RESEARCH

Localizing global descriptors for content based image retrieval.

C. Iakovidou[1*], N. Anagnostopoulos[2], A. Kapoutsis[1], Y. Boutalis[1], M. Lux[2] and S.A. Chatzichristofis[1]

[*]Correspondence:
ciakovid@ee.duth.gr
[1]Democritus University of Thrace, Department of Electrical and Computer Engineering, Xanthi, Greece
Full list of author information is available at the end of the article

Abstract

In this paper we explore, extend and simplify the localization of the description ability of the well-established MPEG-7 (SCD, CLD and EHD) and MPEG-7-like (CEDD) global descriptors, which we call the SIMPLE family of descriptors. Sixteen novel descriptors are introduced, that utilize four different sampling strategies for the extraction of image patches to be used as points-of-interest. Designing with focused attention for content based image retrieval tasks, we investigate, analyse and propose the preferred process for the definition of the parameters involved (points detection, description, codebook sizes and descriptors' weighting strategies). The experimental results conducted on four different image collections reveal an astonishing boost in the retrieval performance of the proposed descriptors compared to their performance in their original global form. Furthermore, they manage to outperform common SIFT and SURF based approaches while they perform comparably if not better, against recent state-of-the-art methods that base their success on much more complex data manipulation.

Keywords: Image Retrieval; Local Features; SIMPLE Descriptors

Introduction

Extracting a meaningful descriptor from an image is a central problem for a variety of computer vision problems. Depending on the application, a successful vectorization of an image's depictions can be utilized to solve matching or correspondence problems. However, the design strategy of a description mechanism for problems like classification, object recognition or tracking must be adjusted accordingly. The impact of factors such as the kind of features employed, computational complexity, storing requirements and scalability can vary significantly in different computer vision domains.

In this paper we are interested in exploring the combination of features that best describe an image with respect to its visual properties and its visual content, specifically focusing on Content Based Image Retrieval (CBIR) tasks. When designing descriptors for CBIR one must take into account the ever growing data involved in the process. Image collections are growing exponentially in a variety of domains (medicine, private life, industry, journalism, tourism etc.), making the need for an effective and yet efficient retrieval system, imperative.

However, trying to define what makes a useful and meaningful retrieval for the user remains still unsolved and is most likely not an engineering problem. Different benchmarking datasets try to cover various retrieval scenarios with diverse types of images and different levels of semantics in query to result relevance interpretation. The complexity of the problem is evident just by thumbing through the great variety of proposed implementations that address the issue [1, 2, 3, 4, 5, 6, 7, 8].

Briefly making a historical overview, the first attempts to vectorize image contents proposed extracting global image features such as color, texture and shapes that are calculated over the entire image. The foremost advantage of extracting global features is the low cost of the single-feature space computations. Moreover, a global vector representation is a very effective strategy for certain retrieval task. For instance, trying to classify natural-scene depicting images, where a number of blue uniform patches that are part of a lake are equally important as highly textured parts depicting leafage. Annotating an image solely by a global feature vector, however, often leads to a rather generalized outline of its visual information.

As collections and retrieval scenarios became more demanding, global feature methods were overshadowed and often also outperformed by methods that employed local features (LF). Among the most popular Points of Interest (POI) detectors are corner detectors Harris, Shi-Tomasi and FAST [9, 10, 11] and blob detectors SIFT [12], SURF [13], to name a few. Using POIs, the representation of the image is mapped into a high dimensional local feature space. In applications like Simultaneous Localization And Mapping (Visual SLAM [14]), panorama construction, object recognition and tracking, these extracted POIs are used directly to find one-to-one matches between depictions. In CBIR, however, direct usage is impractical even with today's available computational resources. Typically, hundreds or even thousands of LF are extracted per image. To reduce memory cost and speed up image matching, the features are quantized through some aggregation procedure.

A widely and extensively used approach is the Bag-Of-Visual-Words (BOVW) model which originated from the document retrieval field. Because of its simplicity, flexibility, and effectiveness, it has been adopted in various applications such as video classification, 3D shape categorization and image retrieval [15, 16, 17]. The BOVW model first constructs a codebook using a clustering algorithm over all detected LF in an image collection. Each cluster represents a visual word while the total number of clusters is typically predefined. Then, an image is represented as a histogram of the visual words and each bin of the histogram is weighted with a tf-idf score or its variants. The aggregation model, manages to achieve a vast reduction of the high dimensionality that LF introduce, but simultaneously burdens the implementation with a number of free parameters such as predicting the appropriate codebook size and the preferred weighting strategy.

Of course, this type of feature quantization introduces the respective loss of the discriminative ability of the features. Thus, over the years numerous improvements and alternatives have been proposed. The soft quantization and soft assignment techniques proposed in [18] and [19] respectively, reduce the quantization error of the original BOVW model paying a price in terms of memory overload and higher searching time. Alternatively, the Fisher Vector [20] uses the Gaussian Mixture Model to train the codebook and quantizes the features by calculating the probability of a feature falling into the Gaussian Mixture. Different approaches like Hamming Embedding [21] improve the model by generating binary signatures coupling visual words and providing thus additional information to filter false positives. Recently an alternative to the BOVW model, the Vector of Locally Aggregated Descriptors (VLAD) [22] has gained the community's attention. Given a codebook, instead of creating a vector of frequencies, the VLAD model creates a vector of differences, as distances, between a feature and the cluster's center. VLAD manages to speed up the aggregation step but leads to high dimensional vector representations per image, which can affect the scalability of a method. Finally, authors in [23] focus on a multilayer deep learning architecture to represent high-level features in an effective compact manner, while

[24, 25, 26] emphasise the need for domain-adaptive dictionary learning and the benefits of effectively fusing multiple information sources.

Acknowledging the fact that there will probably never be a solution that fits all, we are interested in exploring the benefits of revisiting, reusing and combining strategies proposed from both global feature and local feature approaches, seen under the light and understanding of nowadays knowledge. In this paper we propose 16 novel local features' descriptors that adapt on the hybrid approach first introduced in [27], named SIMPLE. In its essence, the SIMPLE scheme suggests a framework that localizes the description mechanism of older well established global descriptors. Originally the SIMPLE features were a combination of the SURF detector, used to sample textured image patches in multiple scales, and the MPEG-7 SCD, CLD, EHD [28] and the MPEG-7-like CEDD [29, 30] global descriptors, used for describing the patches. One of the key elements of the scheme is that it allows for indirect combination of texture and color information, eliminating the need for complicated fusion techniques. Finally, having conducted over 2000 experiments for this work, we put all the obtained data to good use and statistically analyse the impact that the varying BOVW set-ups have on the robustness of the retrieval performance.

Related Work

The MPEG-7 family of global descriptors has been widely studied and referenced in the literature. The compact and effective representation of images that they provide has introduced a great number of improved techniques that build upon the original standard. Here, we will focus on attempts that propose their utilization combined with additional local information of some kind.

The fusion of various low-level MPEG-7 descriptors is proposed in [31] for content-based image classification. A 'merging' fusion combined with a support vector machine (SVM) classifier, a back-propagation fusion with a KNN classifier and a Fuzzy-ART neurofuzzy network strategies are explored, that can be extended in matching the segments of an image with predefined object models. The fusion (baseline fusion and score fusion) of MPEG-7, SIFT, and SURF is also explored and evaluated in [32] to address content-based event search. The detailed results conclude that the MPEG-7, SIFT, and SURF are broadly comparable, and also highly complementary. In [33] a classification-driven similarity matching is presented and evaluated for the biomedical image domain. Various low-level global colour, edge, and texture related features are extracted (CLD, EHD, CEDD, FCTH [34]) and utilized along with a visual concept feature [35] extracted using the 'bag of concepts' model (that comprises of local colour and texture patches) achieving thus, the generation of feature vectors in different levels of abstraction.

Authors in [36] present a grid-based framework for image retrieval where the images are partitioned into blocks. Localized feature representations employing the MPEG-7, HS and the HSV color histogram descriptor [28] are extracted and achieve better results compared to global techniques. In [37] authors index a collection of images combining local and global features. The method extracts SURF local features and five MPEG-7 descriptors (CS, CL, SC, HT, EH) as global features. Each image is associated with six text fields, one corresponding to the bag-of-features obtained from the SURF descriptor and five surrogate text representations, one for each MPEG-7 descriptor. These segments, form the basic units on which search is performed. Finally, authors in [38] use cluster correlograms to combine MPEG-7 descriptors with spatial information, for image categorization. They

employ fixed partitioning and salient points schemes to extract image patches and use four MPEG-7 descriptors to represent them. Similar patterns are aggregated into a cluster codebook. A correlogram is then constructed from the spatial relations between visual keyword indices in an image, in order to obtain high-level information about the relational context. Four 2D signatures (one for each MPEG-7 descriptor) are assigned per image, which leads to a feature dimension of $4m^2$, where m is the number of clusters used in the clustering algorithm.

