not been discussed in this section, will be performed after the integration of soft assignment into the search engine. between key-points that describe the same scene element in different images. We will analyse the distribution of similarity measure for all matches. Ideally, we shall observe the maximal similarity values for the correct matches. However, as shown in Fig. 5.7, we will also be able to detect incorrect matches with high values of similarity measure.


Key-points
outlier
Key-points

0.45	0.3	0.0	0.2	0.0	0.1	0.0	0.0	0.0	0.0	0.0
0.0	**12.3**	0.0	0.0	0.1	**5.4**	0.11	0.0	0.0	3.2	0.0
0.1	0.2	0.1	0.1	0.2	**6.0**	0.1	0.0	0.0	0.0	0.0
0.0	0.0	0.3	**10.1**	0.1	0.1	0.24	0.2	0.14	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.11	**8.25**	0.2	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.34	0.0	0.0	0.0

Figure 5.7: Representation of similarities between descriptors for image patches with known correspondence. Cells represented in grey correspond to the correct matches between key-points. Incorrect match with high similarity value assigned (denoted as an *outlier*) is encircled in red.

Evaluation Measures

The following evaluation measures were used to evaluate different configurations of soft assignment and the influence of their parameters:

- **Sum of similarity values assigned to all correct matches**: used to observe whether a given configuration assigns higher similarities to descriptor pairs representing the same scene elements.
- **Ratio (in percentage) between weights assigned to the correct matches and weights assigned to all matches**: used to observe whether the similarities assigned to descriptors representing the same scene elements are higher than similarities assigned to unrelated descriptor pairs. The complimentary measure – ratio between weights assigned to the incorrect matches and weights assigned to all matches – may be used to observe the noise level.
- **Ratio between the number of correct matches with non-zero similarity value and all the correct matches**: used to verify what percentage of correct matches is discovered by a given visual word assignment configuration.

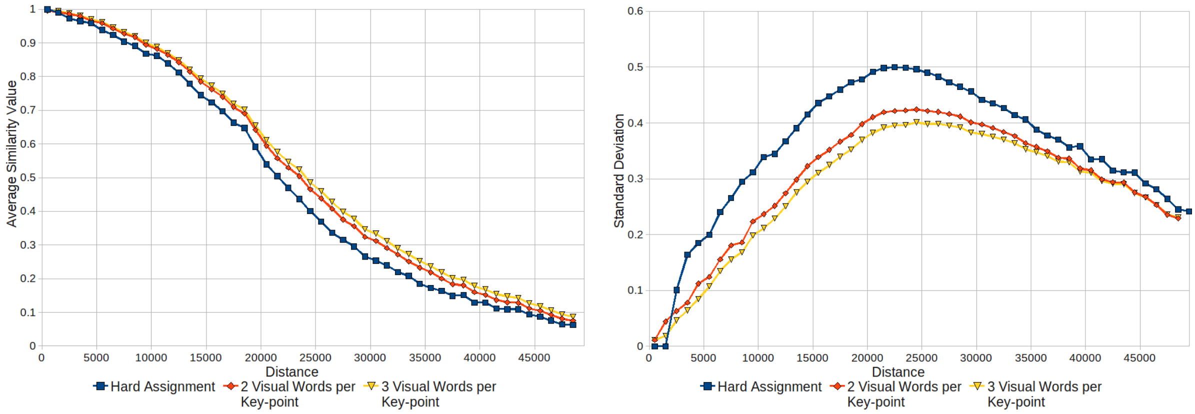
5.3.3 Results

The presentation of experiments' result will be divided into two parts: (i) results of experiments with *adapted solution* proposed in [8] and (ii) with *additional elements and configurations* proposed in this thesis.

Adapted Solution

Fig. 5.8(a) shows the average similarity value assigned to descriptors by the soft assignment with respect to the distance between them. We can observe that the distribution of the average similarity value of the key-point pairs is similar for soft and hard assignment. This can be explained by the fact that in both cases we are assigning a weight in the range of $[0, 1]$ to the visual word. However, in the case of the hard assignment the assigned weight can be equal 0 or 1. With soft assignment weights associated with key-point descriptors have more precise values that depend on the distance to the visual word.

The difference between the assignment of weights with the soft and the hard assignment can be seen in the graph showing the standard deviation of the similarity values versus distances between descriptors' pairs (Fig. 5.8(b)). We can confirm that in the case of the soft assignment weights are assigned in a more consistent way – standard deviation is lower by approximately 0.1 when comparing peak values. Thus, we are finding similarities between the descriptors more reliably. Similar results were obtained for DART descriptor and for All Souls church images (see Fig. B.1 and B.2 in Appendix B).

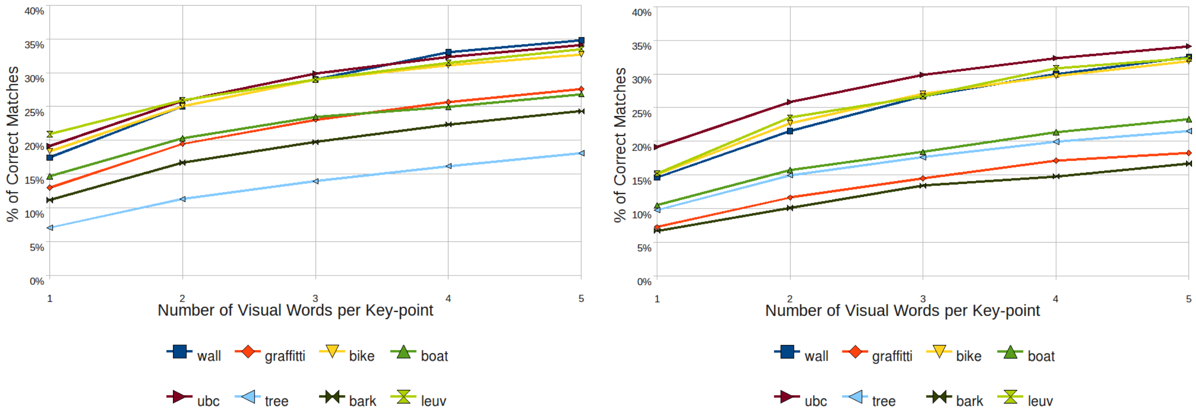


(a) Average similarity value

(b) Standard deviation

FIGURE 5.8: Similarity values produced by different assignment configurations – Coca-Cola can (SIFT)

When analysing the results of the second experimental set-up (Fig. 5.9), we can observe that the number of correct matches with non-zero similarity value assigned increases with increasing number of visual words per key-point. However, the greatest growths can be observed when changing from one visual word per key-point to two and from two visual words to three. For higher number of visual words per key-point, the observed increase is relatively small. We can assume that adding more visual words per key-point cannot help to discover larger number of correct matches. We can also see that our observations hold for both testes descriptors: SIFT and DART behave similarly (see Fig. 5.9(a)) and DART (Fig. 5.9(b)).

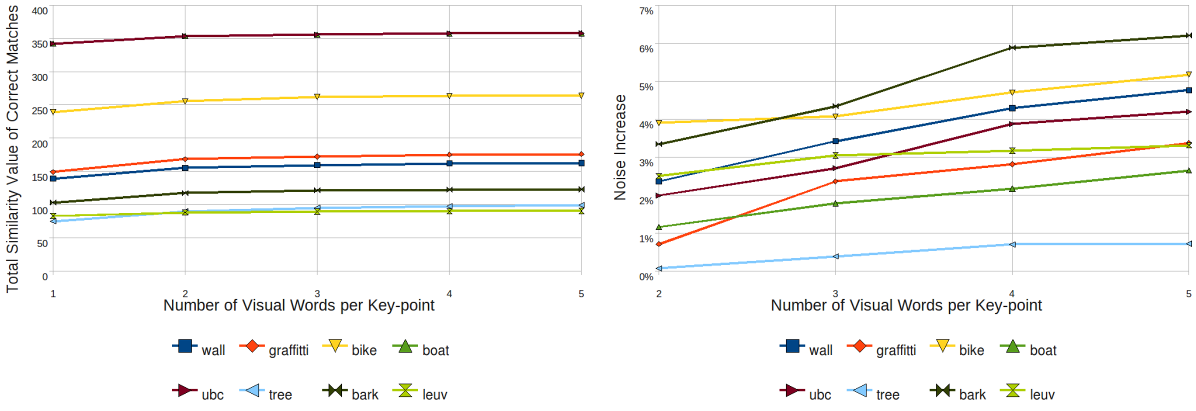


(a) SIFT

(b) DART

FIGURE 5.9: Correct matches with non-zero similarity values assigned

However, in Fig. 5.10(a), which shows the results of similarity value distribution for the true matches, we can see that the sum of similarity values assigned to the correct matches does not increase with increasing number of visual words per key-point. Furthermore, in Fig. 5.10(b) we observe an increase of the noise computed as a ratio of similarity values assigned to incorrect matches against the total similarity value with respect to the corresponding ratio for hard assignment. This phenomenon can be explained by looking again at Eq, 5.2 and the example in Fig. 5.3. According to the formula, even similar descriptors do not obtain similarity value exactly equal to 1 if they do not fall very close to the same visual words. In other words, for the soft assignment sum of similarity values associated with the correct matches is not greater than corresponding sum for hard assignment. In the case of the soft assignment, similarity values are assigned to all relevant matches in a more uniform way. In the case of the hard assignment, all the similarity values were contributed by fewer *lucky* links. Thus, the soft assignment promises better robustness to various image transformations and changes of dictionary quality. Additional results confirming our remarks can be seen in Fig. B.3 in Appendix B.



(a) Similarity values assigned to correct matches

(b) Noise increase

FIGURE 5.10: Characteristics of matches discovered with soft assignment (SIFT)

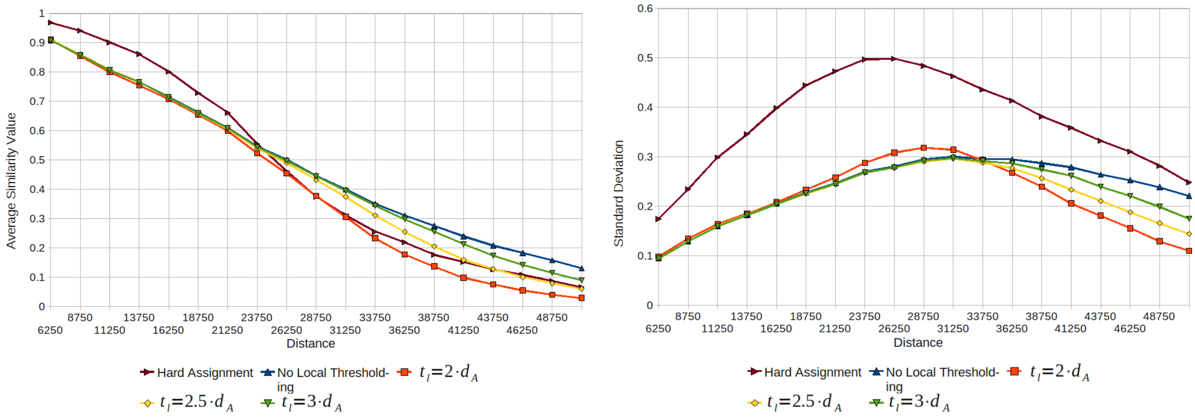
Summarizing, we can state that the results of the experimental set-ups show that the soft assignment can capture similarities between the key-points in a more reliable and consistent way. Since according to the presented results the soft assignment performs well with $r = 3$

visual words per key-point, in the remainder of this thesis this value will be used as a default configuration, unless stated otherwise.

Additional Configurations

As discussed in section 5.2.2, we have proposed additional pre-processing stage that discards visual words, if their distance to descriptor is greater than the threshold t_l defined according to the cluster size d_A. Since $t_l = n \cdot d_A$, we have decided to observe influence of various values of n on the process of weights assignment.

The results in Fig. 5.11 show that using initial distance-based thresholding we are indeed reducing similarity values for matches that consist of distant key-points. At the same time, it shows that distribution of the similarity values within smaller distances remains unchanged independently of the parameter n. This means that the local thresholding based on distances shall not influence the matches of high similarity that consist of similar key-points, but shall reduce computational cost of processing matches with key-points lying within greater distances from each other. Additional results confirming this observation for DART can be seen in Fig. B.4 in Appendix B.



(a) Average similarity value

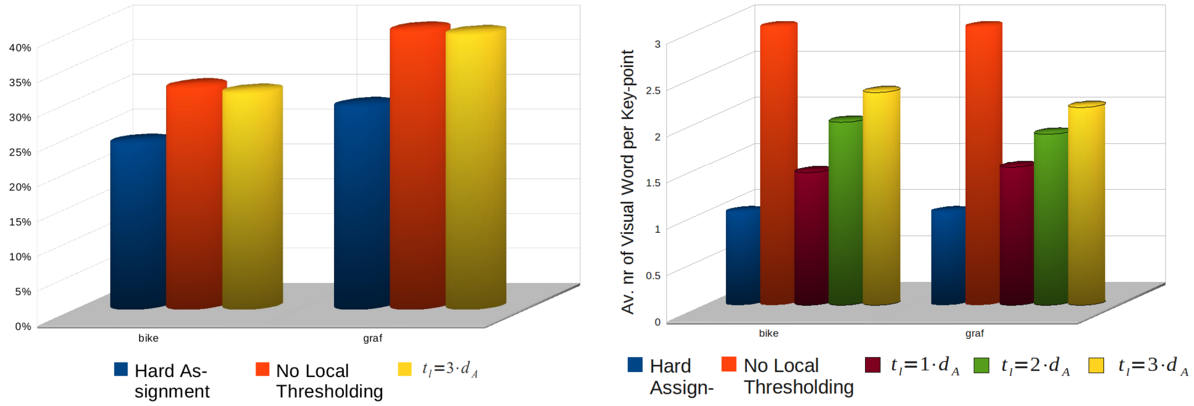
(b) Standard deviation

Figure 5.11: Similarity value distribution with local thresholding – Coca-Cola can (SIFT)

Furthermore, analysing Fig. 5.12(a), we see that the number of correct matches with non-zero similarity value assigned decreases by less than 2% with the local thresholding ($t_l = 3 \cdot d_A$) with respect to the configuration without thresholding. Fig. 5.12(b) shows average number of visual words assigned to key-points. As we can see, the local thresholding with $t_l = 3 \cdot d_A$ reduces the average number of visual words from 3 (no thresholding) to approximately 2. Thus, we can conclude that using distance-based rejection we can reduce the memory consumption (less visual words per key-point) without reducing number of correct correspondences found.

Initial Conclusions

The reader should be advised that in this section we have discussed only the results that provided us with valuable and interesting observations of the soft assignment mechanisms. Extensive evaluation of all the configurations, including the ones that have not been discussed in this section, will be performed after the integration of soft assignment into the search engine.



(a) Percentage of correct matches with non-zero similarity value assigned.

(b) Average number of visual words per key-point

FIGURE 5.12: Benefits of distance based thresholding (SIFT).

As we can see from the above results, the experimental set-ups provide interesting insights into the soft assignment mechanism and help to understand this complex technique. It should be noted that the above mentioned analysis are novel and have not been described in the literature. They enabled us to detect irregularities, confirm or invalidate our expectations and verify our implementation. However, the author is aware that we shall not rely on them as a final parameters optimisation method, since they cannot visualise the impact of the soft assignment on overall system performance.

We have observed that a higher number of visual words per key-point enables more reliable capture of similarities between feature descriptors. We are consistently discovering more matches between images. We have also observed that the additional visual words increase similarity values for incorrect matches that lie close in the descriptor space.

It has been also shown that the local distance-based thresholding discards matches with low similarity values which leads to a reduced number of tentative correspondences and the reduction of the computational cost. Nevertheless, we shall check the influence of this rejection on the whole image retrieval system.

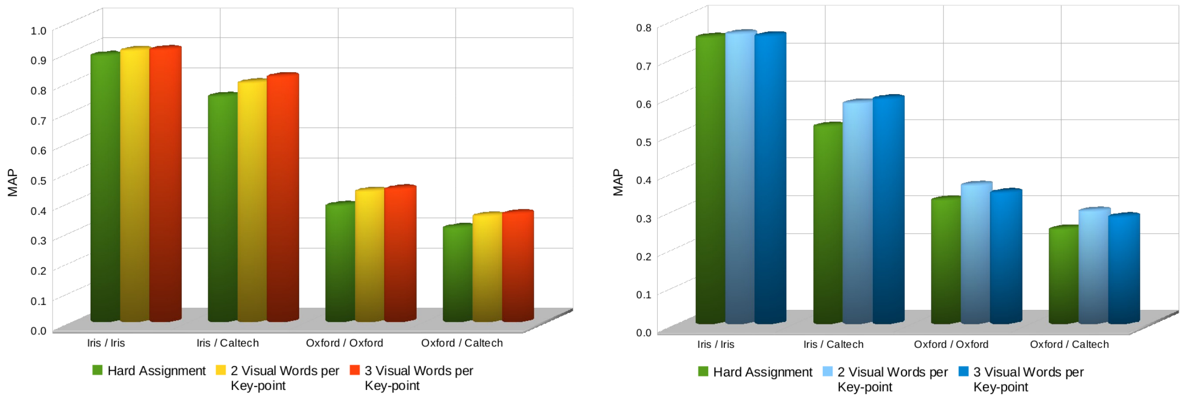
To summarize the initial experiments, we could conclude that, after implementing some extensions, the solution proposed by J. Philbin for *tf-idf* weighting scheme appears to be a very promising option for improving the performance of the visual search engine developed at Telefónica I+D.

5.4 Visual Search Engine Results

In this section the most promising configurations of the soft assignment will be evaluated according to their performance within the entire visual search engine. The evaluation will be done using the framework proposed in chapter 4. Firstly, we will discuss quantitative results obtained for basic configuration of soft assignment followed by results obtained for additional configurations presented in section 5.2.2. Afterwards, final configuration will be defined and examples of qualitative results will be presented.

5.4.1 Quantitative Results

The overall quantitative results of the visual search engine evaluation with the adapted solution of the soft assignment in the basic configuration are presented in Fig. 5.13. When comparing the hard assignment and the soft assignment with $n = 3$ visual words per key-point, for the soft assignment MAP increases 0.04 on average with respect to the hard assignment. This increment is comparable with the results presented in [8] that were obtained in similar conditions. The increase for SIFT is more significant in the case of generic dictionaries (0.07) and more difficult dictionaries, *e.g.* the Oxford collection (0.05 with the dictionary created from Oxford reference images). For SIFT maximal MAP is obtained mostly for 3 visual words per key-point, while for DART it reaches its peak mostly for 2 visual words per key-point. Differences between optimized parameters for descriptors may be caused by different distributions of visual words in the descriptor space for SIFT and DART. Summarizing the results, we can observe that even though the increase of MAP is not relatively high, it is significant and coherent within all the data sets and dictionaries. Low values of MAP increment may be explained by the corresponding increase of noise that is caused by additional irrelevant matches. This shall be eliminated by the spatial verification phase described in chapter 6. Additional results of the performed experiments for the basic configuration can be found in Appendix B (Tab. B.1).



(a) SIFT (b) DART

Figure 5.13: Overall results of the visual search engine evaluation with the adapted solution for different collections and dictionaries. *Legend:* in the x-axis the first name defines collection, the second – origin of the images used for dictionary creation.

We will now discuss results obtained from the evaluation of additional configurations presented in section 5.2.2. When comparing results for different configurations of distance based thresholding (Tab. 5.1 and Tab. 5.2) we have observed no significant differences between various set-ups. No performance drop was noticed also for configuration with weight based final thresholding (Tab. 5.3). Thus, we can state that thresholding in the configurations proposed above does not deteriorate performance quality of the system. However, it can potentially save computational cost.

Additionally, we have compared two variants of the Gaussian weights calculation: constant σ and variable $\sigma_A = n \cdot d_A$ where d_A is distance from visual word to the centre of neighbouring cluster. From results (Tab. 5.4 and Tab. 5.5) we can see that the system performs slightly better with the Gaussian weight assignment based on the constant value of σ. Maximal values of MAP obtained for constant value $\sigma = 10000$ (SIFT) and $\sigma = 5000$ (DART) are higher than those obtained with variable value of σ. This may be caused by the fact that in order to estimate

	SIFT			DART		
t_g Coll. / Dict.	**100000**	**200000**	∞	**100000**	**200000**	∞
Iris / Iris	0.89897	0.90443	0.90441	0.75126	0.75362	0.75363
Iris / Caltech	0.81032	0.81481	0.81481	0.58782	0.58816	0.58820
Oxford / Oxford	0.43012	0.43457	0.43456	0.34012	0.34215	0.34213
Oxford / Caltech	0.35987	0.36065	0.36064	0.27142	0.27883	0.27881

Table 5.1: MAP results for global distance based thresholding for different values of $t_d = t_g$.

	SIFT		DART	
n Coll. / Dict.	**2**	**3**	**2**	**3**
Iris / Iris	0.90443	0.90443	0.72685	0.72695
Iris / Caltech	0.81481	0.81481	0.57995	0.57995
Oxford / Oxford	0.43457	0.43457	0.33191	0.33191
Oxford / Caltech	0.36065	0.36065	0.27645	0.27645

Table 5.2: MAP results for the local distance based thresholding for different values of $t_d = t_l = n \cdot d_A$

	t_w Coll. / Dict.	**No thresholding**	**0.1**	**0.15**	**0.2**
SIFT	Iris / Iris	0.90441	0.90356	0.90452	0.90405
	Iris / Caltech	0.81483	0.81556	0.81695	0.82094
	Oxford / Oxford	0.43462	0.43567	0.43539	0.43844
	Oxford / Caltech	0.36071	0.36120	0.36177	0.36458
DART	Iris / Iris	0.75362	0.75353	0.75239	0.75018
	Iris / Caltech	0.58816	0.58822	0.58838	0.58719
	Oxford / Oxford	0.34215	0.34202	0.34183	0.34217
	Oxford / Caltech	0.27883	0.27881	0.27887	0.27954

Table 5.3: MAP results for final weight based thresholding for different values of t_w

the local cluster size we are using the distance to the nearest neighbour (see section 5.2.2). In the case of clusters whose shape is not circular, but *e.g.* elliptical, this may not be sufficient. Estimation of cluster size with covariance matrix could solve this problem, however, it would significantly increase computational time, which has to be taken into account when defining final configuration.

Computational Cost Analysis

Since the soft assignment implies higher number of visual words per key-point (see Fig. 5.12(b)), we can expect increased number of created hits and corresponding increase of memory use and computational time. This section discusses in details the computational cost of the implemented solution.

	SIFT			DART		
Coll. / Dict. \ σ^2	**5000**	**7500**	**10000**	**5000**	**7500**	**10000**
Iris / Iris	0.9041	0.9044	0.9092	0.7536	0.7485	0.7394
Iris / Caltech	0.8140	0.8148	0.8192	0.5882	0.5870	0.5838
Oxford / Oxford	0.4397	0.4346	0.4307	0.3422	0.3393	0.3372
Oxford / Caltech	0.3577	0.3607	0.3624	0.2788	0.2776	0.2772

Table 5.4: MAP results for constant σ.

	SIFT			DART		
Coll. / Dict. \ n	**1**	**2**	**3**	**1**	**2**	**3**
Iris / Iris	0.8861	0.8828	0.8826	0.7411	0.7343	0.7302
Iris / Caltech	0.8144	0.8187	0.8165	0.5827	0.5803	0.5810
Oxford / Oxford	0.4262	0.4160	0.4128	0.3464	0.3374	0.3345
Oxford / Caltech	0.3577	0.3616	0.3609	0.2798	0.2772	0.2771

Table 5.5: MAP results for variable $\sigma = n \cdot d_A$

Coll. / Dict.	Hard Assignment	Soft Assignment	Soft Assignment + Global Dist. Threshold.	Soft Assignment + Weight Threshold.
Iris / Iris	152.88 ms	205.76 ms	203.24 ms	190.59 ms
Iris / Caltech	215.00 ms	274.41 ms	269.53 ms	257.20 ms

Table 5.6: Average query time results for different configurations (feature extraction time not included). Tests done on machine with Intel Core2 Duo 2.33 GHz and 3 GB RAM.

The soft assignment parameters were based on the above results and the thresholds were set to the highest values ensuring no influence on system performance ($t_g = 200000$ and $t_w = 0.2$). As expected, the results for the different soft assignment configurations (Tab. 5.6) clearly show that the increase of the performance quality (measured in MAP) does not come without additional cost – due to the increased number of correspondences between query and reference images additional computation time is needed. This increase can be explained by the fact that every image feature has more then one visual word assigned and can potentially match more features in the other image. Since the analysis of the correspondences takes more time, the average query time increases. Furthermore, along with the increasing number of visual words per key-point, memory usage growth has been observed. It is caused by greater number of visual words per key-point.

Thus, we expect one of the proposed thresholding mechanism to reduce the number of tentative correspondences and computational cost without decreasing performance quality. As we have seen above, the performance quality does not suffer neither from the distance nor the weight based thresholding. However, the shortest average query time was observed for configuration of soft assignment with the weight based thresholding. Even though the obtained reduction of the computational time might not seem to be important for standalone computers, it may become significant when the system needs to be deployed on mobile devices.

Final Configuration

The best compromise between the retrieval results and the computational cost is obtained by the following configuration of the soft assignment:

- **Number of visual words per key-point**: 3 for SIFT, 2 for DART
- **Weighting method**: constant $\sigma^2 = 10000$ for SIFT, $\sigma^2 = 5000$ for DART
- **Weight based thresholding**: threshold set to $t_w = 0.2$ for SIFT and DART

5.4.2 Qualitative Results

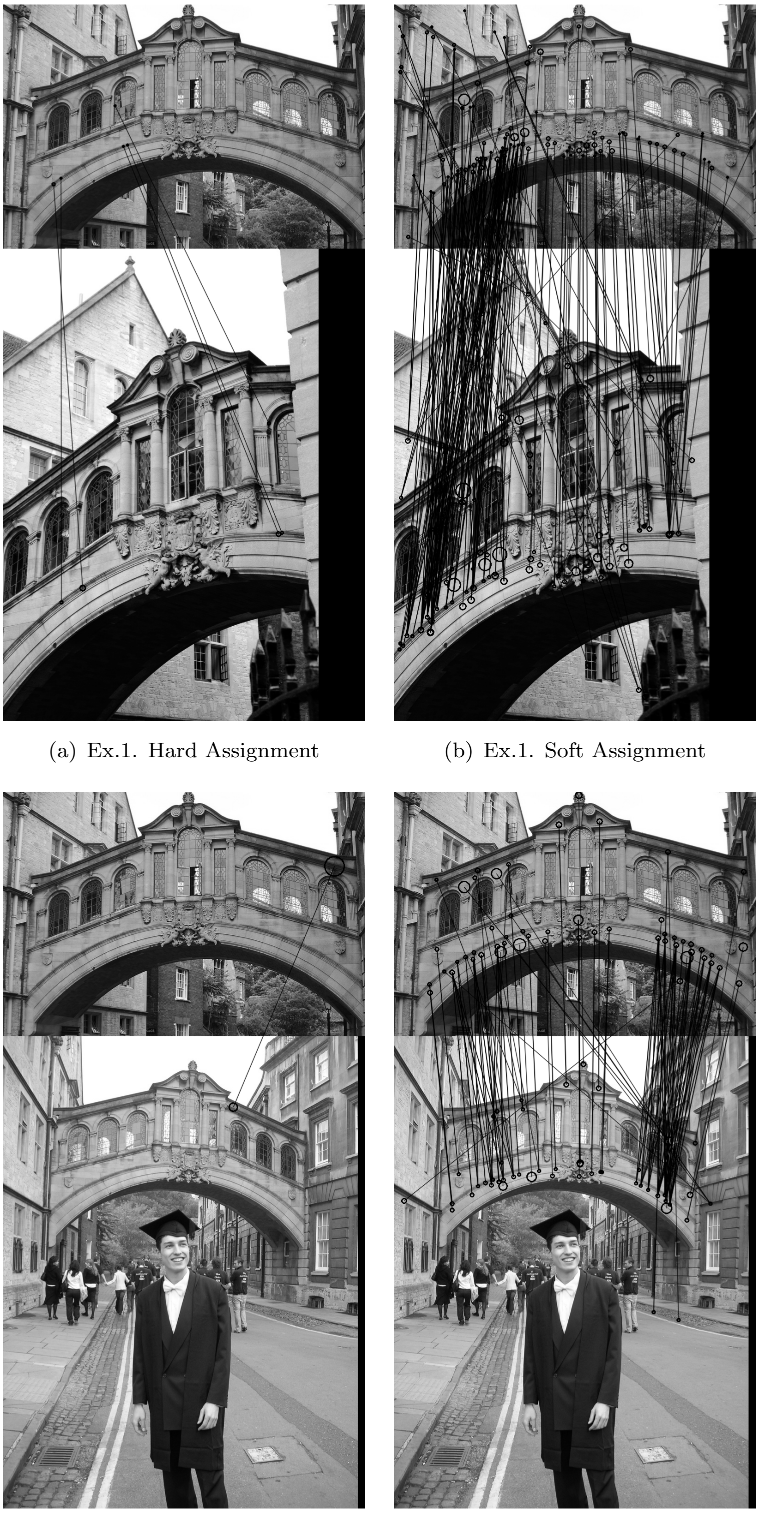
In order to observe the influence of the soft assignment, we have done several visual inspections of the results with the soft assignment mechanism for the configuration defined above. Examples can be seen in Fig. 5.14. In general, we can see many more correspondences found between the query and reference images. This kind of behaviour suggest that the visual search engine with the soft assignment should produce higher recall due to the smaller number of missed relevant matches. It is especially important, when the number of identified matches is as low as 1 (see Fig. 5.14(c)). In this case, reliable identification without the soft assignment is impossible. It should be observed that not all new correspondences shown in Fig. 5.14(b) and Fig. 5.14(d) are correct. This confirms that soft assignment also introduces certain noise to the recognition process. It may affect the precision of the retrieval results. Rectification of this shortcoming will be discussed in chapter 6 when we introduce the additional spatial verification of matches.

Moreover, we have observed that, in general, the final scores computed for reference images retrieved by the search engine in response to a query image behave differently for the soft and the hard assignment. Example of this behaviour can be seen in Tab. 5.7. In the case of the hard assignment, the difference between the scores of the first and the second retrieved reference images is higher than in the case of the soft assignment. Furthermore, for the hard assignment the ratio between the scores of the last relevant image (4^{rd} position) and the first irrelevant image (following position) is relatively low (1.04). This type of behaviour may lead to confusion and may unable meaningful thresholding of retrieved relevant images. This is not the case for the soft assignment – the ratio between the scores of the last relevant and the first irrelevant images is much higher (2.5). In general, we can see that the scores obtained for relevant images with the soft assignment decay much more smoothly than those of hard assignment. It is due to the fact that with the soft assignment we assign weights to the key-points in a much *softer* way, hence mitigating quantisation effects.

5.5 Discussion

Although the soft assignment leads to a higher number of correct matches between the query and the reference images, not all of the new matches are correct. As it is shown in Fig. 5.14, most of these incorrect matches are not spatially coherent. Thus, complementary post-processing shall be performed in order to verify spatial layout of discovered matches (see chapter 6).

During the computational cost analysis we have found that the estimation of clusters' size that were to be used in weight computation with variable σ and the local distance based thresholding



(a) Ex.1. Hard Assignment

(b) Ex.1. Soft Assignment

(c) Ex.2. Hard Assignment

(d) Ex.2. Soft Assignment

FIGURE 5.14: Comparison of correspondences found with hard and soft assignment between query and reference images. Images come from Oxford Dataset and depict Herford College. The employed dictionary was created from reference images from Oxford collection. The spatial consistency verification is not introduced.

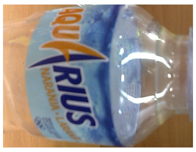
Query Image	Results						
	**Hard Assignment**						
	Score	246.27	23.37	15	14.58	14	13.1
	Soft Assignment						
	Score	241.18	58.41	57.46	52.5	35.2	14.04

Table 5.7: Results of image retrieval with and without soft assignment. 6 top ranked retrieved images are presented.

last over 20 minutes for each dictionary with 100 000 visual words. Such a long computational time renders this solution cumbersome in the practical solutions where there may be a need for rapid retrieval. Furthermore, it shall be noted that we have obtained better results for constant σ parameter and the local distance based thresholding does not lead to any essential performance improvement. Thus, we have decided to not include these solutions in the final configuration.

5.6 Conclusions

Concluding this chapter, we can confirm that the performed experiments, visual inspection of the matches and evaluation of visual search engine performance show improvement when using the soft assignment with respect to the system with the hard assignment.

Thus, it has been positively verified that the solution proposed by J. Philbin for *tf-idf* weighting scheme has been successfully adapted to the architecture of search engine developed at Telefónica I+D.

Although if we analyse only MAP results, improvement obtained for the soft assignment may appear modest, it is due to the noise introduced by the additional matches. We expect that spatial layout verification (see chapter 6) will eliminate that noise, further boosting the results.

Moreover, we have observed that the improvement is more significant for more challenging configurations, *i.e.* with generic dictionaries (Caltech) and more difficult data set (Oxford, outdoor scenes). This becomes even more important when taking into account potential application of the search engine in high precision geo-localisation services where high quality dictionaries may not be available. It is due to the fact that in those applications it may not be practical to use fine tuned dictionaries for different locations and employed databases may change dynamically.

Furthermore, it has been observed that scores obtained for reference images as a result of a query are assigned in a much *softer* way, *i.e.* they decrease more smoothly, mitigating quantisation effects.

Last but not least, we have confirmed that the weight based thresholding mechanism, which was proposed in this thesis as an extension of the soft assignment mechanism described in [8], provides essential reduction of the computational cost without influencing the overall performance quality.

Chapter 6

Spatial Consistency Verification

This chapter discusse the spatial consistency verification mechanism proposed to increase discriminatory power of the visual search engine developed at Telefónica I+D. A RANSAC inspired algorithm is used for similarity transformation model estimation and identification of local correspondences consistent with that model. Finally, the results of visual search engine evaluation are presented and the final conclusions are drawn.

6.1 Problem Statement

The principal objective of visual search is to retrieve images depicting the same object that appears in the query image. In order to do so, many sophisticated methods and techniques have been employed, *e.g.* representation of salient regions with descriptors invariant to various condition changes. However, independent comparison of local features is often insufficient for precise image retrieval. For instance, in the case of outdoor scenes and buildings with façades of repetitive, common and small structures, correct recognition based solely on co-occurences of visual words might be impossible. Even though local features provide relatively high discriminatory power, as discussed in [12], it is probable that there are several image patches in the scene that have identical descriptors, even though they do not represent corresponding patches.

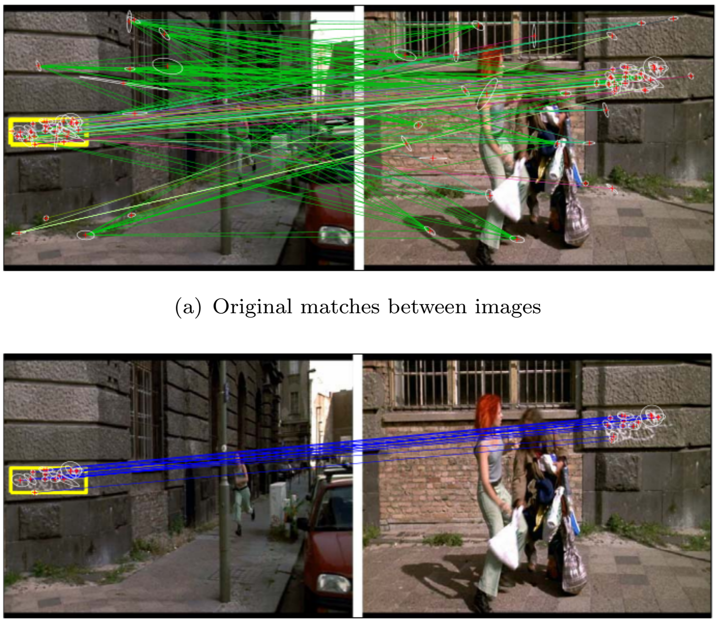
This is especially visible for image retrieval systems based on visual word vocabularies where the number of visual words in the dictionary is the limiting factor of the discriminative power. Hence, the resulting coarseness of quantisation (visual word assignment) can significantly influence the system performance. In general, the larger the dictionary used, the more visual words employed, hence the higher the discriminative power of the system. Therefore, in the cases of dictionaries with lower number of clusters and collections of significant size, comparison of local features may not suffice for acceptable recognition performance.

Bearing in mind potential mobile applications, we would like the visual search system to be generic and as independent as possible from the characteristics of dictionaries and collections used. This way we would increase robustness and reliability of the solution, simultaneously widening the range of possible applications.

6.2 Proposed Solution

One of the possible solutions to the abovementioned problem is an implementation of additional post-processing stage that increases discriminative power by verification of spatial consistency between local correspondences. This extension can reduce the number of situations where several similar image patches in the scene create false matches. This can be achieved by verifying spatial consistency of the correspondences between a query and reference images. This solution has been proven to be successfully implemented in various systems [3, 6, 8]. It can be also interpreted as an analogy to the Google text retrieval mechanism that increases the ranking for documents where the searched words appear close to each other in the retrieved text [3].

Because of the potential growth of the number of correspondences between the images due to employment of the soft assignment mechanism (discussed in chapter 5), we shall be interested in rejecting the additional erroneous matches. The so-called *spatial re-ranking* was found useful to solve this problem [8] due to its ability to eliminate additional false correspondences introduced by the soft assignment. An example of the desired effect of spatial verification can be seen in Fig. 6.1.