Overall, the most commonly followed strategy to combine global and local information usually relies on some late fusion method that severely slows down the retrieval process. Fixed partitioning of images and region based image segmentation are also presented but when applied, not only add a new level of complexity but also tend to suffer in domain specific tasks, were background information and foreground are not easily dissociated. Our proposed implementation of localized MPEG-7 descriptors is designed around the fact that CBIR tasks employ a large number images for indexing and retrieval. Thus, efficiency, low complexity and compactness of the final representation is of great importance.

Extending the SIMPLE family of descriptors

The SIMPLE family of descriptors proposed in [27] is a combination of the SURF local-points detector and three of the global MPEG-7 descriptors along with the MPEG-7-like global CEDD descriptor, to produce new local features specifically designed for CBIR. The SURF detector is employed to locate and extract salient image patches, whose size is determined as a squared area ($s \times s$) according to the scale (S) that the points were detected in. The method proceeds by applying the aforementioned global descriptors on the detected patches, as if they were standalone images. This results in four different kinds of local features that were tested for CBIR using the BOVW model. In this paper we are not only interested in exploring different combinations of detectors and descriptors, but also in analysing the results, so as to gain a deeper understanding of the preferred attributes to incorporate in a CBIR scheme, according to the application's and the user's requirements.

The image datasets

The employed dataset is one of the most important factors when building a CBIR system. Even the most successful implementations reported, cannot guarantee high performance for any kind of datasets. In an effort to draw useful conclusions concerning the preferred type of point detection and description mechanisms in a generalized manner and simultaneously minimize the case that good achieved performances might have to do with specificities of the database, we decided to employ four diverse kind of datasets.

The **UKBench** image database [39] consists of 10,200 images, separated in 2,250 groups of four images each. Each group includes images of a single object placed in the center of the image, captured from different viewpoints and lighting conditions. This dataset represents a much requested retrieval scenario in real-life applications for industrial and commercial purposes. The collection presents high in-class variability and the information concerning localized aspects of the images' content is of great importance. Thus, local features are reported to perform better in this collection than global descriptors do.

The **UCID** image collection [40] consists of 1,338 images on a variety of topics including natural scenes and man-made objects, both indoors and outdoors. All the UCID images were subjected to manual relevance assessments against 262 selected images. UCID, includes several query images where the ground truth consists of images with similar visual

concept to the query image, without necessarily the co-occurrence of the same objects. In contrast to the UKBench dataset, the visual content of the images that form this database favours the performance of the global descriptors [41].

The **INRIA Holidays** dataset [21] consists of 1,491 photos, depicting a variety of natural and manmade scenes, captured mainly during personal holidays. The challenges that a retrieval system has to deal with are rotations, viewpoint and illumination changes, blurring etc. The Holidays dataset is accompanied by a ground truth for 500 queries along with the images that represent the same scene for each one of them.

Finally, the **Zurich Building Database (ZuBuD)** [42] consists of two separate parts. 201 buildings were captured, from five different viewpoints each, forming a dataset of 1,005 images of Zurich's city building. The queries' part contains 115 additional images of lower resolution, depicting some of the buildings of the main dataset captured from a different viewpoint and sometimes under different weather conditions. For each query, only the images that represent the same building are considered relevant.

For readability purposes we focus and provide analytic experimental results on the first two datasets (UKBench and UCID) and proceed with the presentation and comments for the rest of the employed collections in a more condensed form.

Detecting points of interest

Four different points-of-interest detection mechanisms were explored in this paper. In all cases the objective is to extract square image regions, hereinafter referred to as image patches. During the detection stage, we are only interested in locating the position (x, y) of the centres of the image patches and deciding their size. Their description will be handled in a subsequent step, utilizing descriptors formerly used in global features' techniques.

- First we employed the **SURF detector**. The SURF detector uses the determinant of the Hessian to detect both the location and the scale of blob-like structures. The Hessian matrix is approximated, using a set of box-type filters. The scale-space is analyzed by up-scaling the filter size rather than iteratively reducing the image size. Independently of their size, these approximate second-order Gaussian derivatives are evaluated using integral images, significantly speeding up the whole process. The responses are stored in a blob response map, and local maxima are detected and refined using quadratic interpolation.
- The second detector we employed, was the **SIFT detector**. The key points are searched in a scale-space by applying the difference of Gaussians function and locating the maxima and the minima to a series of re-sampled and smoothed images. We define our image patches' size (s x s) according to the scale (S) they were detected.

The first two detectors both focus and locate blob like structures in images. This means that the obtained patches will contain interesting achromatic information. Even if we do not proceed and describe this achromatic information, but instead focus on the colour information contained in the patches, we still achieve to indirectly combine texture and colour information. We are describing colour information with textural attention, i.e. apply a colour descriptor on image regions where something interesting is happening texture-wise. We used the SIFT and SURF emguCV detector implementations, following the default parameter initializations.

However, CBIR tasks are not always oriented towards object recognition and direct matching. Some applications request retrieval results to be similar in a more conceptual

fashion. Image regions that may not carry textural information should still be vectorized. For instance, blue, uniform patches of sky or sea depicting images, could boost the retrieval performance of a system that is ranking landscape images to a provided query. Thus, inspired by the principle that global features CBIR systems are designed around, we implemented and tested two more detectors: a uniform, random, multiscale image patches' generator and a random patches' extractor where the selection of the centres (x, y) of the patches follow the Gaussian distribution.

- The **Random patches' generator**, as its name implies, randomly selects x and y positions in the images, to mark square regions of pixels. The probability of the selection, both for x and y, follows the uniform distribution (for a visual, kindly refer to Fig.1, third column). The sizes of the regions were decided as follows: the smallest patch size (hereinafter referred to as Reference Patch, *RP*) was set to 40 x 40 pixels, so as to be aligned with the highest patch size limitation, that is introduced by the CEDD descriptor (kindly refer to the next Section). From there, we employ a scaling factor (sf) to produce larger patches of sizes $RP*sf \times RP*sf$ pixels. More details about the sf and the total number of patches in this implementation, can be found in the Experimental Set-ups Section.
- The **GaussRandom patches' generator**, operates as the Random generator presented above, only this time, the probability of the selection of an x, and the selection of a y follow two separate univariate Gaussian distributions with the mean values set at the center of the x and y range, respectively. This means that the x,y centres are more densely sampled in the centre of the image and become gradually sparser as we move to the outer parts of the image (for a visual, kindly refer to Fig.1, fourth column). The standard deviation (σ) is automatically adapting to the image dimensions of each dataset so that a 2σ standard deviation includes 95.5% of the samples, while a 3σ covers 99.7%. If for instance the image has a 400×600 resolution the first Gaussian will have a $meanvalue = \frac{400}{2}$ and a $\sigma = \frac{400}{6}$, while respectively the second Gaussian has a $meanvalue = \frac{600}{2}$ and a $\sigma = \frac{600}{6}$.
  We employed this type of sampling, driven by the fact that, usually, the main theme of the image or the dominant objects, both in queries and collections, are the centred depictions.

The global descriptors employed to be localized

Four different global descriptors from the literature were selected to be localized. Three of them originate from the MPEG-7 family of global descriptors (SCD, CLD and EHD) [28], and the fourth global descriptor (CEDD) originally presented in [29] is an MPEG-7-like descriptor, in the sense that its implementation principles are strongly inspired by the MPEG-7 standard.

We proceed focusing on the specific attributes of each method that differentiate the experimental setups and will allow us to get some insight into what type of descriptors are best suited for CBIR, under different circumstances. All the selected descriptors were preferred because they are well established, widely accepted, they are easy to implement and, most importantly, represent the images' features in a compact and quantized manner. Since we are particularly interested in evaluating local features for CBIR, it is intuitive that compactness of the vectors and quantized local feature representations, are imperative.

Image collections can vary from a few thousands to millions of images. Thus, the more compact the descriptor, the more likely it is for the retrieval system to be able to manage

great amounts of data on limited computational resources. Furthermore, the scope of an image retrieval oriented local descriptor, is to provide small vector distances for visually similar patches. Approaches in the literature, however, utilize local features that were originally developed for different tasks. The goal of a local feature intended, for instance, for Simultaneous Localization And Mapping (SLAM) or panorama construction, is to describe each point of interest as uniquely and detailed as possible, so as to achieve a one-to-one matching of points in different images. Retrieval systems that employ such local features are often forced to fail, due to many possible matching candidates whose vector distance is not apparent.

On the other hand, quantizing features means that image properties (like detected colours or edges) are categorized in a preset number of explicitly defined possible variations. When employing such features to describe the image, we get a more abstract image signature. This abstract representation allows for faster and safer comparisons of similarities between images. Especially in CBIR tasks, where the objective is not to find the one and only similar image, but a set of k top correctly retrieved results, this discrete domain of features minimizes classification errors [43].