(a) Original matches between images

(b) Matches after filtering on spatial consistency

Figure 6.1: Example result of spatial consistency verification. *Source:* [3]

There are different methods of enforcing the spatial consistency between local matches – from loose variants, requiring only neighbouring matches in a query image to lie in a surrounding area in a reference image, to more strict that require exactly the same spatial layout in both images. It needs to be pointed out, however, that the decision about the regime of the spatial consistency verification has a significant impact on both the precision and the robustness of the system. Furthermore, more complicated and computationally expensive models, *e.g.* affine transformation, may greatly increase computational time, thus rendering possible mobile application infeasible. Because of that, in our solution only simple transformation of scale, rotation and shift are accounted for.

Several methods of estimating parameters of a transformation model from a data set containing outliers have been proposed in the literature. Most commonly used are Hough transform [32] and RANSAC algorithm [33]. Hough transform provides highly robust and reliable way of model estimation. However, with a high number of model parameters, its computational cost is relatively high due to the memory requirements and the computational complexity. The RANSAC algorithm performs much faster, nevertheless it requires at least 50% of inliers in the data set to estimate the model [33]. Furthermore, it is a non-deterministic method, hence, the correct result is obtained only with certain probability.

Considering all the above, we adapt the most recent solution discussed in the literature and implement spatial consistency verification mechanism based on simple homography. We propose to perform the spatial consistency verification in two steps. Firstly, the spatial consistency of local correspondences are verified through the initial voting mechanism in the reduced pose space (as described in section 3.2.5). Then, only the matches selected in the first stage are passed to a post-processing stage that re-ranks the items retrieved by visual search according to the much stricter measure of spatial coherence between the matches. Since the initial spatial verification is rather coarse, in the second step we attempt to exploit entire spatial information which shall further improve results. Furthermore, in the second stage of the proposed solution, we benefit from the voting weights computed in the first stage and we use them to find the most meaningful transformation model. It shall be also noted that our solution is capable of processing the images of different resolutions.

This chapter primarily focuses on the development of the second stage that in the remainder of this thesis will be referred to as the spatial consistency verification module.

6.3 Implementation

In our variant of the spatial consistency verification an input of the spatial re-ranking module is a subset of reference images that have been initially ranked using the vote accumulators described in details in section 3.2.5. It shall be reminded that this initial solution ensures consistency only in terms of scaling and rotation between the local matches. It shall be noted that only the top ranked reference images are passed to spatial consistency verification stage described in chapter. More precisely, the local matches between a query and reference images with associated voting weights (see section 3.2.5) are passed to the spatial consistency verification module. We use algorithm inspired by RANSAC to generate a set of hypothetical transformations based on the local correspondences. Then, each of the transformations is evaluated and, relying on the sum of weights of inliers, the one fitting correspondences with highest associated weights is chosen. In other words, the proposed spatial consistency verification mechanism estimates the transformation between the object view in a query and reference images and re-ranks reference images based only on matches coherent with that transformation. As an output, we obtain consensus set of inliers along with the corresponding transformation model which may be then used *e.g.* for precise geo-localisation based on image processing.

Detailed description of implementation along with the parameters of the method can be found below.

6.3.1 Overview of the RANSAC Algorithm

The RANdom SAmple Consensus (RANSAC) algorithm [33] has become a standard solution for estimation of model parameters from a set of data points which contains outliers. Unlike many others commonly used robust estimation algorithms, *e.g.* least-median squares, it has not been adopted by computer vision community, but was developed from within it.

As the name indicates, RANSAC is re-sampling technique that generates hypothesis from a randomly selected subset of data points. It runs according to the bottom-up approach: it chooses smallest set that is necessary to define model parameters and proceeds to enlarge it with consistent data points. The overview of algorithm can be seen in Alg. 6.1.

Algorithm 6.1 RANSAC

```
1. Choose at random subset of data points ($n$ samples) that would suffice to determine model
   parameters.
2. Find model parameters.
3. If any of the data points out of subset fit the determined model with a predefined tolerance
   $t$, add it to the consensus subset.
4. If the number of elements in the subset exceeds a predefined threshold $d$, refine the model
   parameters using all the data points in the subset. Afterwards, terminate.
5. Otherwise, repeat steps 1 through 4 (maximal number of iterations $N$ defined as an input).
```

The values of parameters t and d need to be defined empirically taking into account application and data set. Moreover, the initial assumption of the algorithm is that the dataset includes more than approximately 50% inliers and that the distribution of those points can be fitted to a particular model.

Since RANSAC is a non-deterministic algorithm, a reliable and reasonable output is obtained only with a certain probability p. The number of iterations N need to be chosen high enough to ensure that at least one of the sets of random samples does not include an outlier with probability p. Let w be the probability that selected data point is an inlier ($w = \frac{\text{number of inliers}}{\text{number of all data points}}$). Then, w^n, where n is minimum number of data points required to estimate the model, represents the probability that all n points are inliers and $1 - w^n$ defines the probability that at least one of n points is an outlier. This renders selected model to be incorrect. Thus, for N iterations we have that:

$$1 - p = (1 - w^n)^N \tag{6.1}$$

With some manipulation:

$$N = \frac{\log(1 - p)}{\log(1 - w^n)} \tag{6.2}$$

For further information on RANSAC the reader is encouraged to study [33, 34].

6.3.2 RANSAC Inspired Algorithm

Our solution is highly inspired by the RANSAC algorithm. However, we have introduced several modifications, adapting it to our application. First of all, following suggestions from [8], we have excluded non-deterministic aspect of data points processing and we create hypothetical model from each of the possible matches. By selecting restricted set of transformations (see section 6.3.3) we are able to generate a transformation hypothesis with only *one* pair of corresponding features. It shall be noticed that we are interested only in choosing transformation model that fits correspondences with the highest associated voting weights. It is due to the fact that, according to our observations, the weights are able to capture well evidence brought by particular correspondence. Thus, we propose to substitute the threshold d, used to verify consensus set, with the maximal total weight of all correspondences from the previously found best consensus set. The comparison is done between the total weights of all matches in the consensus set instead of the number of correspondences. This way, we obtain as an output the set of matching features' pairs that are fitting the model and whose total weight is maximal with respect to other potential models. However, we take into account threshold d that defines minimum number of elements in the consensus subset in the last stage of the algorithm. The final consensus set is accepted only if it contains at least d elements. Otherwise, we will assume that no transformation model was found.

The overview of the proposed algorithm can be seen in Alg. 6.2.

Algorithm 6.2 Adapted RANSAC

```
1. Choose one pair of corresponding features.
2. Find model parameters for this pair.
3. If any of the remaining matches fit the determined model with a predefined tolerance $t$,
   add it to the consensus subset.
4. If the total weight of correspondences in the consensus subset is greater than total weight of
   matches in previously found best consensus set, set current subset as a new best consensus
   set.
5. Repeat steps 1 through 4 for all matches.
6. If number of elements is greater than threshold $d$ refine the model for the best consensus
   set found and terminate. Otherwise, return information that no correct model was found.
```

6.3.3 Transformation Model

The proposed spatial verification mechanism estimates the transformation model between objects present in query and reference images. Several transformation models can be employed, ranging from the simplest ones, that include only translation, to more complex ones that account for affine transitions. However, while choosing the transformation model to be used, we shall take into account computational complexity of calculating such model. Furthermore, we shall be also aware of the limitations defined by features, since employed local key-points provide different spatial information, *e.g.* depending on the size and shape of the extraction region. For instance, descriptors based on affine regions contain additional information about scales computed along particular orientation. Square based descriptors, *e.g* SIFT, do not comprise this data, since they provide only one scale value per key-point.

Taking into account all arguments mentioned above, we have decided to implement a simple transformation model that accounts for translation, rotation and scaling. In the literature, it is commonly referred to as the *similarity transformation* model [34]. In fact, it describes an isometric transformation composed with isotropic scaling. Its simplicity reduces the computational cost. This model can be computed from a *single* pair of matching key-points, significantly simplifying the hypothesis generation stage.

Let $\mathbf{x} = (x \quad y \quad 1)^T$ be input homogeneous coordinates' vector in and $\mathbf{x}' = (x' \quad y' \quad 1)^T$ define output coordinates [34]. Then, we can write that:

$$\mathbf{x}' = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s \cdot \cos\theta & -s \cdot \sin\theta & t_x \\ s \cdot \sin\theta & s \cdot \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathbf{H_s} \cdot \mathbf{x} \tag{6.3}$$

where the scalar s represents the isotropic scaling, θ represents rotation in radians and $\mathbf{t} = (t_x \quad t_y)$ – translation parameters. $\mathbf{H_s}$ can be defined as the *similarity transformation matrix.*

In our implementation, we take a single pair of corresponding key-points from query and reference images and we generate a model, using Eq. 6.3. The transformation is considered to take place from the reference image to the query. Thus, we compute the scaling factor as a ratio between the scale of the query key-point and the scale of the reference key-point. The rotation is calculated as a difference between the query key-point orientation and the reference key-point orientation. The translation parameters are found by subtracting reference key-point coordinates from query key-point coordinates after accounting for scale and rotation change.

When verifying if a match fits the model, in order to obtain a reliable error measure we compute symmetric transfer error ϵ_s [34]. Hence, we consider forward ($\mathbf{H_s}$) and backward ($\mathbf{H_s^{-1}}$) transformation, and we sum the geometric errors corresponding to each of these two transformations. Geometric error $d(\mathbf{x'_m}, \mathbf{x}')$ is defined as the square Euclidean image distance in the query image between the measured point $\mathbf{x'_m}$ and the point $\mathbf{x}' = \mathbf{H_s} \cdot \mathbf{x}$ at which the corresponding point $\mathbf{x}$ is mapped from the reference image. Thus, the symmetric transfer error ϵ_s for each pair of images is computed according to the following equation:

$$\epsilon_s = \sum_i d(\mathbf{x_i}, \mathbf{H_s^{-1}} \mathbf{x'_{i_m}})^2 + d(\mathbf{x'_{i_m}}, \mathbf{H_s} \mathbf{x_i})^2 \tag{6.4}$$

where i represents a correspondence between images.

It should be noted that, symmetric transfer error ϵ_s computed according to the Eq. 6.4 is expressed in square pixels. This means that in cases of scenes represented with images of low resolution, the error ϵ_s is unproportionally smaller in relation to the represented object than in cases of the same scenes represented with high resolution images. Therefore, in our implementation, we use normalised geometric error with normalisation in respect to square diagonal of the image. This solution, which has not been previously discussed in the literature, allows us to provide robustness of calculations when working with images of different resolutions. Hence, the final calculation of the normalised symmetric transfer error is done according to the following equation:

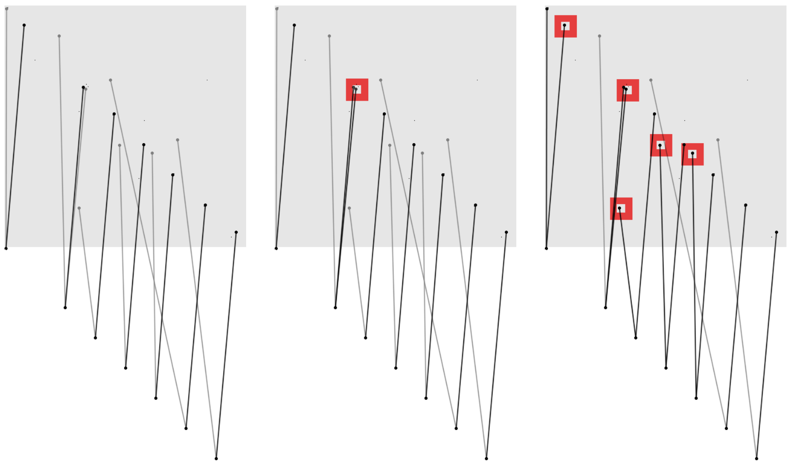
$$\epsilon_s = \sum_i \frac{d(\mathbf{x_i}, \mathbf{H_s}^{-1}\mathbf{x'_{im}})^2}{\phi_R^2} + \frac{d(\mathbf{x'_{im}}, \mathbf{H_s}\mathbf{x_i})^2}{\phi_Q^2} \tag{6.5}$$

where ϕ_R and ϕ_Q are, respectively, diagonals of reference and query images.

6.4 Initial Experiments

This section describes a set of initial experiments designed to validate the behaviour of the spatial consistency verification module.

Synthetic data



(a) $t = 0.01$ (b) $t = 0.05$ (c) $t = 0.1$

Figure 6.2: Initial experiment with synthetic data and various values of error threshold t. Inliers and outliers are represented with black and grey colours respectively. Inside red squares there are matches classified as outliers for $t = 10$ and as inliers for higher values of t.

First of all, we have designed and implemented an experimental set-up to verify the performance of our solution with a synthetic data. Hence, we have manually created several matches and randomly added additional ones. Then, we have selected three values of the error threshold t and checked the inliers selected by the algorithm. The results can be seen in Fig. 6.2. As we can see, the higher the threshold value, the more loose the condition defining if a match fits the model or not. Hence, more matches are classified as inliers.

Afterwards, we performed additional check of the implemented transformation model. We manually defined reference data points' coordinates with randomly chosen transformation parameters (translation, scaling and rotation). Then, we verified if the transformation model parameters returned by the algorithm were identical to the ones chosen at random. After successful validation of the results, we confirmed that the implemented solution performs well with synthetic data and synthetic transformations. An example of images obtained from this experiment can be seen in Fig. 6.3.

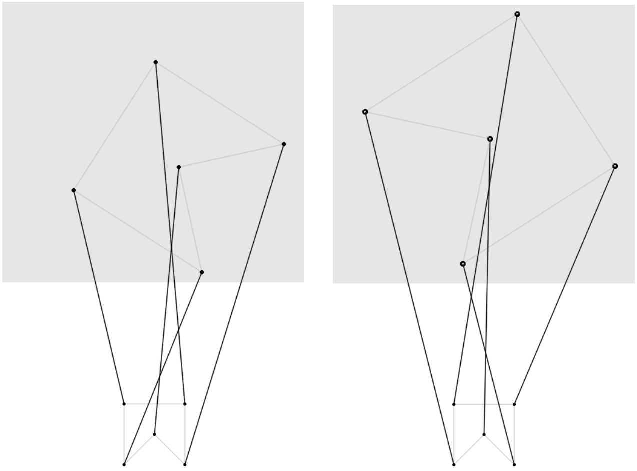


Figure 6.3: Initial experiment with synthetic data. Inliers are represented with black colour. Displayed shape borders are drawn in grey.

Real data with Synthetic Transformations

In order to validate the system performance with real data and synthetic transformations, several simple (translation, rotation, scaling) and more complex (sheer) transformations were applied to the image of the Coca-Cola can from the Iris Dataset (see section 4.4). The main objective of this experiment was verification of Eq. 6.3 and 6.5 with scaling and rotation parameters obtained from extracted key-points and further debugging of the implementation. Selected results can be seen in Fig. 6.4.

While analysing the results of the experiment, we have discovered that in the transformation model we shall take into account key-point scales that have not been normalised with respect to the region of interest (ROI). The normalisation procedure, discussed in details in section 3.2.2, is a consequence of ROI selection which aims to exploit the assumption about the presence of one well-posed object in reference image. Thus, we have concluded that in the spatial consistency verification we shall use only the original scales of key-points.

Real Data with Real Transformations

Finally, we validated the system performance with real data and real transformations. We have carried out a set of visual inspections of the query-reference image pairs and verified selection of the correct matches. Selected results can be seen in Fig. 6.5. We can see that the selected matches (represented with black colour) are spatially consistent and their forward and backward projections are coherent.

During the inspection of the images, we have discovered that while applying transformation equation 6.3 we shall also take into account image resolution by performing appropriate resizing.