Taking into account that colour is a very important element for image retrieval tasks [3, 41, 29, 38], we begin our description of the detected patches employing two MPEG-7 colour descriptors.

- The **Scalable Colour Descriptor (SCD)** [28] is essentially a colour histogram in a fixed HSV colour space, achieved through a uniform quantization of the space. A total of 256 coefficients is used to represent the descriptor. Since it is a histogram, it is rotation and transformation invariant. Moreover, due to the quantization of the colour space, SCD presents good tolerance to change of lightning conditions and hue variations.
- The **Colour Layout Descriptor (CLD)** [28] represents the spatial distribution of the colour in images. In order to incorporate the spatial relationship, each image patch needs to be divided into 8 x 8 discrete blocks. Any image patches too small for this type of division are ignored in our implementation, as if they were never detected. This descriptor quantizes the space domain, allowing some slight sifts and rotations to be flattened and also presents good tolerance to changes in lightning conditions and hue variations because it represents each block by calculating the dominant colour, thus, indirectly quantizing the colour space as well.

Next, we employ an edge descriptor and a descriptor that combines both texture and colour information, so as to widen the spectrum of tested approaches and gain a generalized outlook on local features and their retrieval effectiveness.

- The **Edge Histogram Descriptor (EHD)** [28] represents the spatial distribution of five types of edges in the image. A given image patch is subdivided into 4 x 4 sub-image patches and a local edge histogram is computed. Again, in our implementation any image patch that is too small to undergo such a division is ignored as though never detected. This descriptor quantizes the edge information into five broadly grouped edge types that vary with intervals of 45 degrees, resulting in features that present commensurate rotation invariance.
- The **Color and Edge Directivity Descriptor (CEDD)** [29] utilizes both colour and edge information in a compact, quantized manner. The original CEDD implementation demands a division of the image patch into 40 x 40 blocks of at least 2 x 2 pixels,

each. However the latest version of CEDD[1], adapts to the description of smaller sized images and according to the image's size in question, defines a minimum of 20 x 20 blocks' division of at least 2 x 2 pixels each. For the edge information extraction it adopts the five filters presented in the MPEG-7 EHD descriptor along with an additional Non-Edge filter and it introduces a heuristic pentagon diagram to classify each block into one or more edge types. The colour information is represented by a 24-bins colour histogram where each bin corresponds to a preset colour. This descriptor, just as the EHD, presents rotation invariance of 45 degrees and due to the quantized colour space that it uses, it presents also tolerance to change in lightning condition and hue variations.

Utilizing the SIMPLE local features in a CBIR system

By combining the four different detection mechanisms and localizing the description ability of the four global descriptor presented in the previous section, we produced four sets of local features. Using the SURF detector: SIMPLE srf-SCD, srf-CLD, srf-EHD, srf-CEDD. Using the SIFT detector: SIMLE sft-SCD, sft-CLD, sft-EHD, sft-CEDD. Using the Random detector: SIMPLE rnd-SCD, rnd-CLD, rnd-EHD, rndCEDD. Using the Gaussian Random detector: SIMPLE gaussRnd-SCD, gaussRnd-CLD, gaussRnd-EHD, gaussRnd-CEDD. In order to test them in CBIR tasks, we employed the Bag-of-Visual-Words (BOVW) model to calculate vector image representations and went on calculating 8 weighted equivalents of those vectors by applying an equal number of weighting schemes.

Please note that we deliberately chose to employ the simplest form of the model and not any of the improvements that have been recently proposed in the literature because our goal is to calculate the performance of the local features and their ability to capture the images' contents. The proposed local features can be employed for CBIR using any other retrieval system framework, but this exceed the scope of this paper.

The Bag-of-Visual-Words model

The BOVW model uses an unsorted set of discrete Visual Words (VW) to represent the contents of an image. It is directly inspired by the bag-of-words (BOW) model, which was first introduced for text classification. In our implementation, when all SIMPLE local features have been detected in a collection of images, we randomly select a sample to be clustered via the k-means classifier into a preset number of clusters (Visual Words), so as to form the Codebook. Each image from the collection is then represented by a histogram of the frequencies of the Visual Words that it contains. When a query is set to the system, its features are also extracted and matched to the VWs of the Codebook and the VW histogram is calculated. This is the simplest form of the BOVW model.

The Weighting Schemes

We incorporate the common textual term weighting schemes in the BOVW model. The first weighting factor is the raw Term Frequency ($tf_{t,d}$) where a weight is assigned to every term (t) in the codebook according to the number of occurrences in a document (d). A second factor for assigning weights is the Document Frequency (df_t). This time df_t is defined as the number of documents that contain the term t. Many times, the inverse document frequency $idf_t = log(N/df_t)$ of a collection is used to determine weights, where

[1]The latest version of CEDD can be found in http://tinyurl.com/CEDD-Descriptor

N is the total number of documents in the collection. In our case, "term" equals "Visual Word", and "document" equals "image". Last, a normalization can be performed to quantify the similarity between two documents in terms of the cosine similarity of their vector representation.

The SMART notation is a compact way to describe combinations of weighting schemes in the form of (d.d.d). The first letter denotes the tf weighting method, the second letter denotes the df weighting method, and the third letter specifies the normalization used. Table 1 presents the SMART notation for several $tf.idf$ variants. For more details concerning the weighting schemes adoption, kindly refer to [41].

Table 1 SMART NOTATION

tf	df	Normalization
n(natural): $\mathrm{tf}_{t,d}$	**n**(no): 1	**n**(none): 1
l(log): $1+\log(\mathrm{tf}_{t,d})$	**t**(idf): $\log(N/\mathrm{df}_t)$	**c** (cos):$1/\sqrt{w_1^2+\ldots+w_M^2}$

In our implementation, after generating the VW histogram for every image (collection and query) through the BOVW model, the vectors are recalculated using the 8 weighting schemes (kindly refer to Table 1).

Experimental Set-ups

Sampling Parameters

The SIFT detector produces on average 1000 patches-of-interest per image on the UK-Bench collection, 1400 on the UCID collection, 1000 on Holidays and 1600 on ZuBuD. Respectively, SURF detects on average 600 patches on the UKBench, 800 on the UCID collection, 650 on Holidays and 1850 on ZuBuD, per image. However, the usability of the patches is determined by their size, due to the limitation that the description methods introduce. The percentage of unusable patches can not be foreseen, since it depends on the image collections involved. Through our tests we found that statistically about 20% of SIFT points and about 10% of SURF points are unusable for our implementations.

Another interesting observation made through our tests concerning the SIFT and SURF point detectors has to do with their distribution on the images. Since they are both blob detectors, the total number and the centers' coordinates of the points, vary significantly depending on the depiction. Uniform areas of the images are disregarded completely from these detectors. Thus, we had images with less than 100 points and others with more than 4000.

In the second and third column of Figure 1 we present scatter plots of the x,y centers for SIFT and SURF. The first two rows are example images from the UCID collection, while in the third row the results report the accumulative points over the whole collection. In the first example (Eiffel tower) even though 1513 points are detected, almost no information will be considered from the upper half of the image (sky). This is a significant loss since the depicted landmark, being an outdoors location, is in most cases captured with this blue background. The loss of useful information is even more dramatic in our second example (a person running on the beach). Using the SIFT and the SURF points we gain almost no information about the surroundings (brawn uniform sand, green see and blue sky).

Finally, when plotting the points detected over the whole collection we see that spatially every possible x,y was picked as a point center. What is more interesting, is the distribution

of those coordinates. In this multi-theme collection [2] the x and y variables present no particular distribution pattern when examined per image, but when collectively studied they clearly follow a Gaussian-like distribution.

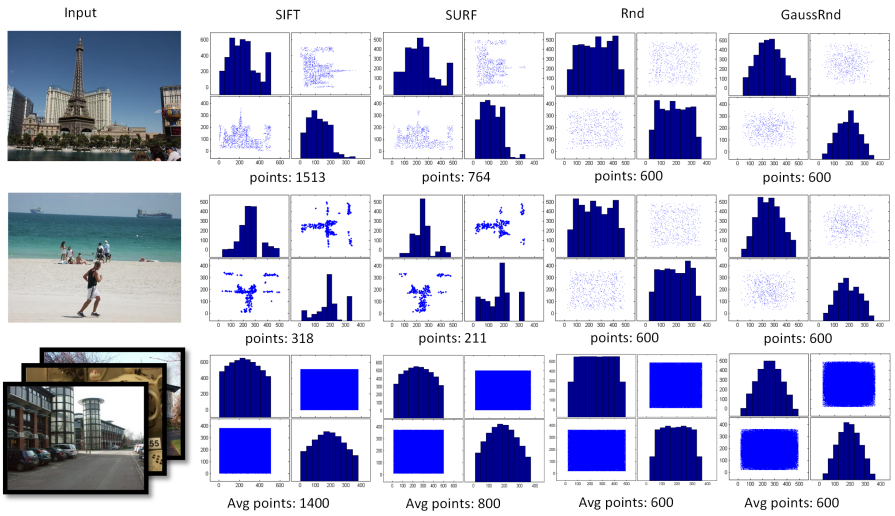

Figure 1 Sampling distributions of the four employed methods: (a) first two rows are example images taken from UCID, (b) the accumulative distribution over all images on UCID.

The aforementioned findings and detected drawbacks inspired the two proposed random patches' generators. As discussed earlier in the "Detecting points of interest" subsection, this type of sampling allows us to utilize information from parts that would be disregarded from blob detectors. Furthermore, the constant number of samples per image (i) produces final vector representations that do not need normalization in order to be compared via a distance measure and (ii) can be pre-defined so as to be manageable depending on the available resources and the scale of the application.

For the two random sampling strategies, in order to maintain the order of magnitude suggested both from SIFT and SURF for the employed collections, we set the number of extracted patches to be a total of 600 per image (i.e 150 samples per scale). Taking into account the highest size limitation introduced by CEDD, we define the minimum patch size to consist of 40 x 40 pixels. This will ensure that all image patches will be usable for description by all four of our employed description methods. Next, we scale the minimum/reference patch size by a scaling factor sf, to produce different patch sizes. The upper patch size limit is related to the employed images. We did not want to produce image patches that would be greater that $1/3$ of the smallest image dimension. Thus, the greatest sf used, was $sf = 3$ resulting into 120x120 pixels image patches. Having an upper and a lower size limit, allowed us to decide on further scaling factors for in-between patch sizes. The sf for both Rnd and GaussRnd generators were 1, 1.6, 2.3 and 3, and the respective patch sizes were 40x40, 64x64, 92x92 and 120x120.

[2]The results are in-line for UKBench and Holidays, while for the ZuBuD dataset the accumulative distribution presents less of a curve but rather a more flat distribution.

BOVW Parameters

Taking into consideration limitations in computational resources and desired efficiency, we set our four different codebook sizes to consist of 32, 128, 512 and 2048 VWs, respectively. The codebooks are generated by forwarding a random 10% sample of the extracted features to a k-means classifier. We weight the histograms of VWs with eight weighting schemes and conduct the similarity search between query and dataset using the Euclidean distance measure.

Querying Mode

For the evaluation of the features' retrieval effectiveness we employ four different image collections (UKBench, UCID, Holidays, ZuBuD) that vary both in theme and relevance interpretation, so as to minimize the probability of good performances occurring due to collection specificities. Concerning the querying mode, for the UKBench collection, the first 250 images of the first 250 groups were used as queries. The ground truth consists of the four images belonging to the same group as the query. For the UCID collection the first 262 images were used as queries. By design, all the UCID images, were subjected to manual relevance assessments against 262 selected images, creating 693 ground truth image sets for performance evaluation. For the Holidays and the ZuBuD collections we also followed the default querying mode, using the 500 and 115 query images that accompany the datasets, respectively.

Baseline Formation and Evaluation Metrics

In order to ensure fair and direct comparison we reimplemented and tested under the same retrieval set-ups five well established local features descriptors from the literature (SURF, SIFT, opponent-SIFT [5] ORB [44] and BRISK [45] [3]) and since our method is a combination of local POIs detectors and global features descriptors, we also conducted experiments for 7 global descriptors (using the img(Rummager) [47] application and their default settings), including of course the original MPEG-7 SCD, CLD, EHD.[4] To evaluate the systems' performance, we calculate the Mean Average Precision (MAP, max at 1) and the Average Normalized Modified Retrieval Rank (ANMRR, max at 0). For the UKBench and the UCID collections we also provide the precision-at-position ($P@k$, with $k = 4$ for UKBench and $k = 10$ for UCID, max at 1) [49] evaluations.

In each experiment, we assumed as baseline the best performance that can be obtained employing a non-SIMPLE descriptor of those we reimplemented. However, in order to allow the readers to compare and get a better perspective of the achieved performances we also include state-of-the-art methods from the recent literature that propose improvements on various different aspects of a retrieval system.

Experimental Results

In total we performed $16_{SIMPLE} \times 4_{Codebooks} \times 8_{WS} \times 4_{Collections} = 2048$ experiments for the evaluation of the proposed local features. In this section we will provide the evaluation of the retrieval performances of the proposed SIMPLE descriptors and discuss the impact of

[3] Local Features descriptors were tested using the recently proposed GRIRe [46] open source framework and the respective OpenCV implementation of the descriptors.

[4] The MPEG-7 descriptors availiable on img(Rummager) follow the implementation found in the LIRE [48] open source library.

the weighting schemes. Please note that the experimental results will be focused on the first two datasets (UKBench and UCID) for readability reasons. A more condense presentation of the results is followed for the Holidays and ZuBuD collections. The experimental results and the drawn conclusions are in line for all four employed datasets.

We prepared separate tables of results for the tested non-SIMPLE and SIMPLE descriptors. Table 2 presents the performances evaluated using MAP, of 7 Global Features' (GF) and 5 x (4 Codebooks) Local Features'(LF) descriptors from the literature all re-implemented and tested in the same retrieval system for fair comparison with the SIMPLE descriptors. The respective performance evaluations by P@4, P@10 and ANMRR can be found in [27].

Tested on the UKBench collection, the best performing non-SIMPLE descriptor with a MAP score of 0.8159 was the SURF LF descriptor with a codebook size of 512 VWs. This will be considered the "baseline" UKBench result for further reference. On the UCID image collection, CEDD, a global descriptor (as expected, due to the nature of the depictions in this dataset), with a MAP score of 0.6748, will be the baseline performance for comparison with our SIMPLE descriptors.

Tables 3 and 4 summarize the experimental results of all 16 proposed SIMPLE descriptors on the UKBench and the UCID dataset, respectively. The tables consist of four sub-tables, for the facilitation of the reader. Every sub-table shows the performance evaluations by MAP, P@k and ANMRR, per detector used, for all four descriptors, in all four codebook sizes. The weighting scheme (WS) reported in the tables was the highest performance among the 8 WS.

Table 2 Experimental results of reimplemented non-SIMPLE descriptors on UKBench and UCID.

UKBench Collection				UCID Collection			
Descriptor	**Size**	**WS**	**MAP**	**Descriptor**	**Size**	**WS**	**MAP**
SURF(baseline)	512	l.n.n	0.8159	CEDD(baseline)	Global		0.6748
CEDD	Global		0.8026	SURF	512	l.n.n	0.6513
SURF	128	l.n.n	0.7634	SIFT	512	l.n.n	0.6261
Oppo. SIFT	128	n.n.c	0.7475	SURF	2048	l.n.c	0.6259
Oppo. SIFT	512	n.n.c	0.7390	Oppo.SIFT	2048	n.t.c	0.6244
SIFT	512	l.n.n	0.6984	Oppo.SIFT	512	n.n.c	0.6072
SURF	2048	n.c.c	0.6911	SIFT	2048	n.n.c	0.6046
SIFT	128	l.n.n	0.6903	SURF	128	n.n.c	0.5927
SIFT	2048	n.n.c	0.6638	Oppo.SIFT	128	n.n.c	0.5872
Oppo. SIFT	32	n.n.n	0.6613	SIFT	128	n.n.c	0.5849
BTDH[50]	Global		0.6468	SURF	32	l.n.n	0.5492
SURF	32	l.n.n	0.6377	SIFT	32	n.n.c	0.5453
MPEG-7 CLD	Global		0.6181	MPEG-7 CLD	Global		0.5361
Oppo. SIFT	2048	n.t.c	0.5926	BTDH	Global		0.5353
SIFT	32	l.n.c	0.5683	MPEG-7 EHD	Global		0.5326
ORB	512	n.n.c	0.5371	Oppo.SIFT	32	n.n.n	0.5240
MPEG-7 EHD	Global		0.5271	MPEG-7 SCD	Global		0.4998
ORB	2048	n.t.c	0.4913	ORB	512	l.n.n	0.4929
ORB	128	n.n.c	0.4830	ORB	2048	n.n.c	0.4913
MPEG-7 SCD	Global		0.4716	ORB	128	n.n.c	0.4642
Color Hist.	Global		0.4133	BRISK	128	l.n.n	0.4636
BRISK	128	l.n.n	0.3904	BRISK	32	n.n.n	0.4532
ORB	32	n.n.n	0.3880	ColorHist.	Global		0.4443
BRISK	32	n.n.n	0.3550	Tamura	Global		0.4411
BRISK	512	l.n.n	0.3463	BRISK	2048	n.t.c	0.4360
Tamura[51]	Global		0.3130	ORB	32	n.n.c	0.4360
BRISK	2048	n.n.c	0.3096	BRISK	512	l.n.n	0.4345

Results on UKBench: Overall, 10 out of the 16 proposed SIMPLE descriptors managed to surpass the baseline experiment in this collection. In all cases, the best performing com-