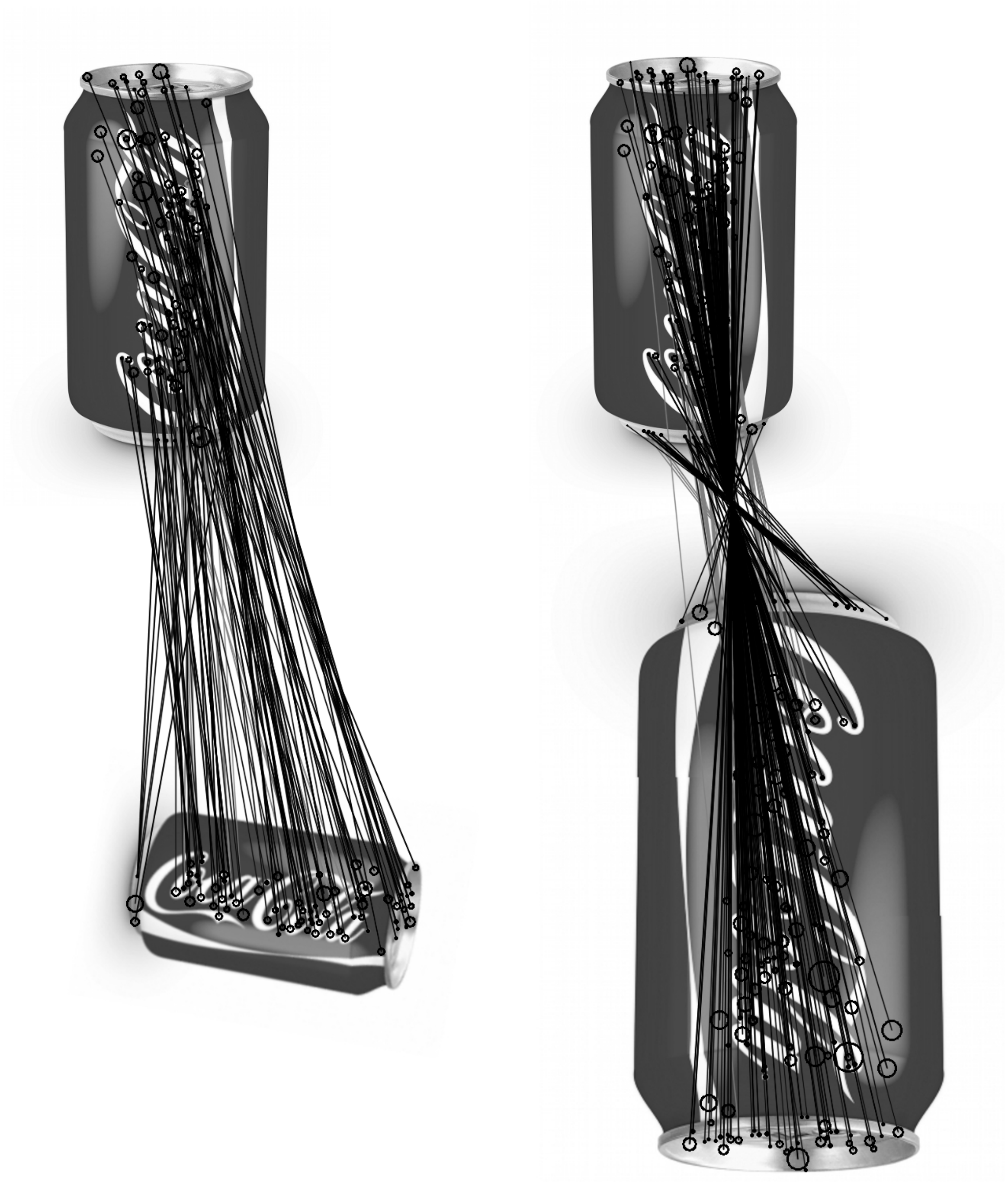
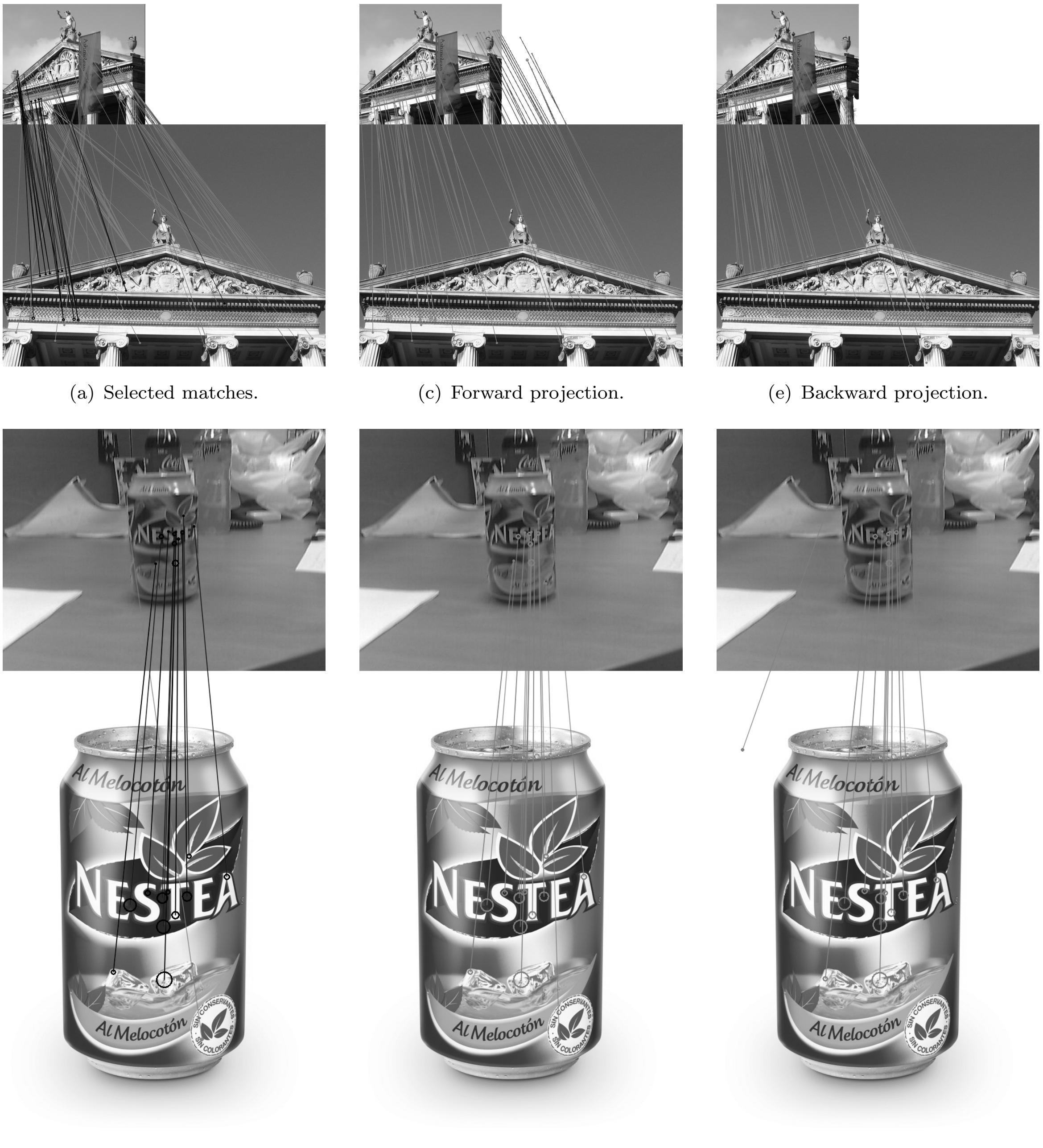


Figure 6.4: Initial experiment with real data and synthetic transformation. Inliers and outliers are represented with black and grey colours respectively.

6.5 Results

Once the initial tests validating the correctness of the implementation has been completed, the solution was incorporated into the visual search engine and its parameters optimised. Specifically, two parameters – the minimum number of inliers in the consensus set d and the symmetric error threshold t – needed to be tuned. The optimisation of the spatial consistency verification module was done with the soft assignment parameters set beforehand according to the results



(a) Selected matches.

(c) Forward projection.

(e) Backward projection.

(b) Selected matches.

(d) Forward projection.

(f) Backward projection

FIGURE 6.5: The initial experiment with real data and real transformation. Query images are displayed above reference images. Projections are done according to the best found transformation model. (a, b) Selected matches. Inliers and outliers are represented with black and grey colours respectively. (c, d) Projection of reference key-points into query image. (e, f) Backward projection of query key-points into reference image.

from chapter 5. It shall be noticed that after the integration with the search engine the spatial consistency verification was applied for subset of $n = 50$ reference images with the highest scores after ranking done in the earlier stages of the search algorithm.

The minimum number of inliers in the consensus set d defines the lowest number of consensus set elements that is required for the corresponding transformation model to be accepted as a correct model. Hence, the higher the value of the parameter d, the more discriminative power the algorithm has, because only models supported by more significant number of inliers will be accepted. The results of the optimisation of the parameter d can be seen in Fig. 6.6. In general, increasing value of d results in decreasing value of MAP. It should be noted that in the case of the Iris Data Set, the average number of possible correspondences (and hence potential inliers) per image is significantly lower than the average number of possible key-point pairs in the case of the Oxford Data Set. Thus, we can see that the results for the Iris Data Set are much more sensitive to the changes of the parameter d than those of the Oxford Data Set. In other words, for increasing values of d the decrease of MAP for collection with lower number of the possible matches is much more significant than the decrease for collection with higher number of the correspondences. This behaviour can be explained by the fact that the collections depicting simple scenes generate lower number of the matches with respect to the collections with the images of textured, complex scenes, hence being more sensitive to the changes of the system's discriminatory power.

Based on the abovementioned results it appears that the best compromise between collections that have many potential matches and collections with few matches can be obtained for $d = 3$. This value will be used in the further experiments, unless stated otherwise.

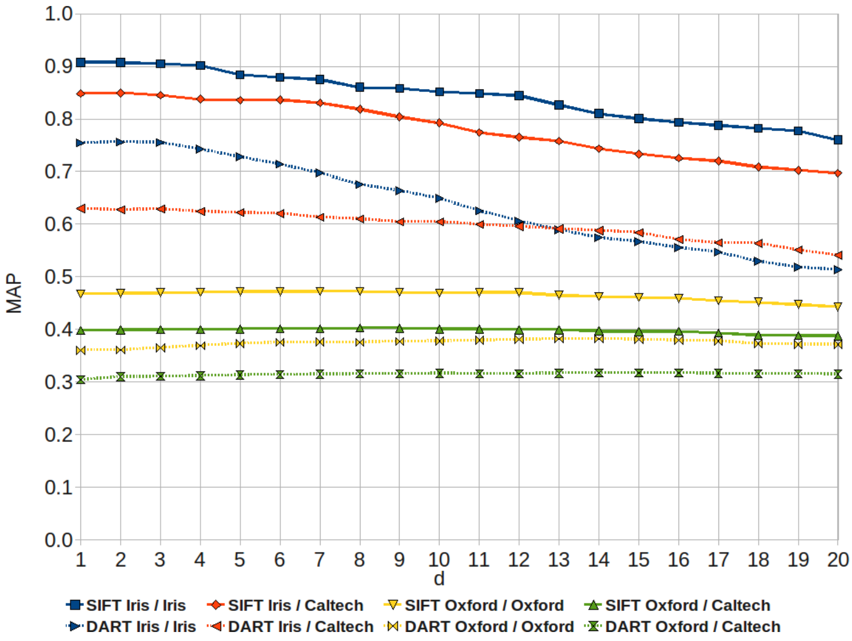
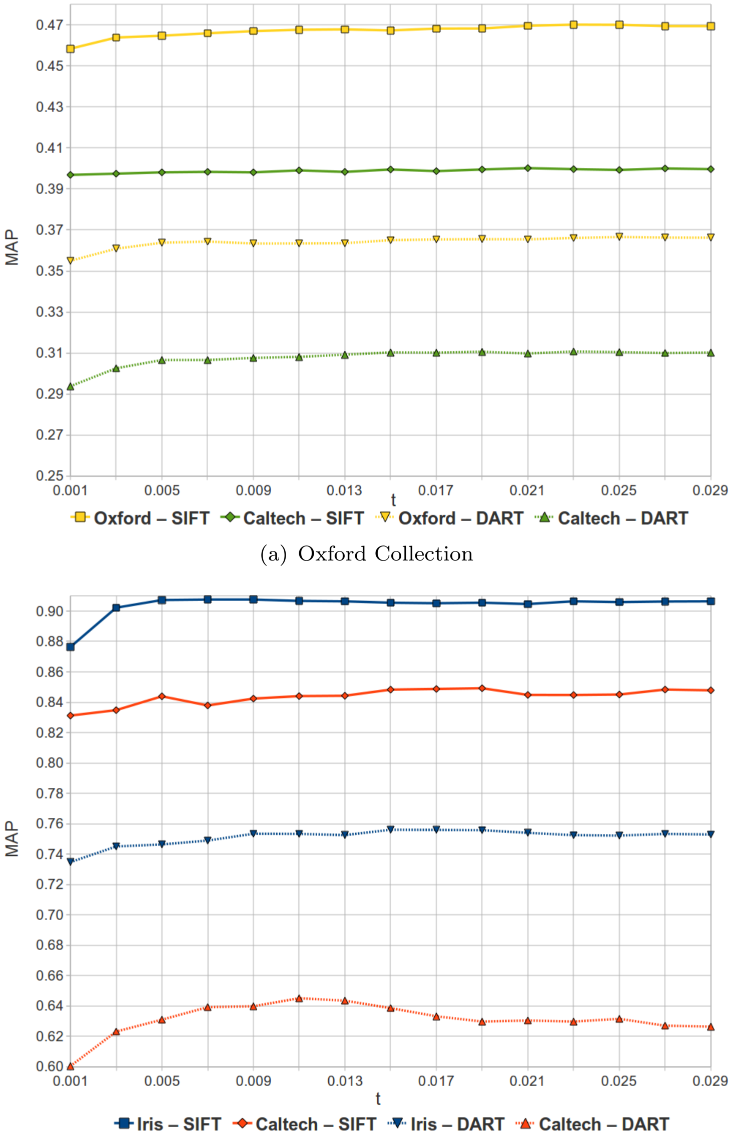


FIGURE 6.6: Optimisation of minimum number of inliers in the consensus set d. First name defines collection, second – origin of the images used for dictionary creation.

It shall be noted that value of parameter t has a significant impact on the discriminative power of the system. Lower values of this parameter will result in an increase of the discriminative power, at the same time implying a decrease of the module's robustness to affine transformations of the matches. Thus, we would like to set the parameter t to the lowest value that ensures high discriminatory power, while not being too strict with small spatial inconsistencies between matches. The results of the optimisation of symmetric error threshold t can be seen in Fig. 6.7. The saturation of the system with Oxford Data Set (Fig. 6.7(a)) can be seen for $t = 0.015$, but relatively high results are observed for even such low values as $t = 0.01$. In the case of the Iris Data Set (Fig. 6.7(b)), the peak value of MAP can be observed around value $t = 0.01$. Those difference between between the evaluated collections can be explained by the fact that the Iris Dataset contains images depicting more difficult affine transformations, whereas the Oxford Dataset provides much simpler transformations. Thus, once again, the Iris Dataset is more sensitive to the change of the system discriminatory power than the Oxford collection. In general, it appears that the best compromise covering majority of cases is provided by $t = 0.01$.



(a) Oxford Collection

(b) Iris Collection

Figure 6.7: Optimisation of symmetric error threshold t. First name denotes descriptor, second – collection evaluated, third – origin of the images used for dictionary creation.

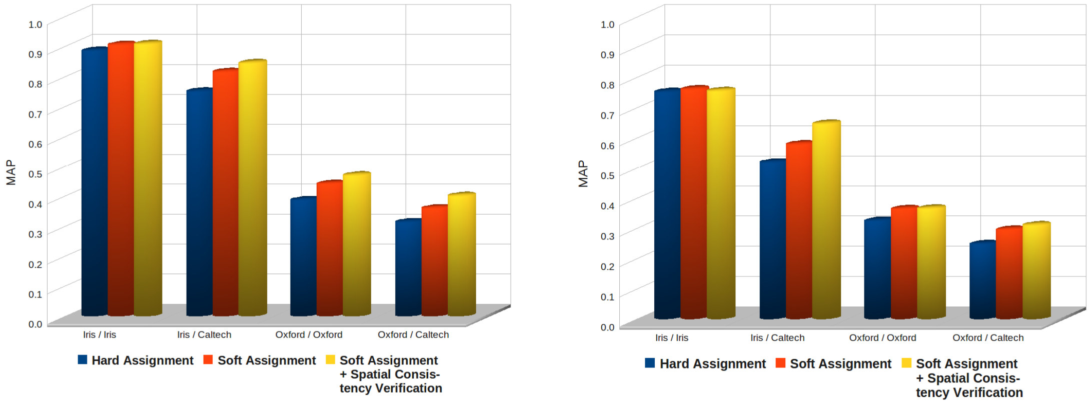
The final results comparison can be seen in Fig. 6.8. For the system with the spatial consistency verification, MAP increases 0.023 on average with respect to the system only with the soft assignment. Most significant increase can be observed for the Iris collection tested with Caltech dictionary. However, an essential improvement can be seen also for the Oxford Dataset (SIFT).

In general, the increase of MAP may appear rather modest. This can be explained by the fact that the coarse pose clustering performed in the previous stage of search is already performing sufficiently well for such small collections. In the cases where there is no improvement (Iris Data Set with Iris Dictionary) it is likely that we reaching the performance saturation. We have confirmed this observation by manually verifying that no false positive matches pass through the spatial consistency check module. The observation suggest also that further improvement is difficult to obtain with the spatial consistency verification. This is mainly due to the fact that

Collection / Dictionary	Original	$+SA$	$+SA+SCV$	$+SA+SCV+T$
Iris / Iris	152.88 ms	190.59 ms	344.83 ms	246.19 ms
Iris / Caltech	215.00 ms	257.20 ms	510.93 ms	328.39 ms

Table 6.1: Average query time comparison for system in original configuration, with soft assignment ($+SA$), with soft assignment combined with spatial consistency verification ($+SA+SCV$) and with soft assignment, spatial consistency verification and thresholding mechanism ($+SA+SCV+T$). Feature extraction time not included. Tests done on machine with Intel Core2 Duo 2.33 GHz and 3 GB RAM.

this mechanism verifies the spatial layout of the existing correspondences and its performance is limited to the number of matches obtained from the earlier stages of search engine.



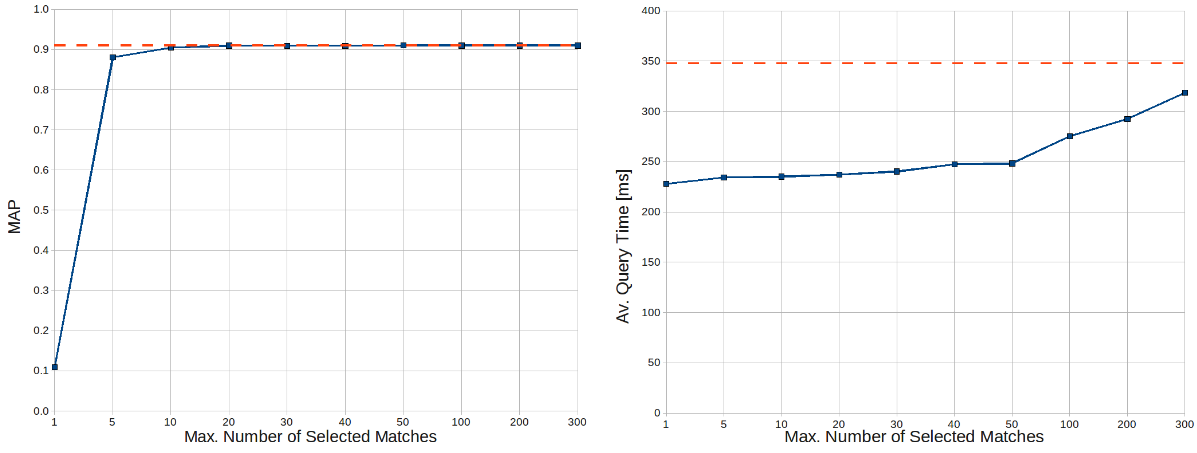
(a) SIFT

(b) DART

Figure 6.8: Final results comparison for hard assignment, soft assignment and spatial consistency verification. In the x-axis first name defines origin of the images used for dictionary creation, second – descriptor.

Computational Cost Analysis

Not surprisingly, the average query time for system with the spatial consistency verification increases with respect to the previous configurations (compare columns 2-4 of Tab. 6.1). This is due to the computational cost introduced by the processing of additional matches. Bearing in mind the close-to-real-time requirements of potential mobile application, we propose additional thresholding mechanism that ensures that only the most significant key-point matches obtained from the initial voting process are passed to the spatial consistency verification module. Specifically, we propose to select a predefined number of correspondences with highest similarity value and pass only them to the verification stage. As presented in Tab. 6.1, it allowed us to significantly reduce computational time. The graphs that show average query time and MAP values depending on the maximal number of initial correspondences passed to the spatial verification module can be seen in Fig. 6.9. According to them, value of $N = 50$ matches was found to offer the best compromise between computational cost and performance quality. We can observe that while selecting only $N = 50$ most important correspondences we obtain significant reduction of computational time with no decrease in MAP value.