Table 3 Experimental Results of all 16 SIMPLE descriptors on the UKBench dataset. MAP results in bold fonts mark performances that surpass the baseline performance. Underlined results mark the highest performance achieved per detector

		SIFT detector				**SURF detector**			
	Size	**WS**	**MAP**	**P@4**	**ANMRR**	**WS**	**MAP**	**P@4**	**ANMRR**
CEDD	2048	n.t.c	0.6402	0.5360	0.2769	l.t.c	**0.8280**	0.7580	0.1207
	512	l.t.c	0.8139	0.7710	0.1562	l.t.c	**0.8964**	0.8670	0.0879
	128	l.n.n	0.7911	0.7350	0.1695	l.n.c	**0.8665**	0.8260	0.1104
	32	n.n.n	0.6797	0.2741	0.2878	l.n.n	0.7806	0.7250	0.1771
SCD	2048	n.t.c	0.7195	0.6230	0.2058	l.t.c	**0.8730**	0.8180	0.0871
	512	l.t.c	**0.8696**	0.8350	0.1058	l.t.c	**0.9145**	0.8960	0.0713
	128	l.n.n	**0.8764**	0.8350	0.0961	l.t.c	**0.8941**	0.8640	0.0858
	32	l.n.n	0.7649	0.6890	0.1822	l.n.n	0.7956	0.7420	0.1672
CLD	2048	n.n.c	0.6327	0.5340	0.2847	l.n.c	0.7693	0.6890	0.1706
	512	l.t.c	0.7699	0.7180	0.1874	l.t.c	**0.8446**	0.8160	0.1333
	128	l.n.n	0.7649	0.7200	0.1950	l.n.n	0.8112	0.7640	0.1576
	32	n.n.n	0.5944	0.5430	0.3575	n.n.n	0.6857	0.6290	0.2725
EHD	2048	l.n.c	0.2712	0.2600	0.7052	n.n.c	0.4093	0.3720	0.5422
	512	l.n.c	0.2689	0.2560	0.7075	n.n.c	0.4276	0.4010	0.5321
	128	n.n.n	0.2708	0.2640	0.7054	n.n.n	0.3972	0.3760	0.5590
	32	n.n.n	0.2752	0.2640	0.6954	n.n.n	0.3570	0.3330	0.5987
		Rnd (600 samples)				**GaussRnd (600 samples)**			
	Size	**WS**	**MAP**	**P@4**	**ANMRR**	**WS**	**MAP**	**P@4**	**ANMRR**
CEDD	2048	l.t.c	**0.9183**	0.8890	0.0683	l.t.c	**0.9245**	0.9030	0.0655
	512	l.t.c	**0.9146**	0.8870	0.0707	l.t.c	**0.9227**	0.8940	0.0624
	128	l.n.c	**0.8892**	0.8460	0.0886	l.t.c	**0.8895**	0.8540	0.0894
	32	l.n.n	0.7993	0.7410	0.1632	l.n.n	0.7894	0.7300	0.1685
SCD	2048	l.t.c	**0.9268**	0.8980	0.0573	l.t.c	**0.9254**	0.9020	0.0608
	512	l.t.c	**0.9186**	0.8950	0.0674	l.t.c	**0.9218**	0.8930	0.0638
	128	l.t.c	**0.8876**	0.8420	0.0888	l.t.c	**0.8917**	0.8580	0.0865
	32	l.n.c	0.7884	0.7260	0.1704	l.n.c	0.8095	0.7560	0.1582
CLD	2048	l.t.c	**0.8831**	0.8480	0.1024	l.t.c	**0.8893**	0.8560	0.0926
	512	l.n.c	**0.8718**	0.8400	0.1069	l.t.c	**0.8715**	0.8360	0.1059
	128	l.n.c	**0.8184**	0.7730	0.1545	l.n.c	**0.8347**	0.7890	0.1345
	32	l.n.n	0.6455	0.5860	0.2978	l.n.n	0.6851	0.6200	0.2577
EHD	2048	l.t.c	0.6235	0.5780	0.3378	l.t.c	0.6185	0.5830	0.3425
	512	l.n.c	0.5629	0.5270	0.3919	l.n.c	0.5788	0.5310	0.3762
	128	l.n.n	0.4944	0.4610	0.4551	l.n.n	0.5053	0.4640	0.4408
	32	l.n.n	0.3166	0.3070	0.6017	n.n.c	0.4153	0.3810	0.5340

bination involved the SCD description method. When detecting patches using the SIFT detector, and due to the percentage of non-usable patches, only SIMPLE sft-SCD (which uses a descriptor that does not introduce minimum patch size limitations) manages to present a performance improvement, compared to the baseline. However, compared to their global equivalences, SIMPLE CEDD, SCD and CLD descriptors, perform vastly better. A degradation in performance is reported for SIMPLE sft-EHD. This leads to the assumption, that employing a detection mechanism that searches for interesting texture patches of one type, and then describes them with texture descriptors of another type, is an abortive attempt.

SIMPLE descriptors that employ the SURF detector, perform significantly better than SIFT. SIMPLE srf-SCD and srf-CEDD 512, in particular, achieve an almost perfect retrieval score for all evaluation metrics. Please note that, compared to their global equivalences, SIMPLE srf-(CEDD, SCD, CLD) perform comparable -if not better- even with a tiny codebook size of 32 VWs. SIMPLE srf-EHD, showed better results than the SIFT-based implementation, but still did not manage to surpass the EHD-global performance, corroborating the aforementioned assumption concerning texture based descriptors on texture based detectors.

Impressive results were obtained employing the Rnd and GaussRnd patches' generators. As reported in Table 3, we scored comparable performances to the SIMPLE SURF

based descriptors, and in many cases even outperformed those results with both generators. However, the last two implementations (Rnd and GaussRnd) are additionally much more efficient and light-weighted, since they strip the respective computational overhead that the detectors (SIFT and SURF) introduce. An increase in the performance of the SIMPLE rnd/gaussRnd-EHD descriptor is achieved. For the first time we managed to outperform the global-EHD score, on the respective collection.

Table 4 Experimental Results of all 16 SIMPLE descriptors on the UCID dataset. MAP results in bold fonts mark performances that surpass the baseline performance. Underlined results mark the highest performance achieved per detector

		SIFT detector				**SURF detector**			
	Size	**WS**	**MAP**	**P@10**	**ANMRR**	**WS**	**MAP**	**P@10**	**ANMRR**
CEDD	2048	l.t.c	0.6571	0.2134	0.3098	l.t.c	**0.7811**	0.2595	0.1892
	512	n.t.c	0.6636	0.2145	0.2961	l.t.c	**0.7635**	0.2531	0.2054
	128	l.n.n	**0.6813**	0.2252	0.2704	l.n.n	**0.7332**	0.2447	0.2260
	32	n.n.n	0.6088	0.1981	0.3455	l.n.c	0.6443	0.2141	0.3089
SCD	2048	l.t.c	**0.7045**	0.2332	0.2606	l.t.c	**0.7718**	0.2550	0.1968
	512	l.t.c	**0.7145**	0.2378	0.2457	l.t.c	**0.7648**	0.2515	0.2010
	128	l.n.n	**0.7065**	0.2351	0.2536	l.t.c	**0.7275**	0.2382	0.2355
	32	n.n.c	0.6354	0.2042	0.3196	l.n.n	0.6450	0.2095	0.3118
CLD	2048	l.t.c	0.6305	0.2107	0.3287	l.t.c	**0.7161**	0.2393	0.2502
	512	l.t.c	0.6304	0.2080	0.3304	l.n.n	**0.6765**	0.2225	0.2829
	128	l.n.n	0.6233	0.2023	0.3335	l.n.n	0.6291	0.2073	0.3288
	32	n.n.n	0.5243	0.1679	0.4344	n.n.n	0.5610	0.1809	0.3994
EHD	2048	l.n.c	0.4042	0.1130	0.5711	n.n.c	0.5030	0.1599	0.4600
	512	l.n.n	0.4044	0.1115	0.5724	n.n.c	0.5066	0.1576	0.4609
	128	n.n.c	0.4049	0.1126	0.5692	n.n.c	0.4973	0.1553	0.4644
	32	n.n.c	0.4062	0.1145	0.5632	n.n.c	0.4682	0.1450	0.4948
		Rnd 600 samples				**GaussRnd 600 samples**			
	Size	**WS**	**MAP**	**P@10**	**ANMRR**	**WS**	**MAP**	**P@10**	**ANMRR**
CEDD	2048	l.t.c	**0.7890**	0.2626	0.1756	l.t.c	**0.7955**	0.2672	0.1752
	512	l.t.c	**0.7745**	0.2527	0.1947	l.t.c	**0.7834**	0.2607	0.1797
	128	l.t.c	**0.7414**	0.2427	0.2194	l.t.c	**0.7367**	0.2447	0.2247
	32	l.n.n	0.6600	0.2183	0.2962	l.n.n	0.6725	0.2233	0.2852
SCD	2048	l.t.c	**0.7794**	0.2573	0.1892	l.t.c	**0.7876**	0.2611	0.1820
	512	l.t.c	**0.7610**	0.2534	0.2016	l.t.c	**0.7691**	0.2573	0.1950
	128	l.n.c	**0.7233**	0.2393	0.2382	l.t.c	**0.7400**	0.2427	0.2232
	32	l.n.c	0.6443	0.2118	0.3110	l.n.n	0.6565	0.2179	0.2963
CLD	2048	l.t.c	**0.7170**	0.2359	0.2481	l.t.c	**0.7191**	0.2408	0.2425
	512	l.t.c	**0.6781**	0.2176	0.2890	l.t.c	**0.6820**	0.2256	0.2800
	128	l.n.n	0.6375	0.2118	0.3191	l.n.n	0.6356	0.2065	0.3266
	32	l.n.n	0.5375	0.1763	0.4158	l.n.n	0.5560	0.1809	0.3975
EHD	2048	l.n.c	0.6557	0.2164	0.3057	l.n.c	0.6622	0.2198	0.2950
	512	l.t.c	0.6186	0.2061	0.3409	l.t.c	0.6407	0.2092	0.3194
	128	l.n.n	0.5666	0.1863	0.3920	l.n.n	0.5870	0.1931	0.3707
	32	n.n.c	0.5041	0.1573	0.4590	n.n.c	0.5037	0.1538	0.4770

Results on UCID: On the UCID collection, 11 out of the 16 proposed SIMPLE descriptors outperform the baseline non-SIMPLE descriptor.

In all cases, CEDD is involved in the best performing SIMPLE combinations, except when employing SIFT. Again, when SIFT is involved, the high percentage of non-usable patch sizes, leads to low performance scores for descriptors that introduce size limitations (CEDD has the highest limitation of minimum 40x40 pixels patches).

SURF-based, Rnd-Based and GaussRnd-Based sample strategies perform similarly, for all respective codebook sizes, when combined with CEDD, SCD or CLD. We would like to underline that in this collection, SIMPLE rnd/gaussRnd-EHD performances, not only present an impressive increase, but actually surpass the second-best non-SIMPLE descriptor (kindly refer to Table 2). This allows us to assume, that the efficient SIMPLE

rnd/gaussRnd-EHD descriptors, would prove to be competitive choices for similar datasets, where no colour information is available.

Tables 5 and 6 present in a ranked manner, the % improvement of the metrics MAP, P@k and ANMRR, that the proposed SIMPLE descriptors attained against the respective baseline non-SIMPLE descriptor. In order to keep the tables concise, we only included the top 10 SIMPLE descriptors that best the baseline results in both collections. On UKBench, 27 descriptors with varying codebooks surpassed the baseline MAP score of 0.8159. Nine of them managed to improve MAP by more than 12%, P@4 by more than 15% and ANMRR by an impressive more than 53%. On UCID, 28 SIMPLE descriptors achieved a higher MAP evaluation compared to the respective baseline (CEDD global) . The top six SIMPLE descriptors improved MAP by more than 15%, P@10 by more than 13% and ANMRR by more than 32%.

Table 5 The top 10 SIMPLE descriptors that surpass the baseline's MAP score on UKBench.

BASELINE UKBench						
Descriptor		**Size**	**WS**	**MAP**	**P@4**	**ANMRR**
SURF		512	l.n.n	0.8159	0.7730	0.1535
SIMPLE descriptors						
Descriptor		**Size**	**WS**	**%MAP Improvement**	**%P@4 Improvement**	**%AMNRR Improvement**
SIMPLE	rnd-SCD	2048	l.t.c	13.59	16.17	62.67
SIMPLE	gaussRnd-SCD	2048	l.t.c	13.42	16.69	60.39
SIMPLE	gaussRnd-CEDD	2048	l.t.c	13.31	16.82	57.33
SIMPLE	gaussRnd-CEDD	512	l.t.c	13.09	15.65	59.35
SIMPLE	gaussRnd-SCD	512	l.t.c	12.98	15.52	58.44
SIMPLE	rnd-SCD	512	l.t.c	12.59	15.78	56.09
SIMPLE	rnd-CEDD	2048	l.t.c	12.55	15.01	55.50
SIMPLE	rnd-CEDD	512	l.t.c	12.10	14.75	53.94
SIMPLE	srf-SCD	512	l.t.c	12.08	15.91	53.55
SIMPLE	srf-CEDD	512	l.t.c	9.87	12.16	42.74

Table 6 The top 10 SIMPLE descriptors that surpass the baseline's MAP score on UCID.

BASELINE UCID						
Descriptor		**Size**	**WS**	**MAP**	**P@10**	**ANMRR**
CEDD		Global		0.6748	0.2267	0.2823
SIMPLE descriptors						
Descriptor		**Size**	**WS**	**%MAP Improvement**	**%P@10 Improvement**	**%AMNRR Improvement**
SIMPLE	gaussRnd-CEDD	2048	l.t.c	17.89	17.87	37.94
SIMPLE	rnd-CEDD	2048	l.t.c	16.92	15.84	37.80
SIMPLE	gaussRnd-SCD	2048	l.t.c	16.72	15.17	35.53
SIMPLE	gaussRnd-CEDD	512	l.t.c	16.09	15.00	36.34
SIMPLE	srf-CEDD	2048	l.t.c	15.75	14.47	32.98
SIMPLE	rnd-SCD	2048	l.t.c	15.50	13.50	32.98
SIMPLE	rnd-CEDD	512	l.t.c	14.77	11.47	31.03
SIMPLE	srf-SCD	2048	l.t.c	14.37	12.48	30.29
SIMPLE	gaussRnd-SCD	512	l.t.c	13.97	13.50	30.92
SIMPLE	srf-SCD	512	l.t.c	13.34	10.94	28.80

Table 7 The Standard Deviation of the best performing MAP scores after multiple runs (5) of the SIMPLE rnd-SCD and gaussRnd-SCD, on both collections

Standard Deviation (600 samples)				
	UKBench		UCID	
Size	**Rnd**	**GaussRnd**	**Rnd**	**GaussRnd**
2048	0.004458	0.003389	0.002695	0.002743
512	0.005249	0.003422	0.003234	0.003206
128	0.006146	0.003767	0.003684	0.003595
32	0.008951	0.005187	0.004278	0.003801

Overall, the light-weighted and efficient combinations of rnd and gaussRnd detectors with SCD, CEDD and CLD descriptors dominated the top results for both collections. Concerning the random-patches techniques, please note, that every time we tested a descriptor for a given codebook size, newly extracted random patches were generated. In other words, for the presented results in Tables 3 and 4 under "Rnd (600 samples)" we generated 16 times, 600 random patches. The same applies for the GaussRnd experiments, as well. This strategy was chosen deliberately, in order to test the robustness of the random based implementations. However, we went on and further tested the robustness of these methods by repeating the SIMPLE rnd-SCD and gaussRnd-SCD experiment multiple times (five). The calculated standard deviations of the obtained MAP scores can be found in Table 7.

Table 8 MAP evaluations of SIMPLE rnd/gaussRnd-based descriptors, with 300 and 100 samples per image.

300 samples	UKBench		UCID	
	Rnd	**GaussRnd**	**Rnd**	**GaussRnd**
CEDD 512	0.9087	0.9102	0.7657	0.7663
SCD 512	0.9025	0.9130	0.7526	0.7533
CLD 512	0.8484	0.8506	0.6529	0.6576
100 samples	**Rnd**	**GaussRnd**	**Rnd**	**GaussRnd**
CEDD 512	0.8773	0. 8793	0.7394	0.7347
SCD 512	0.8770	0.8663	0.7421	0.7431
CLD 512	0.7086	0.7226	0.6019	0.6033

Studying the experimental results (Tables 3 and 4) and focusing on the two random sampling techniques, we can see that introducing the Gaussian distribution for the localization of the patches, allows for better performances in almost all combinations, which is more evident as the codebook sizes shrink. Moreover, the results in Table 7 suggest that the gaussRnd generator is a more robust approach, especially when employing small codebook sizes. On a last note, we would like to comment that, as for any detection method, both rnd and gaussRnd generators' robustness is subject to the employed images. However, when employing the UKBench collection, where the images are background clutter-free centred depictions of objects, the gaussRnd generator which samples for patches more densely in image centres, presents higher robustness compared to the rnd generator, whose standard deviation doubles as we move to smaller codebook sizes.