(a) MAP (b) Average Query Time

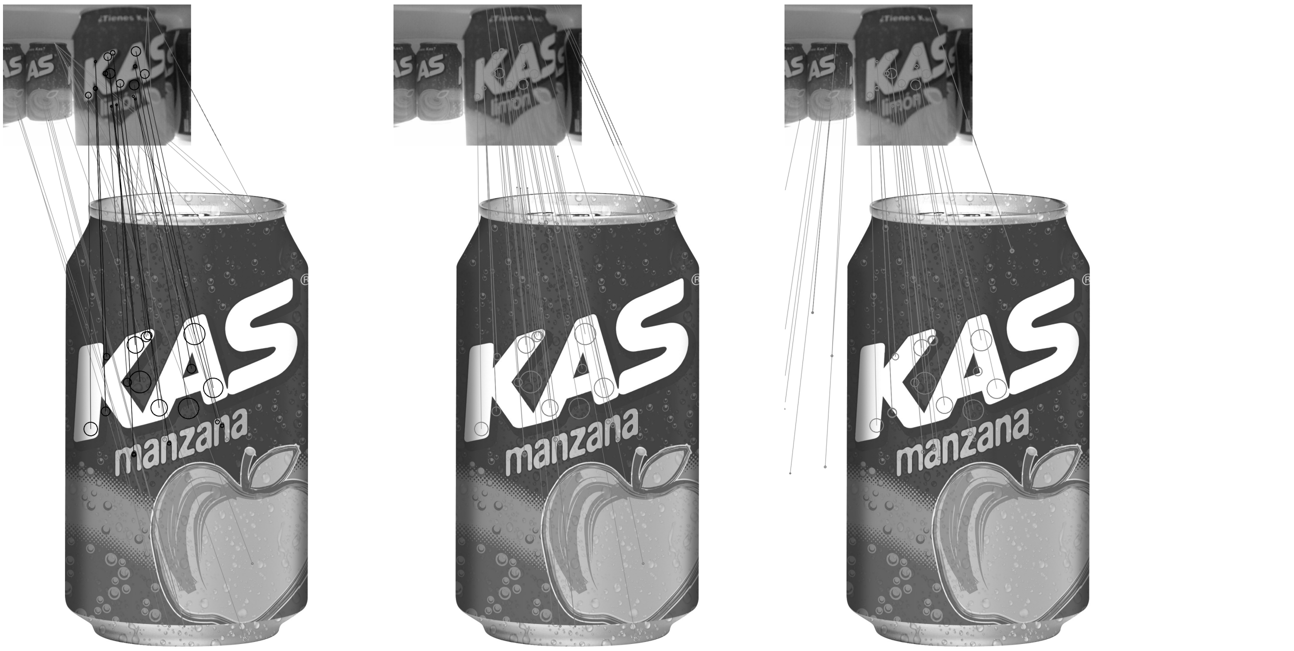
Figure 6.9: MAP values and average query time depending on the number of selected correspondences for the spatial consistency verification (see section 6.5). The dotted line presents the results obtained when all the matches are selected. The analysis done for the Iris Dataset with the Iris Dictionary (SIFT).

6.6 Discussion

Since the spatial consistency verification estimates the transformation between object view in reference and query images, we shall discuss the situation when there are many instances of one object present in the query image. It is an interesting problem, because the transformation model does not account for object replication. In this case, we expect the proposed algorithm to choose only the most salient instance and define the transformation with respect to this particular item. As we can see from an example shown in Fig. 6.10, our solution performs according to expectations and only one instance of object is used for model creation. Thus, only matches with this instance will be verified positively, even though matches with other instances may be correct as well. In case of multiple instances of different objects in queries, the situation is much simpler, since reference images are assumed to depict only one object per frame and there is only one meaningful transformation model per reference image to be found. (see Fig. B.5 in Appendix B).

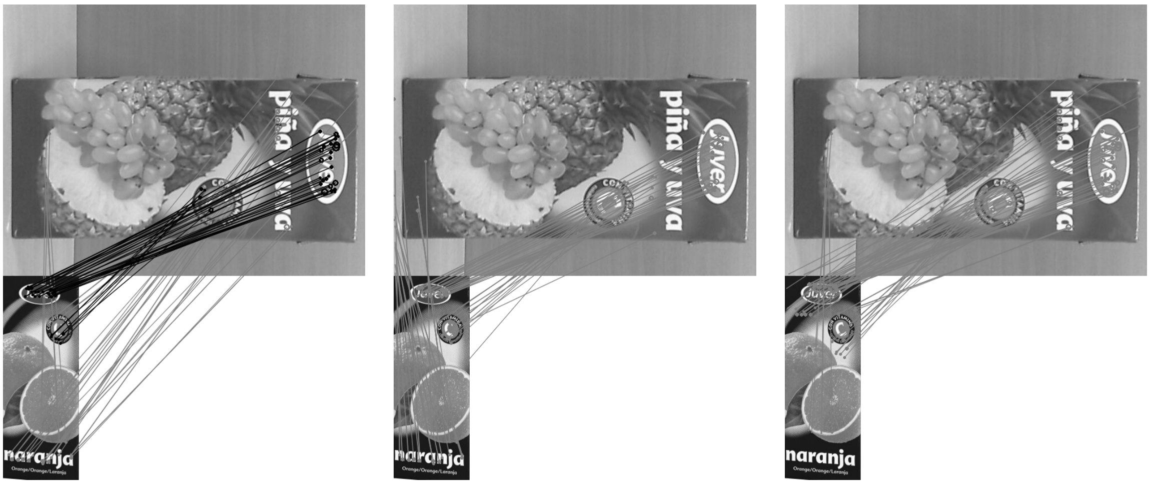
Another interesting challenge is related to the retrieval of objects with the same logotype (see example in Fig. 6.11). In the presented example, correspondences between two logotypes have been verified as spatially consistent. Additionally, matches between the circular symbol have been accepted as consistent, even though the relative location of the symbol with respect to the logotype is slightly different in both images (they are closer in the bottom image). It is due to the fact that we accept matches within certain error threshold t.

The effects of the implemented solution on the example query with retrieved images is shown in Tab. 6.2. In general, scores obtained after the spatial consistency verification are lower than those obtained in the previous stages of the search. However, it is especially important that in the case of the irrelevant image ranked at 6^{th} position, the spatial consistency module puts the score of reference image to a much lower value (2.54). After the spatial consistency verification, the ratio between scores of the last relevant image (5^{th} position) and the first irrelevant image (following position) is much higher than before the verification process (9.5 after spatial verification with respect to 2.5 before). Thus, we can observe that the noise introduced by false matches with the soft assignment is reduced and discriminatory power of



(a) Selected matches. (b) Forward projection. (c) Backward projection

Figure 6.10: Retrieval example with many instances of one object present in the query scene. Query images are displayed above reference images. Projections are done according to the best found transformation model. (a) Selected matches. Inliers and outliers are represented with black and grey colours respectively. (b) Projection of reference key-points into query image. (c) Backward projection of query key-points into reference image.



(a) Selected matches. (b) Forward projection. (c) Backward projection

Figure 6.11: Example retrieval of objects with the same logotypes. Query images are displayed above reference images. Projections are done according to the best found transformation model. (a) Selected matches. Inliers and outliers are represented with black and grey colours respectively. (b) Projection of reference key-points into query image. (c) Backward projection of query key-points into reference image.

the system increases. In other words, we can avoid confusion between relevant and irrelevant

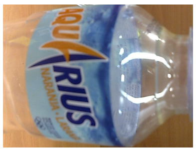
Query Image		Results					
	**Before Spatial Consistency Verification**						
	Score	241.18	58.41	57.46	52.5	35.2	14.04
	After Spatial Consistency Verification						
	Score	216.14	46.63	31.13	30.44	24.13	2.54

Table 6.2: Results of image retrieval with and without spatial consistency verification. 6 top ranked retrieved images are presented.

images when analysing the scores of retrieved images.

6.7 Conclusions

The above results clearly show that the spatial consistency verification improves consistently performance quality of visual search engine. The improvement varies depending on the collection and dictionary employed.

MAP value increases 0.023 on average with respect to the system with soft assignment only. This quantitative improvement may appear rather modest. However, it confirms the fact that the initial pose clustering implemented in the original approach performs relatively good with the evaluated data sets.

The spatial consistency verification has been proven to perform well with the visual search engine. Relatively modest MAP improvement may be explained by the fact that the initial clustering in the reduced pose space from the original approach was already sufficiently powerful to ensure low level of false positives for the available collections. Nonetheless, exhaustive visual inspections confirmed that in the case of the original approach, we could observe certain amount of false positives correspondences. Those matches have been successfully removed after the introduction of the spatial consistency verification. Therefore, we can conclude that the spatial consistency verification shall play important role when larger collections with much higher number of false positives will be evaluated.

6.7.1 Further work

As discussed earlier in section 6.3.3, the transformation model employed in the proposed spatial consistency verification algorithm is relatively simple and is not ideal for complex affine transformations. However, it is a common scenario, especially in the outdoor recognition applications, that analysed objects are captured from different angles and perspectives. Thus, this simplified model may not be sufficient for recognition of such objects.

On the other hand, computational cost of complex affine transformation models is relatively high. Since it involves estimation of additional parameters, its requirements may exceed capabilities of mobile devices.

Taking into account all the above observations, possible extension of the solution could include additional step for Alg. 6.2. It could comprise *model refinement* using more complex homography or affine transformation and *search for inliers* that fit the newly generated complex model. This step would be performed only once with the final consensus set. Furthermore, calculation of the model may include more than a single pair of corresponding key-points, since the consensus set would contain only matches whose spatial consistency has been already verified. After refining the model, search for inliers would be performed. While iterating over matches not included in the consensus set, we would verify if they fit refined model and could be added to the set. This way we would increase robustness of our solution without substantial increase of computational cost.

Moreover, it should be noted that at the moment only a constant number of reference images is spatially verified. Another improvement could focus on extending existing solution with dynamic thresholding [29] mechanism. It would be used to define the number of reference images passed to the final spatial consistency verification stage, depending on the distribution of scores. This may increase the robustness of the system and may further reduce the computational complexity.

Chapter 7

Final Results

In this chapter final results of the visual search engine with all the proposed extensions are presented. Furthermore, the performance of the system before and after introduction of improvements is compared and discussed in details.

7.1 Introduction

In order to provide extensive analysis of the results of the visual search engine performance, we will firstly discuss overall results for all collections with different types of dictionaries. We will compare them, using Mean Average Precision (MAP), since it is the most informative and stable evaluation measure and it is widely accepted in information retrieval community (see chapter 4). Afterwards, we will discuss the most interesting detailed results obtained for the particular data sets, using additional evaluation measures.

In the remainder of this thesis, we will refer to the system configuration without the soft assignment and the spatial consistency verification, introduced in chapters 5 and 6, as the *original* configuration. The configuration that will include the soft assignment and the spatial consistency verification mechanism will be referred to as *extended* configuration.

Since one of the main motivations of this thesis is reducing the influence of dictionary quality on retrieval outcome, we will present the final results for different types of dictionaries used. Results will be ordered according to the dictionary type in the following way:

1. Results obtained with a dictionary created using reference images from the evaluated collection.

2. Results obtained with a dictionary created using reference images that represent similar application scenario as the evaluated collection.

3. Results obtained with a dictionary created using reference images from the generic data set – Caltech 101 4.4.

7.2 MAP Comparison

As mentioned above, in this section we will compare MAP results for all the collections described in chapter 4 before and after the introduction of the improvements proposed in this thesis.

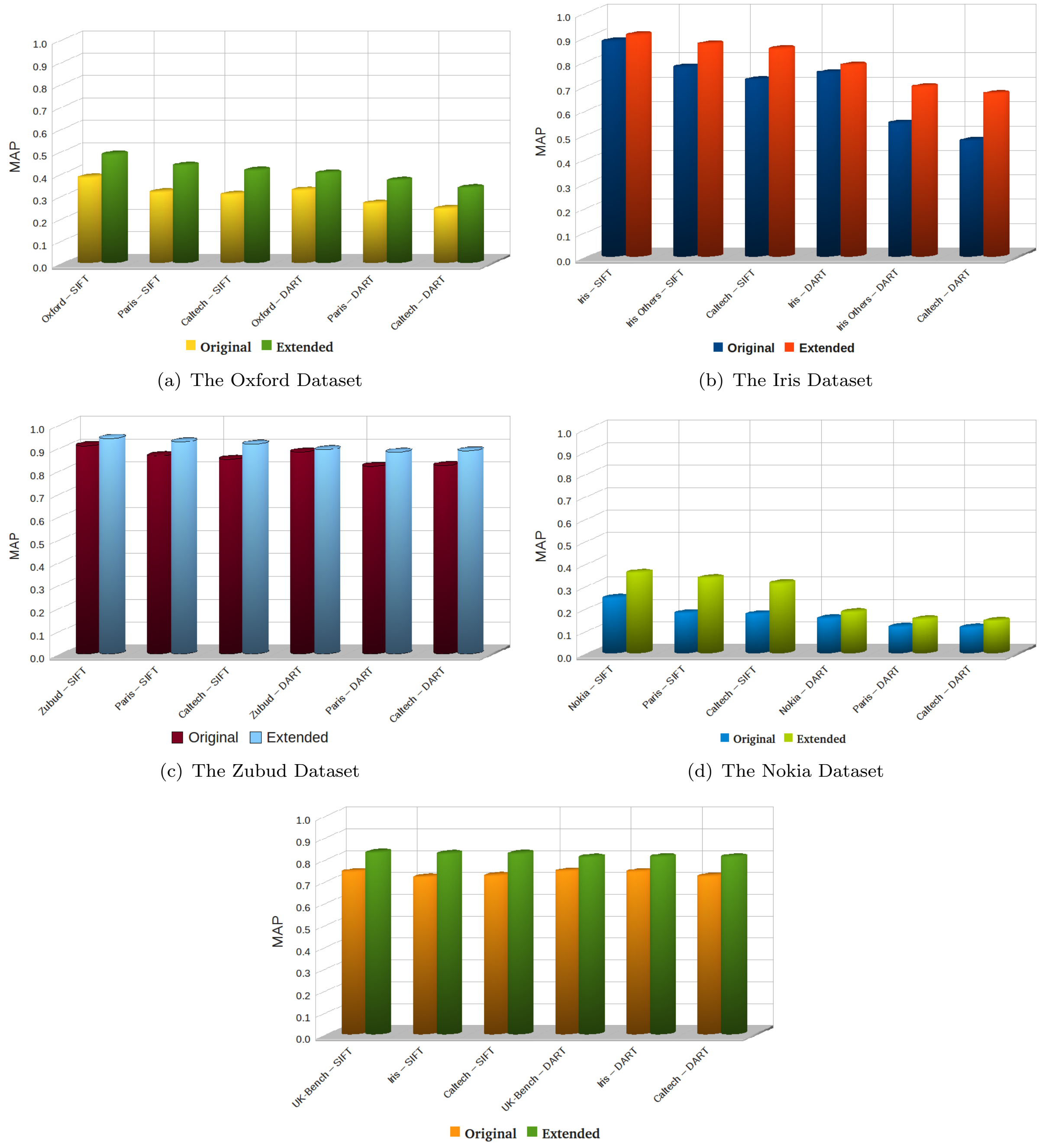
Fig. 7.1 shows the comparison of the MAP results with different types of dictionaries used. The detailed numerical results can be found in Appendix B in Tab. B.2. We can see that the MAP increase varies from 0.01 to 0.19 depending on the collection and dictionary used. Thus, we can state that modifications introduced within this thesis improve performance quality of visual search engine. Furthermore, we can see that the results obtained with the extended configuration are less dependent on the dictionary type employed. For instance, in Fig. 7.1(b) we can see that the search engine performance in the original configuration deteriorated highly when changing from the dictionary created from the reference images from the evaluated dataset to the generic dictionary. After the introduction of the improvements proposed in this thesis the differences are much smaller. In fact, the difference between the results obtained for the dictionary created from the "*Iris others*" dataset (same application) is almost unnoticeable. This is due to the fact that soft assignment combined with spatial consistency verification reduces the influence of quantisation effects and, at the same time, eliminates additional noise introduced by soft assignment. This can be confirmed by similar observations for other collections.

7.3 The Oxford Dataset

Fig. 7.2 shows the detailed results obtained for the Oxford Dataset. We can see a moderate increase in the percentage of queries for which the perfect match is retrieved at the first position (Fig. 7.2(a)). We can also observe relatively modest increment of the query percentage where at least one perfect match was retrieved within the first five images (Fig. 7.2(b)). It is due to the fact that by the introduction of the soft assignment and the spatial consistency verification we have increased the recall. The observed improvement appears to be smaller then corresponding increase in MAP (see Fig. 7.1(a)). This could be explained by the fact that MAP captures better the differences at further positions on the retrieved images' lists, whereas the perfect match measures presented here provide us only with information about the changes concerning a very small part of that list. Furthermore, the percentage of queries for which any perfect match image was retrieved within the five top ranked images is relatively high and equals approximately 75% (49 out of 55 queries). The six remaining queries (see Fig. 7.3) are relatively difficult, *e.g.* they contain affine transformations of objects or they depict objects in very poor lighting conditions. These kind of problems clearly cannot be improved by the proposed extensions but should be tackled by modification of used low level features.

When analysing the graph of relation between average scores of the perfect match reference image and the first irrelevant image (Fig. 7.2(c)), we can see that it increases greatly in the case of the system with the proposed improvements with respect to the original configuration. This can be explained by the fact that the spatial consistency verification increases the discriminatory power and the scores of irrelevant elements are reduced.

Fig. 7.4 displays the precision – recall curves for both, SIFT and DART. The original configuration is presented with dashed line, whereas the extended one – with continuous line. We can observe that after the introduction of the improvements we obtain significantly higher precision for a given recall.



(a) The Oxford Dataset

(b) The Iris Dataset

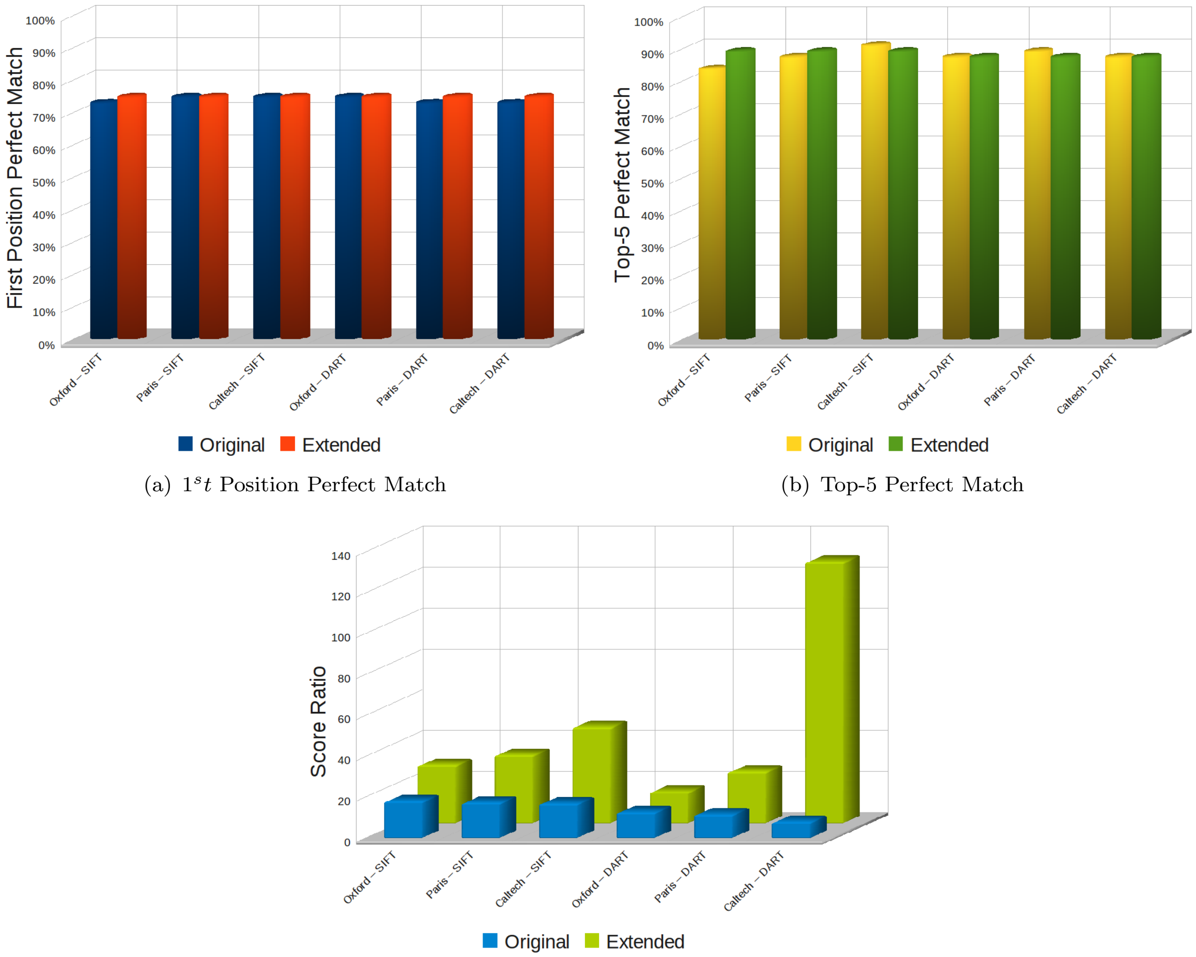
(c) The Zubud Dataset

(d) The Nokia Dataset

(e) UK-Bench Dataset

FIGURE 7.1: The overall results of the visual search engine evaluation with different collections before and after the introduction of the improvements proposed in this thesis. In the x-axis the first name denotes the origin of the images used for the dictionary creation and the second name denotes descriptor.