Finally, we experimented with lower numbers of generated samples for our SIMPLE descriptors that employ the rnd or gaussRnd patches' generators. We tested for 300 samples and 100 samples for combinations of rnd and gaussRnd with CEDD, SCD and CLD of 512 VWs codebooks, in both collections, evaluated by MAP. The experimental results that can be found in Table 8, show that even with half the samples the performances are directly comparable to those achieved when extracting 600 samples for the respective descriptors and codebook. What is more interesting, is that satisfying MAP evaluations are reported with as little as 100 image patches per image. However, we need to underline that these are early results that need to be extended for more combinations, codebook sizes and types of collections, in order to draw conclusive statements about appropriate sampling rates.

Wrapping up the results on the first two collections and in order to provide a wider perspective on the achieved retrieval performances we collected some of the best reported MAP scores for those collections. For the UKBench, our best performning descriptor **SIMPLE gaussRnd-SCD** scored a **0.9254 MAP** evaluation. Further methods from the literature implemented and tested under the same querying mode are SURF 16-VLAD with a

MAP=0.668, SIFT 64-VLAD MAP=0.804 [48], while some of the best reported methods on this collection are [19] with a MAP=0.8780, [52] with a MAP=0.9070 and [53] with MAP=0.9170.

In UCID, the best performing SIMPLE descriptor (**gaussRnd-CEDD**) achieves a MAP score of **0.7955**. SURF 64-VLAD has reportedly a MAP=0.6441 score, SIFT 64-VLAD a MAP=0.6933 [48] and Local- SIFT Global Search achieves a MAP=0.625 evaluation [1].

Results on Holidays and ZuBud: Tables 9 and 10, present the experimental results of the SIMPLE descriptors and results from methods from the literature on the two collections, respectively. The Holidays collection consists of images with diverse depictions of scenery, landmarks, objects etc. and presents rotation, viewpoint and illumination challenges. We could roughly say that it is a collection with characteristics that land in-between of those previously discussed datasets (UKBench and UCID). Again, the proposed descriptors achieve a great increase of the retrieval performance compared to the performances of the original methods they emerged from (Table 10: re-implemented). Furthermore, employing the random sampling strategies yield results that are directly comparable and often outperform some of the much more sophisticated and complex methods from recent literature (Table 10: reported in literature).

Finally, the ZuBuD collection which is depicting urban scenery, uses query images of smaller resolution, forcing descriptors that are not scale invariant to fail by default. Thus, the former global descriptors gain significantly when localized through the SIMPLE scheme. Furthermore, in this collection due to the specifics of the depictions (buildings photographed up-close) the two POIs detectors, SURF and SIFT, locate a much higher number of POIs compared to the other collections. However, even when dealing with images that present these repetitive patterns while also querying with smaller images, the random samplers preserve their robustness, that is now verified for all four collections.

Table 9 MAP evaluations of the SIMPLE descriptors, on the Holidays and ZuBuD collections.

		SURF		SIFT		Rnd (600)		GaussRnd (600)	
Holidays									
CEDD	512	l.n.n	0.7733	l.n.n	0.7441	l.t.c	0.8048	l.t.c	0.8039
	2048	l.t.c	0.7763	l.t.c	0.7335	l.t.c	0.8077	l.t.c	**0.8172**
SCD	512	l.n.n	0.7469	l.n.n	0.7506	l.t.c	0.7873	l.n.c	0.7807
	2048	l.t.c	0.7531	l.n.n	0.7375	l.t.c	**0.8042**	l.t.c	0.7968
CLD	512	l.n.n	0.7375	l.n.n	0.7094	l.t.c	0.7507	l.t.c	0.7506
	2048	l.t.c	0.7385	l.n.n	0.7126	l.t.c	**0.7651**	l.t.c	0.7629
EHD	512	n.n.c	0.6323	n.t.n	0.4919	l.n.c	0.6756	l.n.c	0.6732
	2048	n.n.c	0.6271	n.n.n	0.4872	l.n.c	**0.6816**	l.n.c	0.6789
ZuBuD									
CEDD	512	l.t.c	0.7901	l.n.c	0.6726	l.t.c	0.7675	l.t.c	0.7729
	2048	l.t.c	0.834	l.t.c	0.6854	l.t.c	**0.8338**	l.t.c	0.8287
SCD	512	l.n.c	0.697	l.t.c	0.5451	l.t.c	0.7585	l.t.c	0.7687
	2048	l.t.c	0.7453	l.t.c	0.5019	l.t.c	**0.8287**	l.t.c	0.8117
CLD	512	l.n.c	0.7921	n.t.c	0.5018	l.n.c	0.7529	l.n.c	0.7213
	2048	l.t.c	**0.8491**	n.t.c	0.5931	l.t.c	0.8011	l.t.c	0.7995
EHD	512	n.n.c	0.2398	l.n.c	0.0539	l.n.c	0.1659	l.n.c	0.1549
	2048	n.n.c	**0.2906**	l.n.c	0.0449	l.n.c	0.1815	l.n.c	0.1615

Analysing the weighting schemes' impact

When an image is processed by the BOVW model, a histogram of the VWs that it contains becomes its vector representation. This vector is weighted and normalized by the tf, df and $normalization$ variants of the weighting schemes (WS).

Table 10 MAP evaluations of state-of-the-art methods for the Holidays and ZuBuD collections.

Holidays			ZuBuD		
Re-implemented					
CEDD	global	0.7263	CEDD	global	0.7226
SCD	global	0.5369	SCD	global	0.3508
CLD	global	0.6480	CLD	global	0.5874
EHD	global	0.5551	EHD	global	0.3819
OppHist	global	0.6583	OppHist	global	0.5809
SIFT	BOVW-512/nnc	0.6914	SIFT	BOVW-2048/nnc	0.6240
SURF	BOVW-512/nnc	0.6777	SURF	BOVW-2048/nnc	0.6131
SIFT(V)	VLAD-64	0.7581	SIFT(V)	VLAD-64	0.7582
SURF(V)	VLAD-16	0.7169	SURF(V)	VLAD-64	0.6922
Reported in Literature					
Co-indexing[54]		0.8090	SIFT global search[2]		0.8130
Improving BoF [19]		0.8130	Color histogram[2]		0.7560
Asymmetric HE [55]		0.7940	LF patches histogram[2]		0.6470
Coupled Binary Embed[56]		0.7960	LF patches signature[2]		0.4260

The tf variant refers to the number of occurrences of a given VW in an image. Since the histogram calculated by BOVW is exactly that (i.e. VW frequencies in the image), when employing $n.*.*$ weighting schemes we do not alter the weighting factor based on tf. On the other hand, when employing $l.*.*$ weighting schemes we suggest that relevance does not increase proportionally with VW frequency. It is a well-known fact in information retrieval that a document with $tf = 10$ occurrences of a term is more relevant than a document with $tf = 1$ occurrence of the same term, but not ten times more relevant.

The df variant refers to the number of images in a collection that contain a given VW. When employing $*.n.*$ schemes we do not alter the vectors based on df. When using $*.t.*$ schemes, we suggest that when a VW is found in many images in the collection, then the VW is rather general and hence is given a smaller weighting factor.

For **the normalization** of the vectors, $*.*.n$ refers to "no normalization" while $*.*.c$ schemes normalize the descriptors using cosine similarity, so that all image vectors turn into unit vectors.

Figure 2 Weighting schemes' MAP scores on UKBench. SIMPLE SCD descriptor with all four detector for: a) codebook 2048, b) codebook 512, c) codebook 128, d) codebook 32

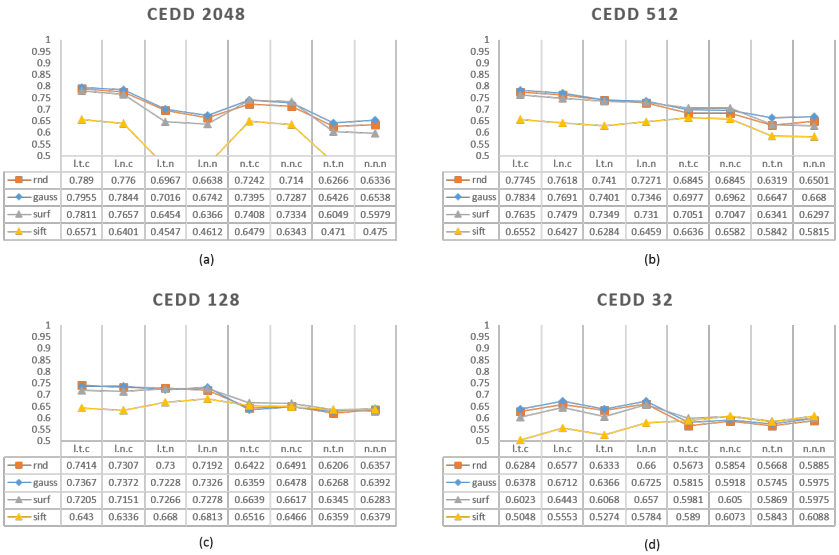

Figure 3 Weighting schemes' MAP scores on UCID. SIMPLE CEDD descriptor with all four detector for: a) codebook 2048, b) codebook 512, c) codebook 128, d) codebook 32

For the purposes of this paper, Figures 2 and 3 present the behaviour of the 8 WS of the best performing descriptor per collection (UKBench, UCID), combined with all four detectors, for the four different codebooks[5].