Finally, Fig. 7.5 shows average precisions obtained for all the Oxford Dataset queries. As we can see, we obtain improvement in precision for most of the queries and the precision of the remaining ones does not change. For instance, for the *christ_church_000179* query, the improvement is significant – the average precision increases from 0.16 to 0.63. This clearly confirms that by the introduction of the soft assignment and the spatial verification mechanism we have improved the overall performance of the system.



(a) 1^{s}t Position Perfect Match

(b) Top-5 Perfect Match

(c) First Perfect Match to First Irrelevant Image – score ratio

FIGURE 7.2: Oxford results. In the x-axis first name denotes origin of the images used for dictionary creation and the second name denotes descriptor.

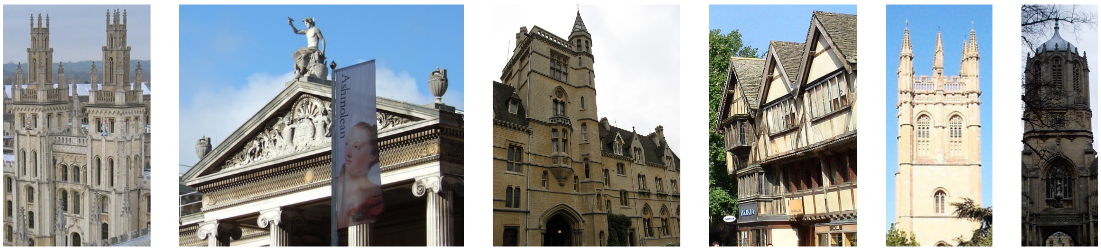
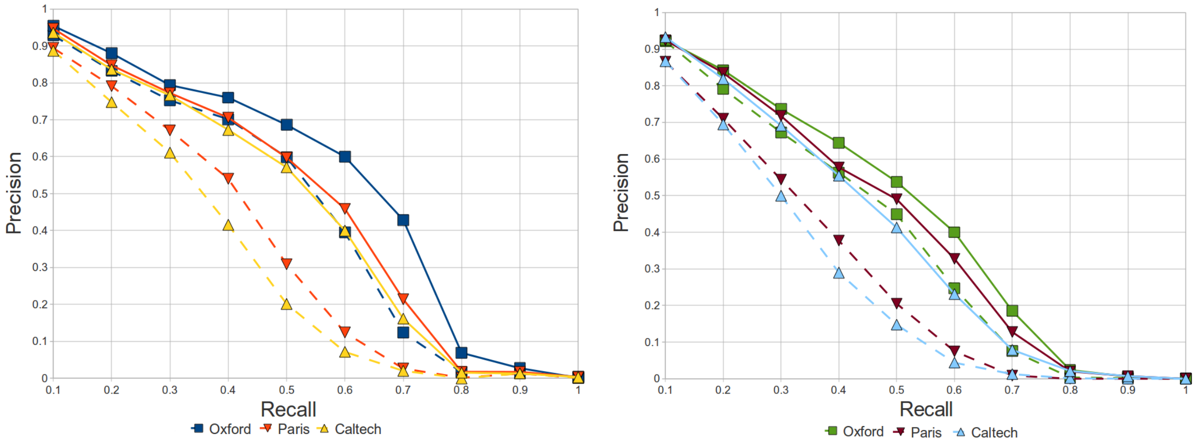


FIGURE 7.3: The queries from the Oxford Dataset for which retrieved Perfect Match is ranked out of top five images.

The observed improvements measured in MAP are comparable to the improvements obtained in [8] under similar conditions. It shall be noted, however, that due to the differences in the search engine configurations, such as different dictionary size and low level features employed, the exact comparison would not be meaningful.



(a) SIFT (b) DART

Figure 7.4: Precision – Recall curves for the Oxford Dataset. Lines showing the results obtained with the original configuration are dashed, while lines showing the results obtained with the extended configuration are continuous.

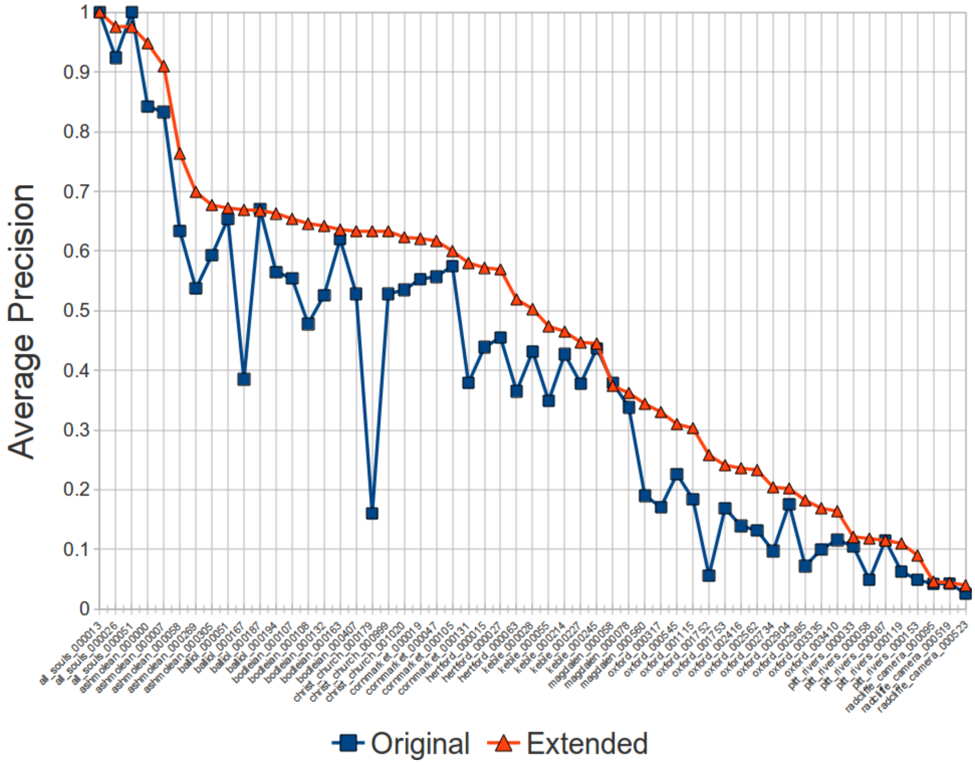


Figure 7.5: Average precision computed for the set of the Oxford Dataset queries. In the x-axis names of the queries can be found. Presented results were obtained for the SIFT descriptor with the Oxford dictionary.

7.4 The ZuBuD Dataset

Since we have found the ZuBuD Dataset to be relatively easy for image matching and detailed extensive evaluation results would not provide additional information, in this section we will present only the most interesting results for this collection. In Fig. 7.6 we can see the average scores' relations between the top ranked Perfect Match reference images and the top ranked irrelevant images retrieved. We can observe that the proposed extensions improve the ratio significantly. Interestingly, it is higher for the results obtained with the dictionaries created from the images of collections different than the evaluated one. This can be explained by the fact that the spatial consistency verification increases discriminatory power much more in those cases and irrelevant images obtain scores close to zero.

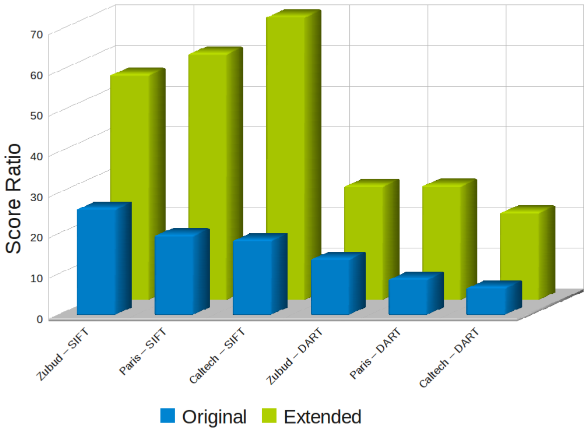


FIGURE 7.6: The ZuBud results: First Perfect Match to First Irrelevant Image – score ratio. In the x-axis first name denotes origin of the images used for dictionary creation and the second name denotes descriptor.

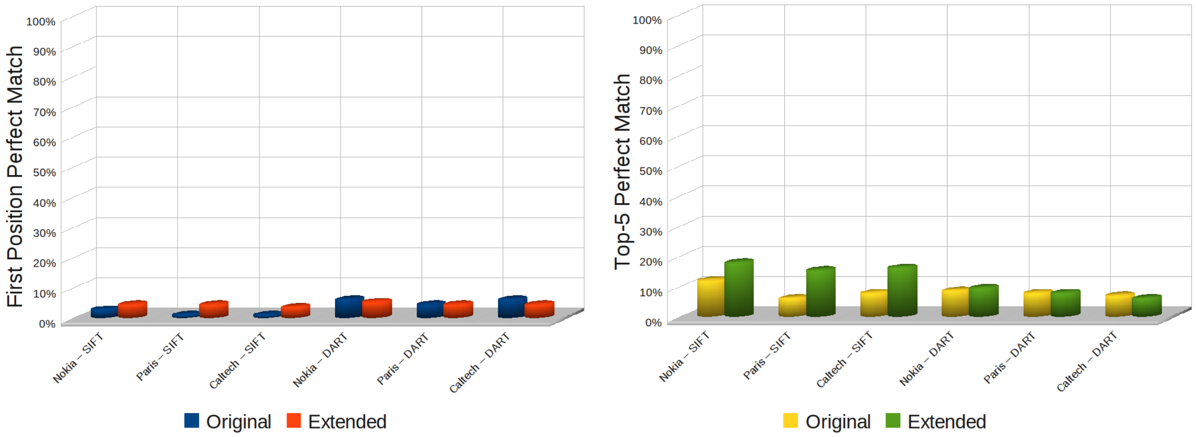
7.5 The Nokia Dataset

On the contrary to the previously discussed ZuBuD Dataset, the Nokia Dataset has been found relatively difficult. This may be explained by the fact this collection was created to be used for testing of retrieval system for mobile phones that matched images against a database of location-tagged images. Therefore, in [1], the localisation data was included in the recognition process and used to limit the number of images that had to be analysed. Our testing conditions of the visual search engine are much more challenging, because we do not use the localisation information. Thus, obtained results are rather modest and the Nokia collection is assessed as difficult.

Fig. 7.7 shows detailed results obtained for the Nokia Dataset. As we can observe from graphs 7.7(a) and 7.7(b), the introduced improvements increase the position of the first perfect match image retrieved for more queries – at least one perfect match reference image was among the top five images retrieved.

7.6 The Iris Dataset

Fig. 7.8 shows the detailed results obtained for the Iris Dataset. We can observe that the measures based on the perfect match reference images improve when the system is working with the soft assignment and the spatial consistency verification with respect to the original configuration. Relatively higher increases can be observed for dictionaries created from Iris Others and Caltech 101 collections. In this case this can be explained that with high quality dictionaries created from the evaluated images the system obtains extremely good results, even without the



(a) 1^st Position Perfect Match

(b) Top-5 Perfect Match

FIGURE 7.7: The Nokia results. In the x-axis first name denotes origin of the images used for dictionary creation and the second name denotes descriptor.

proposed extensions. Furthermore, the results obtained with the extended configuration are not affected by different types of dictionary as much as in the case of original configuration.

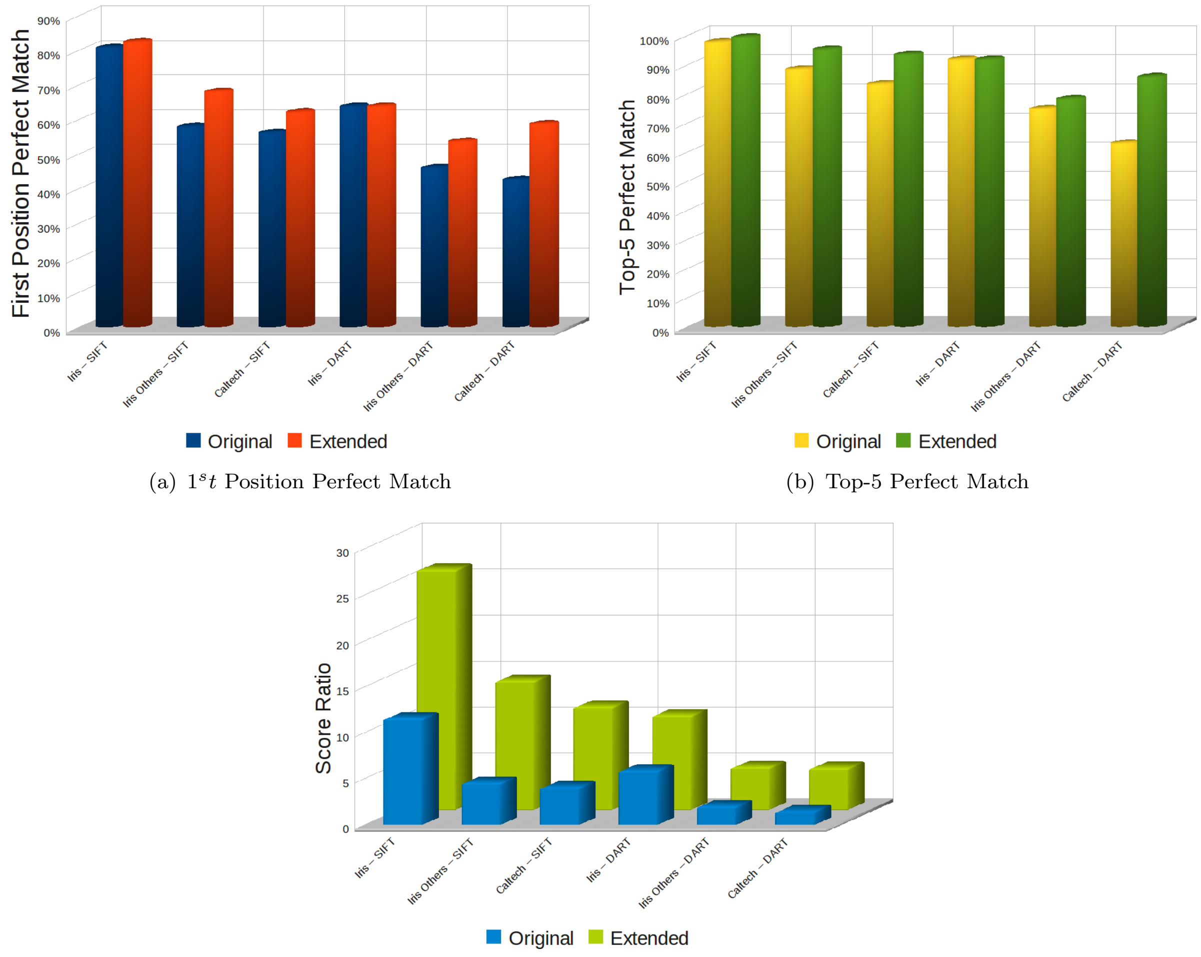
Moreover, analysing precision – recall curve in Fig. 7.9, we can observe that for the same level of recall we obtain higher precision for system with extensions. The difference is greater for DART. Once again, it may be explained with increase of discriminatory power provided by spatial consistency check and the reduction of quantisation effects through soft assignment mechanism that results in higher recall. Moreover, we can see that performance quality of system with Iris Others and Caltech dictionaries used is similar. This also confirms the earlier observation that with the extended configuration we significantly reduce the impact of dictionary quality on the system performance.

7.7 Discussion

We can observe a consistent improvement of the performance obtained by introducing mechanisms proposed in this thesis. The influence of these mechanisms has been evaluated using various collections and with several dictionaries.

Overall results confirm that we have indeed reduced the impact of dictionary quality on system performance. Obtained MAP results clearly show that the performance of the system with dictionaries created from the generic reference images and from the reference images that represent similar application scenario as the evaluated collection is comparable. Despite the fact that the results obtained for the dictionaries created from the evaluated dataset are still better than those obtained for other dictionaries, we can observe that the proposed extensions significantly reduce the gap between those results. This is confirmed by the analysis of the precision – recall curves that clearly show full potential of proposed solutions.

Moreover, exhaustive visual inspection of the results suggest that the system with extended configuration reached a point where further improvements may be obtained by the means of improvements of the used local features.

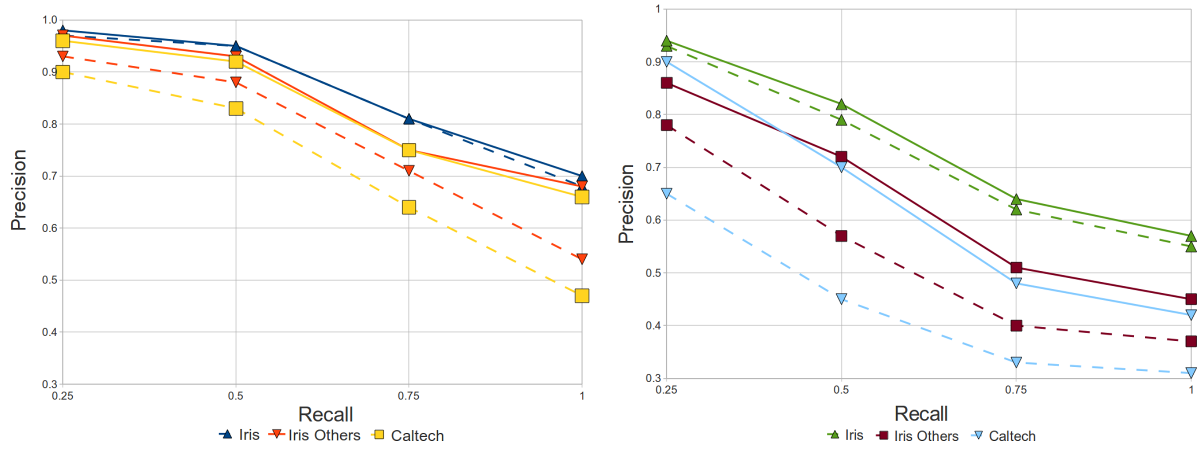


(a) 1^{s}t Position Perfect Match

(b) Top-5 Perfect Match

(c) First Perfect Match to First Irrelevant Image – score ratio

FIGURE 7.8: The Iris results. In the x-axis first name defines origin of the images used for dictionary creation, second – descriptor.



(a) SIFT

(b) DART

FIGURE 7.9: The Precision – Recall curves for Iris Dataset. Lines showing the results obtained with the original configuration are dashed, while lines showing the results obtained with the extended configuration are continuous.

Chapter 8

Conclusions and Outlook

This chapter concludes the work done within this thesis and discusses further research directions and additional amendments.

8.1 Conclusions

In this thesis we have investigated several problems concerning visual search systems, focusing on alleviating several limitations of the visual search engine developed at Telefónica I+D Barcelona. Building on this state of the art image retrieval system, we have proposed a set of extensions that tackle the main challenges of reliable visual search: improvement of robustness of the detection of similarities between images and increase of discriminatory power. The proposed extensions have been developed in the context of matching outdoor scenes for improved geo-localisation. While implementing proposed solutions, we have not only focused on improving quality of performance, but we have also borne in mind potential computational cost requirements of mobile applications.

Work done within this thesis aimed to advance visual search engine so it can be employed in high precision geo-localisation application. Thus we have concentrated on increasing reliability and precision of the system for collections representing outdoor scenes. Furthermore, we have focused on improving the performance of the engine independently of the quality of the visual word vocabulary used. More precisely, we successfully extended the existing solution so the differences between the performance quality using various dictionaries are minimised.

The more detailed conclusions regarding the main contributions of this thesis can be summarized as follows:

Evaluation Framework

In chapter 4, in order to provide exhaustive evaluation platform, we have adapted existing assessment methods and extend them with the measures inspired by the subjective assessment of user. As a result, we have obtained a complete evaluation framework for image content retrieval system. It is capable of assessing the performance of visual search engine with five different collections, using six different evaluation measures.

We have extensively employed the evaluation framework, while developing and assessing the influence of various proposed extensions. Not only has it enabled us to understand the behaviour of the implemented solutions, but it has also answered significant questions regarding incorporation of those solutions into the search engine. It has been observed that in the context of this work the Mean Average Precision (MAP) is the most representative measure as it consistently captures different aspects of the performance quality of the retrieval system.

Furthermore, in order to obtain meaningful results, we have gathered and organised several evaluation collections. Since they represent various application scenarios, we were able to analyse in detail the performance of the system in various potential applications.

Soft Assignment

As it has been shown in chapter 5, we have successfully adapted the soft assignment method described by J. Philbin in [8] to the Telefónica I+D visual search engine. Since it was originally proposed for image retrieval system based on *tf-idf* weighting scheme, we have designed and implemented several additional elements, *e.g.* weight fusion scheme, that enabled incorporation of the soft assignment into image content retrieval system of Telefónica I+D.

In order to analyse in detail the behaviour of soft assignment, we have developed a set of novel experiments. They provided us with significant insights into the proposed solution. Moreover, it enabled us to understand the influence of different variants and configurations of soft assignment and draw constructive conclusions concerning the integration of this mechanism into the visual search engine.

Finally, after the incorporation of the solution into the system, the extensive evaluation confirmed that the soft assignment mechanism enabled us to capture the similarities between the features in a much more reliable way. As a result, we have obtained significant improvement of the performance quality in terms of various assessment methods.

Spatial Consistency Verification

As described in chapter 6 of this thesis, following the solutions discussed in literature, we have proposed and developed an additional post-processing stage that verifies the spatial layout of correspondences between the images' local features. The solution increases the discriminative power of the matching process and reduce the possibility of returning false positive matches. The proposed solution relies on RANSAC inspired algorithm estimating transformation between query and reference images. We have extended the solution described in the literature and introduced an extension that builds up on the weighting scheme implemented in the original version of visual search engine.

After the integration of the proposed solution, we have fine tuned and validated the behaviour of proposed mechanism. The quality of the proposed approach was evaluated thoroughly. The results confirmed that, thanks to the implemented solution, the discriminative power of visual search engine increases and quality of system performance improves.

8.2 Future Work

Several perspectives for extension of the work presented in this thesis seem worth investigating. They are briefly described and discussed below.

First of all, it shall be noted that the optimisations of the parameters of the spatial consistency verification and soft assignment were done independently. It could be interesting to see if the joint optimisation of those two modules gave the same values of the optimal parameters. For example, in chapter 5 we have observed that increasing the number of visual words per key-point leads to the increase of number of the discovered correct matches (see Fig. 5.10(a)), but it also introduces more noise (see Fig. 5.10(b)). We may expect that since the spatial consistency verification increases the discriminative power, the noise introduced by additional visual words assigned to a key-point could be eliminated. At the same time, if the additional correct matches are spatially coherent with the discovered model, they may improve the quality of the search.

Another opportunity for a further research could involve introduction of different types of low level feature detectors used with the visual search engine. This is motivated by the fact that the visual inspections of the results obtained after the integration of the proposed solutions suggest that for the features used in the current approach, namely SIFT and DART, the system is approaching a saturation point. More precisely, the proposed solutions were not able to provide better performance due to the limitations of the currently employed detectors. In the literature, several other detectors are proposed, *e.g.* the Hessian-Affine detector [8] or one of the most recent affine low level feature – ASIFT [35].

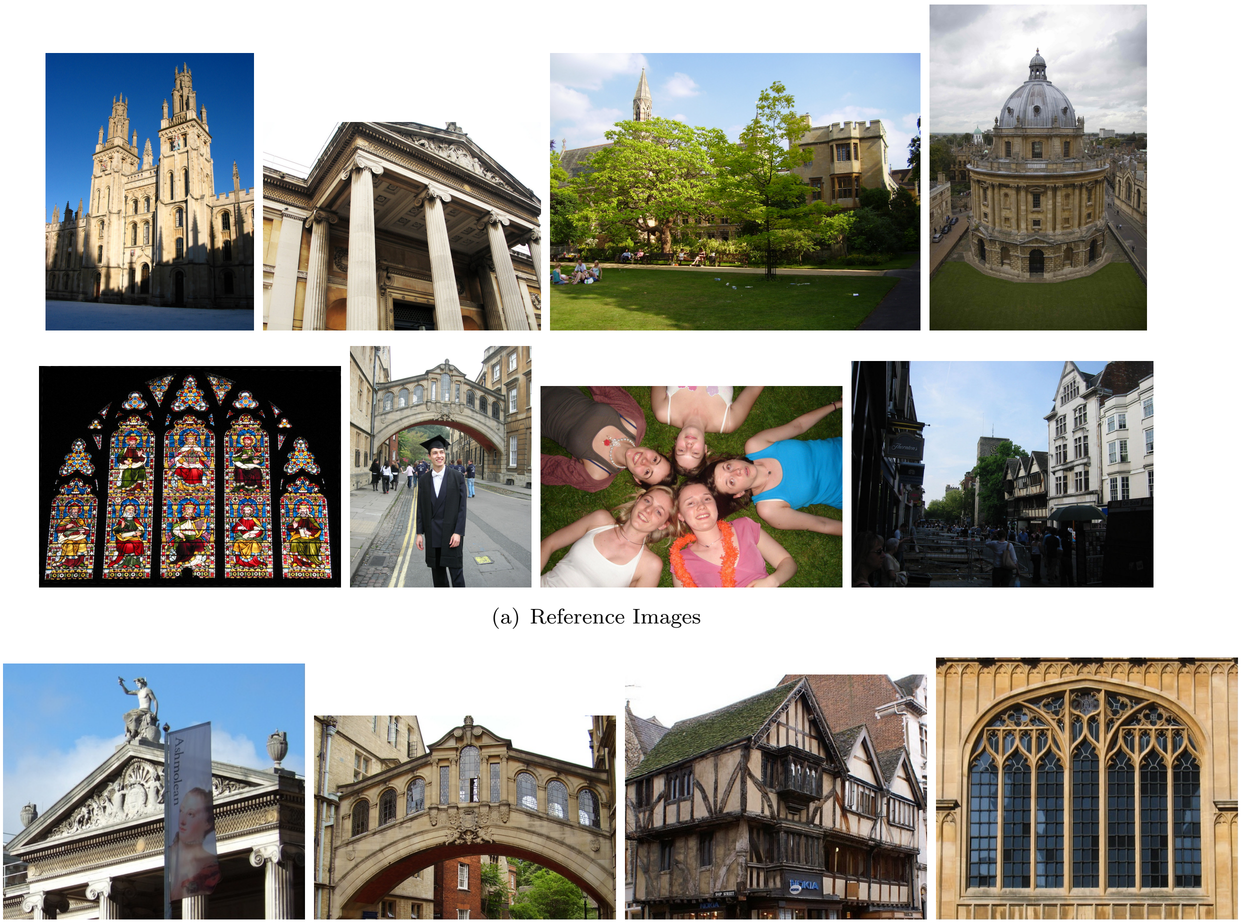
Although the transformation model used in the proposed spatial consistency verification module seems to perform sufficiently well with the local features such as SIFT and DART, more complex models could be needed when different types of affine local features are used. Specifically, a potential perspective of research could involve extending the spatial consistency verification with more complex affine transformation models that could fully benefit from the advantages of those affine features.

Final Word

In summary, closing the circle to the introductory statements of this thesis, we proposed several extensions that improve the performance quality of the visual search engine developed at Telefónica I+D. Moreover, we advance the existing image retrieval system so it could be used for high precision geo-localisation. We believe that all the challenges tackled in this thesis will provide an interesting opportunity for further research.

Appendix A

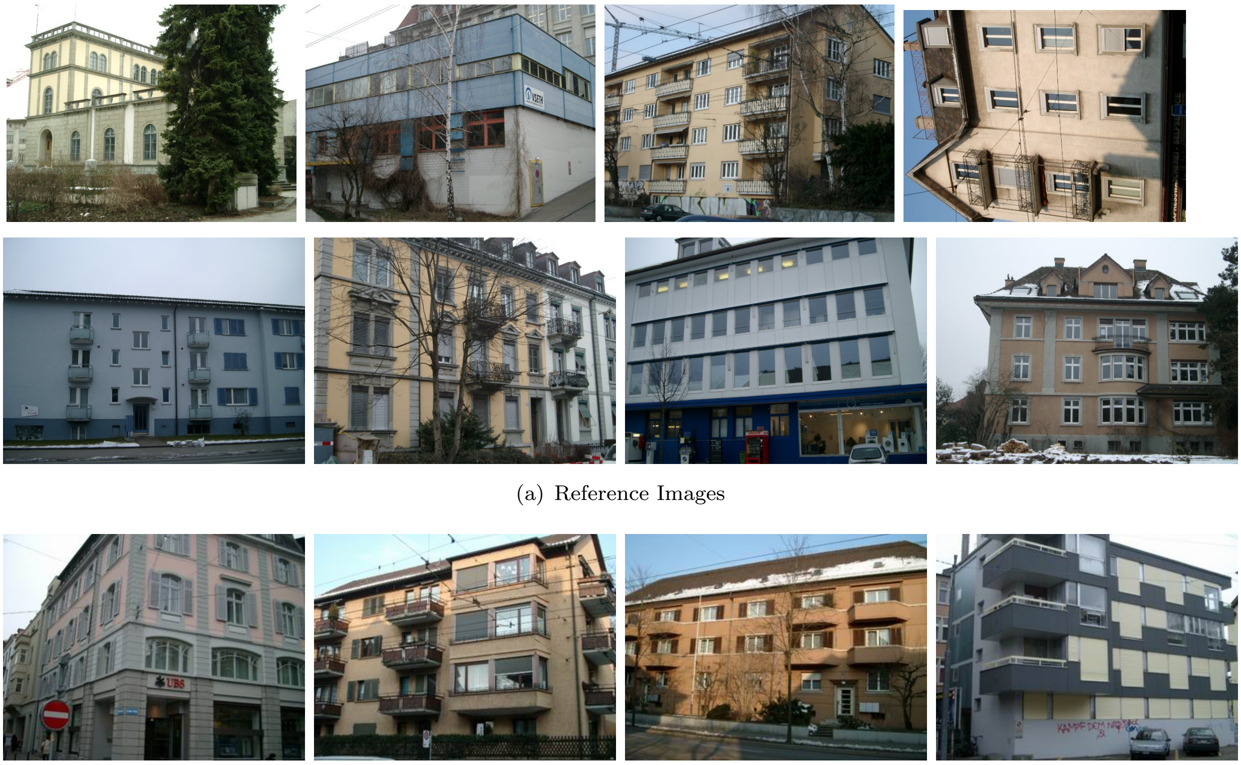
Evaluation Datasets



(a) Reference Images

(b) Query Images

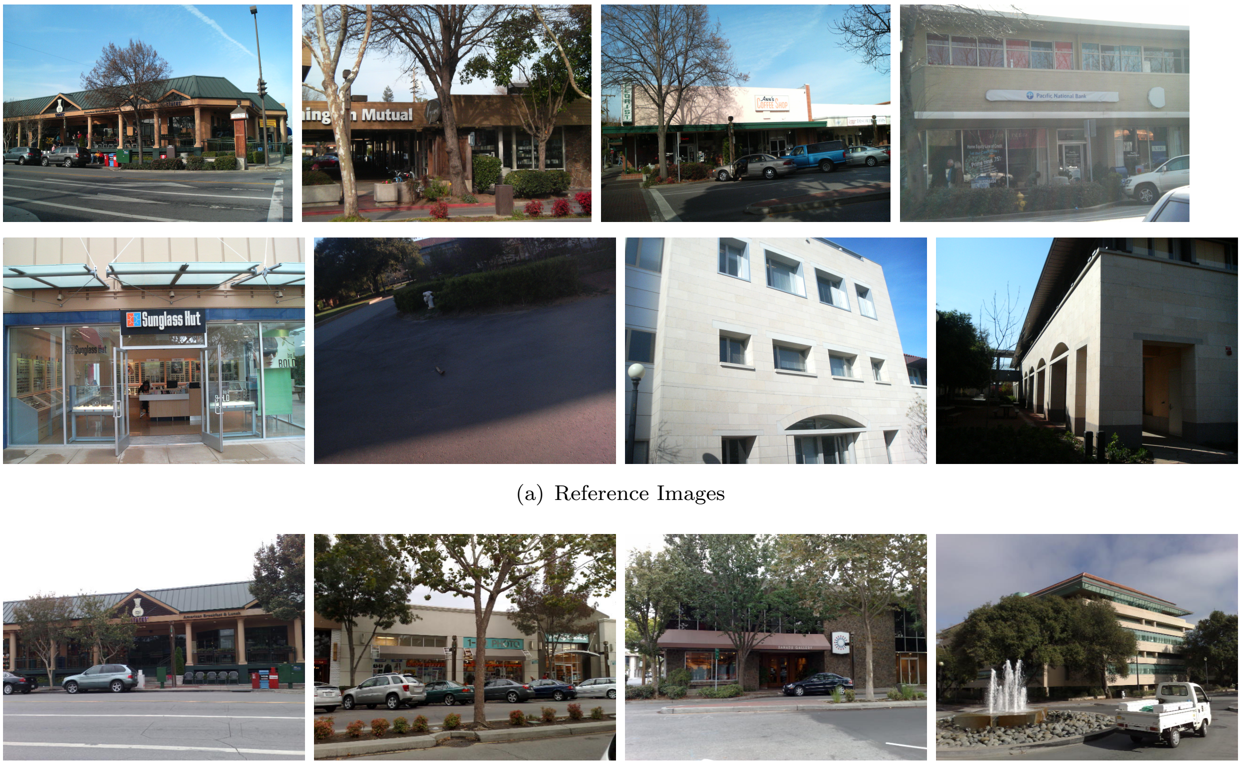
FIGURE A.1: The selected images from the Oxford Dataset



(a) Reference Images

(b) Query Images

FIGURE A.2: The selected images from the ZuBuD Dataset



(a) Reference Images

(b) Query Images

FIGURE A.3: The selected images from the Nokia Dataset

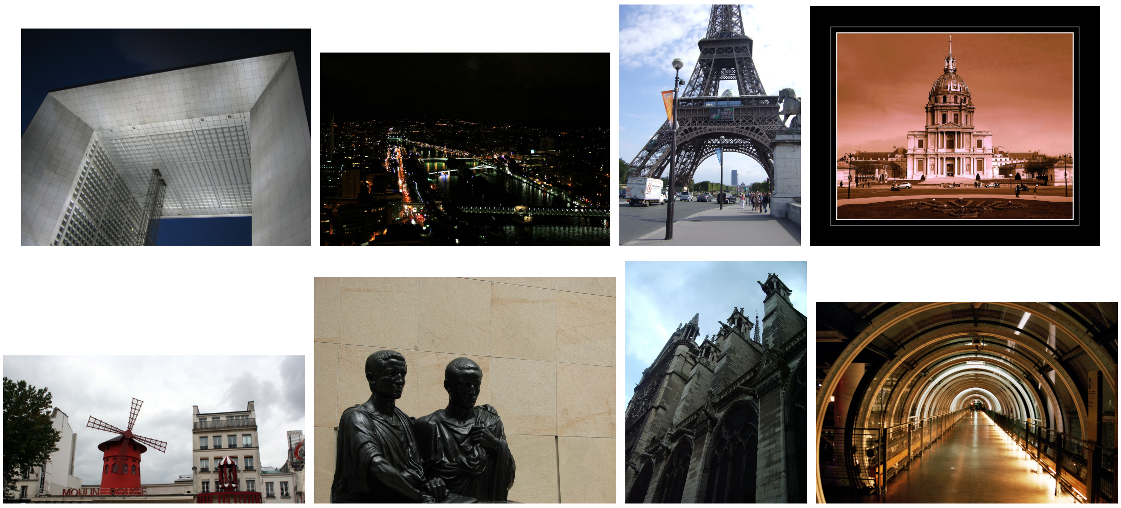


FIGURE A.4: The selected images from the Paris Dataset

(a) Reference Images

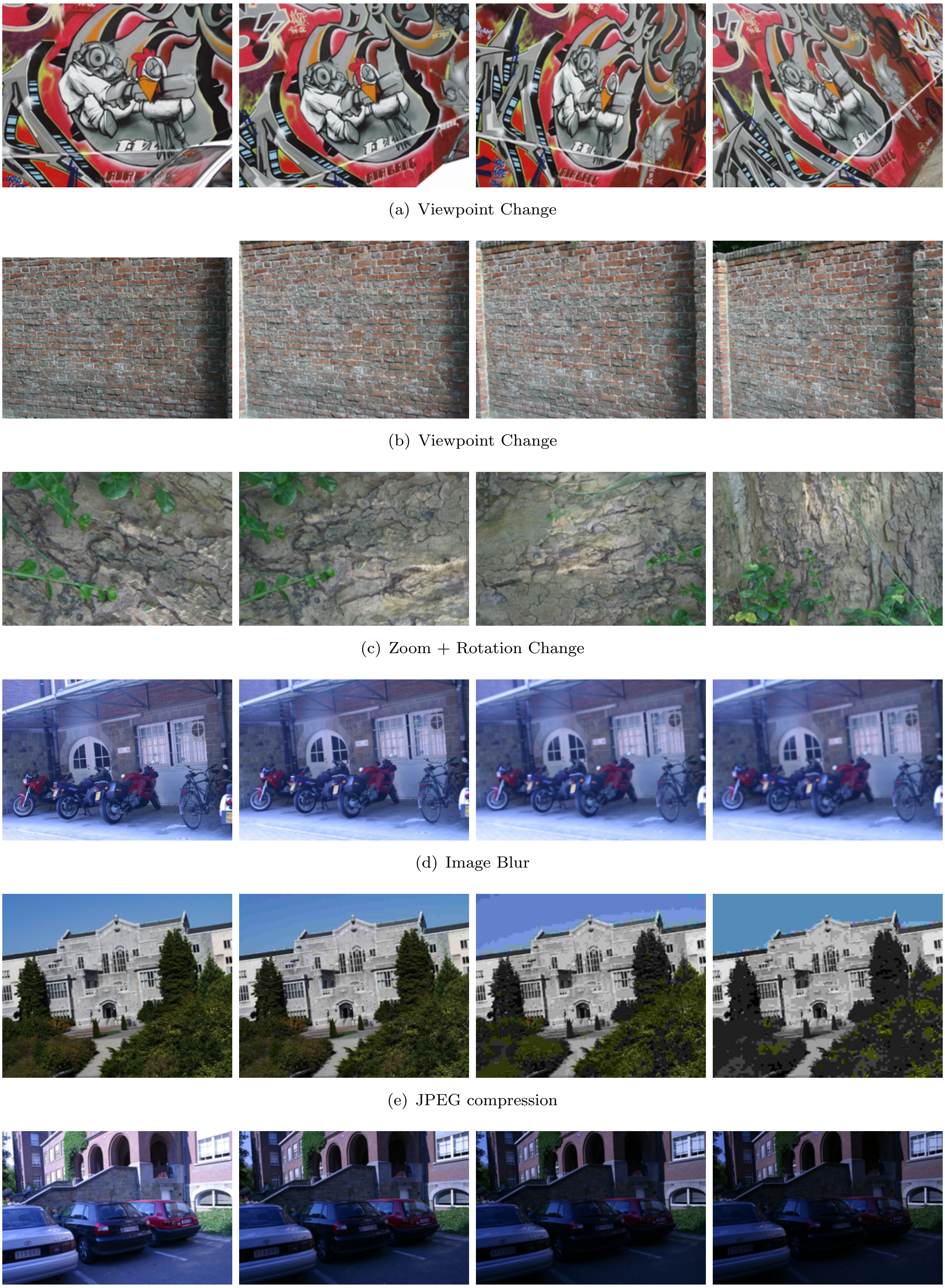
(b) Query Images

FIGURE A.5: The selected images from the Iris Dataset

(a) Reference Images

Figure A.6: The selected images from the UK-Bench Dataset. Query images are selected from reference images.

Figure A.7: The selected images from the Caltech Dataset.



(a) Viewpoint Change

(b) Viewpoint Change

(c) Zoom + Rotation Change

(d) Image Blur

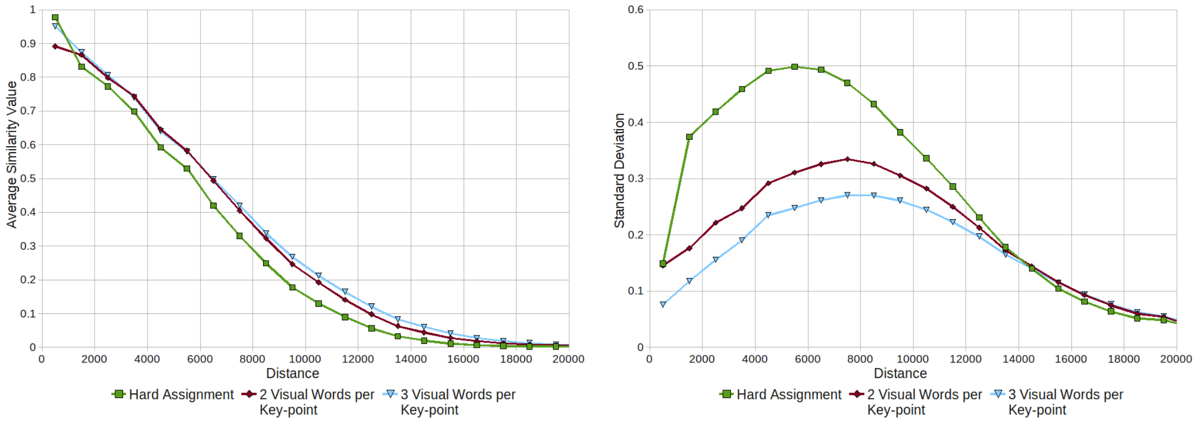
(e) JPEG compression

(f) Illumination Change

Figure A.8: The selected images from the "Mikolaczyk" Dataset

Appendix B

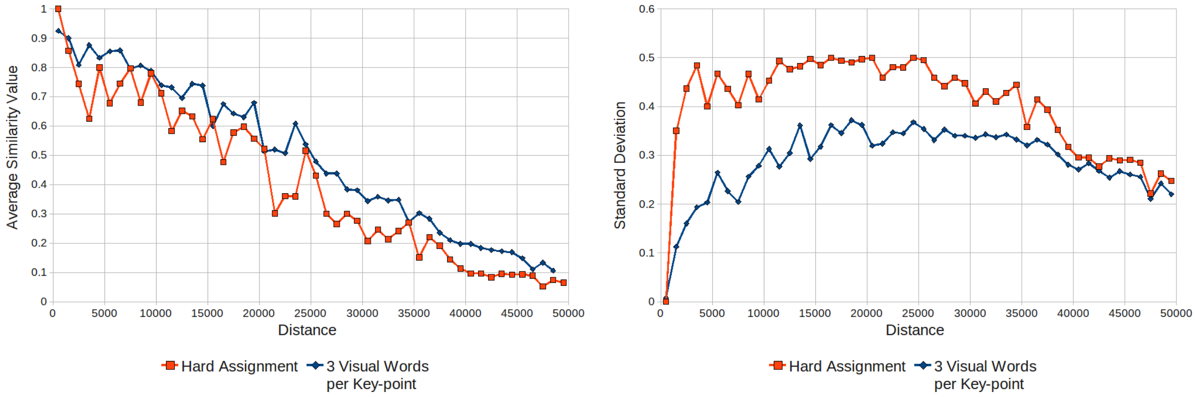
Results



(a) Average similarity value

(b) Standard deviation

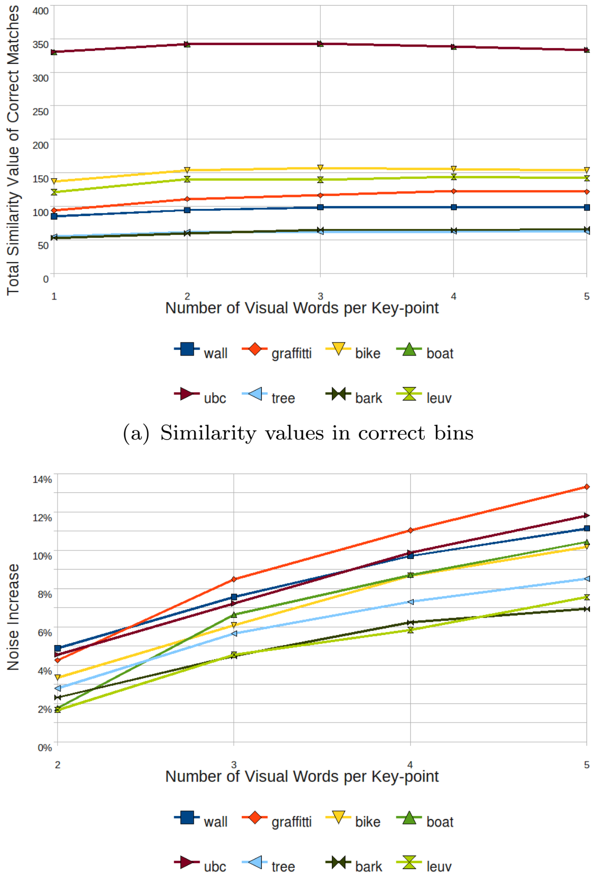
FIGURE B.1: Similarity value distribution – Coca-Cola can (DART). The experimental results obtained for the adapted solution (see section 5.3.3).



(a) Average similarity value

(b) Standard deviation

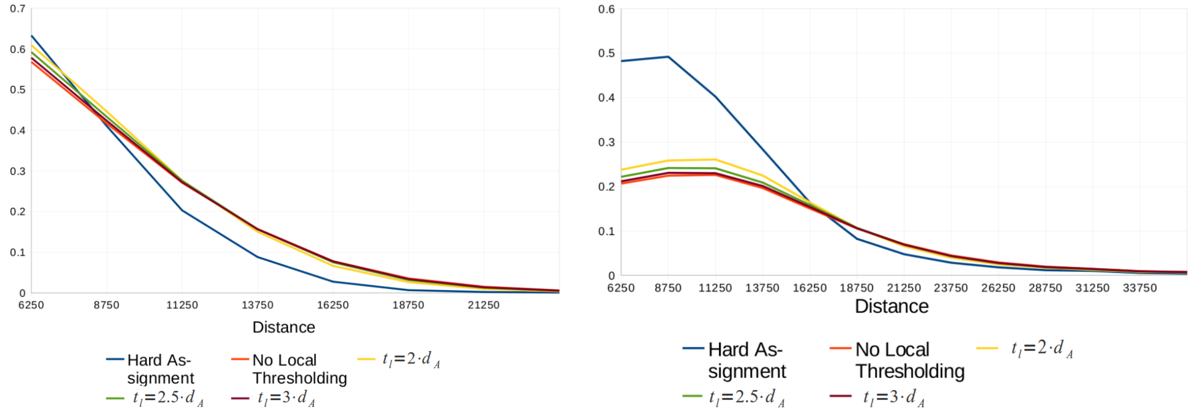
FIGURE B.2: Similarity value distribution – All Souls church (SIFT). The experimental results obtained for the adapted solution (see section 5.3.3).



(a) Similarity values in correct bins

(b) Noise increase

Figure B.3: Characteristics of additional matches discovered with soft assignment (SIFT). The experimental results obtained for the adapted solution (see section 5.3.3).



(a) Average similarity value

(b) Standard deviation

Figure B.4: Similarity value distribution with local thresholding – Coca-Cola can (DART). The experimental results obtained for the additional configurations of the implemented solution (see section 5.3.3).

Collection / Dictionary	**Descriptor**	**Assignment**	**Av. Position of 1st Perfect Match**	**Top-5 Perfect Match**
Iris / Iris	SIFT	Hard	1.720339	97.46%
		Soft	1.576271	99.15%
	DART	Hard	3.440678	90.68%
		Soft	3.491525	88.14%
Iris / Caltech	SIFT	Hard	4.203390	85.59%
		Soft	3.237288	91.53%
	DART	Hard	9.779661	64.41%
		Soft	5.711864	78.81%
Oxford / Oxford	SIFT	Hard	20.363636	83.64%
		Soft	17.690909	90.91%
	DART	Hard	12.563636	89.09%
		Soft	8.454545	89.09%
Oxford / Caltech	SIFT	Hard	34.163636	89.09%
		Soft	7.758710	87.27%
	DART	Hard	15.8909091	87.27%
		Soft	15.581818	89.09%

Table B.1: Evaluation of system with adapted soft assignment solution for different collections (see section 5.4.1) – further results.

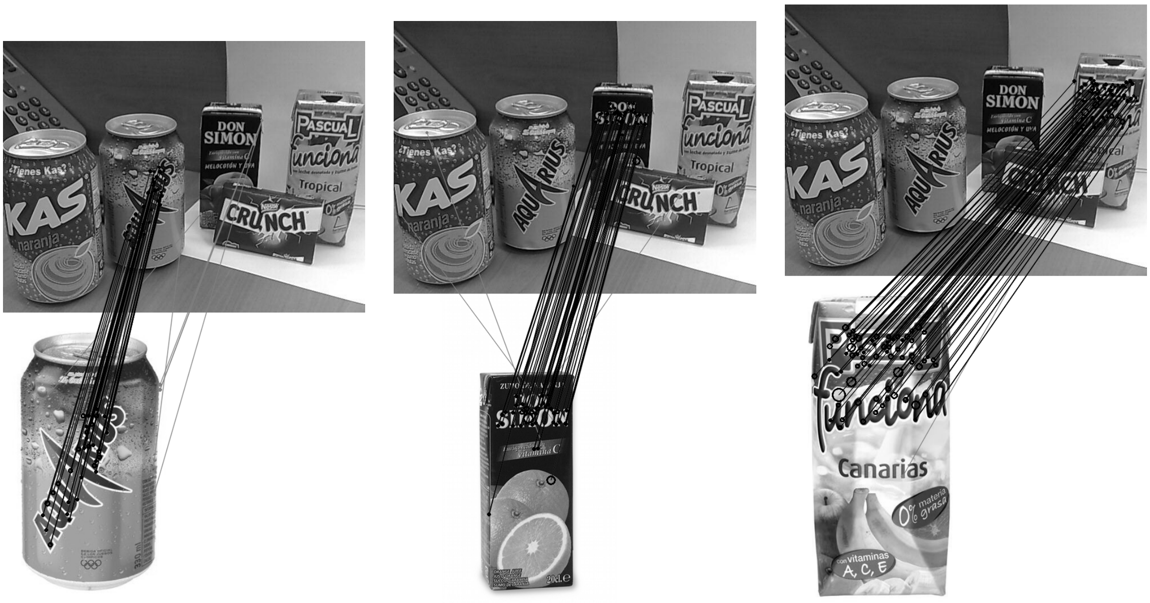


Figure B.5: Example of the use cases where many objects are present in the scene (see section 6.6). The inliers are coloured in black, the outliers – in grey.

Collection	**Descriptor**	**Dictionary Origin**	**MAP**	
			Original Configuration	**Extended Configuration**
Iris	SIFT	Iris	0.88137	0.9070
		Iris Others	0.77511	0.8696
		Caltech	0.72312	0.8502
	DART	Iris	0.75125	0.7829
		Iris Others	0.54462	0.6943
		Caltech	0.47369	0.6665
Oxford	SIFT	Oxford	0.38091	0.4832
		Paris	0.31434	0.4342
		Caltech	0.30416	0.4117
	DART	Oxford	0.32289	0.3993
		Paris	0.26373	0.3668
		Caltech	0.24087	0.3341
Zubud	SIFT	Zubud	0.9092	0.9416
		Paris	0.8660	0.9263
		Caltech	0.8506	0.9169
	DART	Zubud	0.8825	0.8927
		Paris	0.8198	0.8819
		Caltech	0.8236	0.8876
UK-Bench	SIFT	UK-Bench	0.74084	0.82900
		Iris	0.71650	0.82378
		Caltech	0.72338	0.82457
	DART	UK-Bench	0.74305	0.8079
		Iris	0.73114	0.0.8098
		Caltech	0.72049	0.8093
Nokia	SIFT	Nokia	0.2452	0.3579
		Paris	0.1764	0.3332
		Caltech	0.1720	0.3106
	DART	Nokia	0.1533	0.1816
		Paris	0.1157	0.1495
		Caltech	0.1133	0.1425

Table B.2: MAP final results for all the collections

Bibliography

[1] Takacs et al. Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In *Proc. MIR*, 2008.

[2] X. Anguera, P. Obrador, T. Adamek, D. Marimon, and N. Oliver. Telefonica research content-based copy detection. TRECVID Submission.

[3] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. IEEE 9th International Conference on Computer Vision*, pages 1470–1477, 2003.

[4] Y. Rui, T. S. Huang, and S. Chang. Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 1999.

[5] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 2008.

[6] J. Philbin et al. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007.

[7] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval.* Cambridge University Press, 2008.

[8] J. Philbin. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proc. CVPR*, 2008.

[9] Y. Wang. Principles and applications of structural image matching. *Journal of Photogrammetry Remote Sensing*, 53, 1998.

[10] K. Mikolajczyk et al. A performance evaluation of local descriptors. In *Proc. PAMI*, 2005.

[11] K. Mikolajczyk et al. A comparison of affine region detectors. In *Proc. IJCV*, 2006.

[12] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. In *Proc. IJCV*, 2004.

[13] D. Lowe. Distinctive image features from scale-invariant keypoints, cascade filtering approach. In *Proc. IJCV*, 2004.

[14] A. W. Gruen. Adaptive least squares correlation: a powerful image matching technique. *South African Journal of Photogrammetry, Remote Sensing and Cartography*, 1985.

[15] S. Edelman, N. Intrator, and T. Poggio. Complex cells and object recognition. Manuscript, 1997.

[16] H. Bay et al. Surf: Speeded up robust features. In *Proc. ECCV*, 2006.

[17] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. In *Proc. Biritish Machine Video Conference*, 2002.

[18] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006.

[19] M. Weber, M. Welling, and P. Perona. Towards automatic discovery of object categories. In *Proc. CVPR 2000*, 2000.

[20] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *ECCV 2002*, 2002.

[21] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *Proc. VISSAPP*, 2009.

[22] J. H. Freidman, J. L. Bentley, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Trans. Math. Softw.*, 1977.

[23] C. Silpa-Anan and R. Hartley. Optimised kd-trees for fast image descriptor matching. In *Proc. CVPR*, 2008.

[24] T. Quack. *Large-scale Mining and Retrieval of Visual Data in a Multimodal Context.* PhD thesis, ETH Zurich, January 2009.

[25] D. Marimon, T. Adamek, A. Bonnín, and R. Gimeno. Darts: Efficient scale-space extraction of daisy keypoints. *Submitted to CVPR*, 2010.

[26] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21, 1994.

[27] E. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[28] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 178–185, 2009.

[29] P. L. Rosin. Unimodal thresholding. *Pattern Recognition*, pages 2083–2096, 2001.

[30] Shao et al. Fast indexing for image retrieval based on local appearance with re-ranking. In *Proc. ICIP*, 2003.

[31] C. M. Bishop. *Pattern Recognition and Machine Learning.* Springer, 2006.

[32] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Comm. ACM*, 15, 1972.

[33] M. A. Fischler and R. C. Bolles. Random sample consensus. *Comm. of the ACM*, 1981.

[34] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2006.

[35] J. M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2, 2009.

List of Figures

List of Tables

Abbreviations

MSER	**M**aximally **S**table **E**xtrema **R**egions
MAP	**M**ean **A**verage **P**recision
RANSAC	**RAN**dom **SA**mple **C**onsensus
SA	**S**oft **A**ssignment
SIFT	**S**cale **I**nvariant **F**eature **T**ransform
SCV	**S**patial **C**onsistency **V**erification
SURF	**S**peeded **U**p **R**obust **F**eatures
TREC	**T**ext **RE**trieval **C**onference