Beginning the analysis with the tf variant, we have observed that for smaller codebooks, the term "l" (log-weighted term frequency) behaves better. Small codebooks involve high term frequencies, making the use of the log frequency weight necessary. In larger codebooks, the use of the log frequency weight does not affect the results significantly. In Figures 2 [a] and 3 [a], where a large codebook is employed l.t.c scores comparable to n.t.c, l.n.c comparable to n.n.c, and so on. On the other hand, as the codebooks get smaller in graphs [c] and [d], weighting schemes that use the "l" term perform significantly better.

Regarding the df, for small codebooks due to the fact that there is a limited number of VW available for indexing, most VW are found in multiple images. Thus, employing the "t" term many VW are falsely credited with the same significance value and we notice a degradation in performance (in both collections, graphs [d] show that l.n.c performs better than l.t.c, the same for l.t.n and l.n.n, etc.). As the codebooks get larger the df does not seem to significantly alter the performances.

Finally, normalizing each vector by the cosine similarity so that all image vectors turn into unit vectors, seems to add to the performance of methods with large codebook sizes. This is justified by the fact that the use of larger codebooks produces descriptors with greater length than smaller codebooks. Thus the benefits of the normalization are more evident as the sizes grow.

Overall, the behaviour of the weighting schemes seems to be collection independent. Methods that utilize large codebooks can benefit by weighting the produced descriptors with an l.*.c weighting scheme or even an *.*.c scheme so as to reduce computational cost with a small discount performance-wise. On the other hand, for small codebooks an l.n.* weighting scheme will result in the best performance.

[5]resources in the form of spread sheets presenting all results are available upon request.

Large-scale experiments

The common practice [39, 56, 21, 19, 57] to evaluate large-scale image retrieval performance is to employ a large image database as distractors included in the retrieval database. This is a strategy that allows the evaluation of the scalability of a method overcoming the fact that there is not a publicly available large dataset with an assigned ground truth for CBIR. The evaluation of a method is based on the retrieved ranked list of images per query, compared to the initial collection's ground truth. This means that retrieved images that are part of the distractors are considered false results. The theme of the images used as distractors, their resolution and possible artifacts caused by their encoding can bias the evaluation.

With that being said, we populate the UKbench, UCID, Holidays, and ZuBuD datasets with a random fraction of 100,000 images (distractors) of the MIR Flickr 1M dataset [58]. The MIR Flickr dataset was chosen because it has relevant depictions to three out of our four initial collections (UCID, Holidays and to some degree with UKBench), it has the same encoding (JPG) with half our collections (UKBench and Holidays) and a resolution of the same order of magnitude with three of our datasets (only Holidays has a significantly higher resolution).

We test and evaluate the best performing descriptors (CEDD and SCD) with all extractor combinations (SIFT, SURF, Rnd, GaussRnd) in all four datasets. The codebooks were re-generated after randomly forwarding a 10% sample of extracted features from the combined collections (UKBench+MIRFlickr, UCID+MIRFlickr, etc) to the k-means classifier. This strategy ensures a more fair and realistic set-up, so as not to favour the description of images belonging to the initial collections.

Table 11 MAP evaluations for the large-scale experiments with MIR Flickr image distractors.

		UKBench				**UCID**			
		Cb	Dataset	+ Distract.	Loss	Cb	Dataset	+ Distract.	Loss
SIFT	CEDD	512	0.8136	0.8072	0.8%	128	0.6813	0.6218	8.7%
	SCD	512	0.8764	0.8208	6.3%	512	0.7145	0.6523	8.7%
SURF	CEDD	512	0.8964	0.8712	2.8%	2048	0.7811	0.6951	11.0%
	SCD	512	0.9145	0.8466	7.4%	2048	0.7718	0.6932	10.2%
Rnd	CEDD	2048	0.9183	0.9009	1.9%	2048	0.7890	0.7001	11.3%
	SCD	2048	0.9268	0.8869	4.3%	2048	0.7794	0.6981	10.4%
Gauss	CEDD	2048	0.9245	0.8956	3.1%	2048	0.7955	0.7028	11.7%
Rnd	SCD	2048	0.9254	0.8884	4.0%	2048	0.7876	0.7066	10.3%
		Holidays				**ZuBuD**			
		Cb	Dataset	+ Distract.	Loss	Cb	Dataset	+ Distract.	Loss
SIFT	CEDD	512	0.7441	0.7082	4.8%	2048	0.6854	0.6422	6.3%
	SCD	512	0.7506	0.7064	5.9%	512	0.5451	0.5173	5.1%
SURF	CEDD	2048	0.7763	0.7528	3.0%	2048	0.8340	0.7567	9.3%
	SCD	512	0.7531	0.7237	3.9%	2048	0.7453	0.6900	7.4%
Rnd	CEDD	2048	0.8077	0.7633	5.5%	2048	0.8338	0.7744	7.1%
	SCD	2048	0.8042	0.7462	7.2%	2048	0.8287	0.7626	8.0%
Gauss	CEDD	2048	0.8172	0.7545	7.7%	2048	0.8287	0.7731	6.7%
Rnd	SCD	2048	0.7968	0.7277	8.7%	2048	0.8117	0.7571	6.7%

Table 11 summarizes the experimental results per dataset. Overall the proposed descriptors present robust retrieval performances.

The average loss in performance for the UKBench, Holidays and ZuBuD collections is only 3.8%, 5.8%, 7.1% respectively, while even when challenged with distractors, the calculated performances in many cases exceed the baseline (non-SIMPLE) descriptors without

distractors. A higher loss is reported for the UCID collection with an average 10.3% degradation. However again, the absolute performances in the large scale experiments match or even exceed the performances of the baselines without distractors.

In order to test how the scalability of the proposed localized descriptors compares to that of the methods they originated from, we performed the large-scale scenarios for the original CEDD and SCD methods. CEDD reported a loss of 6.95% in UKBench, 15.64% in Holidays, 8.18% in ZuBuD and 20.96% in UCID. For the SCD descriptor the losses were 12.32%, 15.03%, 14.70% and 24.12%, respectively.

It is evident through the results that the retrieval accuracy of the proposed methods as the datasets scale up, not only remains sufficiently high in absolute numbers but, more importantly, also significantly outperforms the scalability of the original methods, validating the overall robustness and reliability of the scheme.

Discussion and future work

Through our experimental results we verified that the proposed scheme for localizing the discrimination ability of the compact MPEG-7 and MPEG-7-like global descriptors, is an effective strategy for CBIR. A significant boost of their retrieval performance is reported not only compared to their original global form but moreover, the proposed local features tested in the most straightforward retrieval model, perform comparably and even outperform some of the most recently proposed retrieval models that base their success in much more complex data manipulations.

Regarding the sampling strategies, we explored two different directions; first we employed two POIs detectors from the literature (SIFT and SURF) that search for salient textural information in an image, in multiple scales and then we introduced two different generators that randomly extract multiscale random image patches. Through the experimental results we observed that detection mechanisms based on texture saliency are successful when combined with descriptors that vectorize colour information, since they achieve colour description of POIs with textural attention. However, depending on the employed description method, this strategy can potentially suffer if the extracted patches are too small to be treated by the descriptors.

The success of the random generators, on the other hand, is most likely associated with the fact that in CBIR we are not always interested in one-to-one matching of points between images. We examined this allegation by employing four different image collections which vary both in depiction and in relevance association. In many cases the useful information is not constrained at textured image parts. Searching exclusively for salient texture parts, limits the retrieval effectiveness. Additionally, it was found that even though the distribution of POIs from blob detectors follows no particular pattern when seen per image, over a large number of images the overall distribution has a Gaussian-like behaviour. The random sampling strategies furthermore allow us to have much better control over the number of patches and their sizes, are light-weighted and can be adjusted depending on the available computational resources. Even though the tests conducted are preliminary at this stage, sampling with as little as 100 samples per image performs promisingly enough to be further examined. The number of extracted patches can affect vastly the overall usability of a method. Extracting a high number of patches per image (for instance following a dense sampling strategy) could make a method more robust but demands extensive use of memory and storing resources making it impractical for large scale retrieval scenarios.

Regarding the description parameters that should be selected for CBIR tasks, and although they are heavily subject to the images involved, we confirmed that quantized, compact representations of image features allow for better retrieval performances. The abstract representation allows for faster and safer comparisons of similarities between images because the discrete domain of features minimizes classification errors. Moreover, due to the massive amount of data that is usually involved in CBIR, compact descriptions are imperative when computational resources are limited.

Finally, weighting the descriptors with eight different weighting schemes and analysing their impact, gave us useful insight into the relationship of codebook sizes and local features. Having employed four very different kinds of image collections, four description methods and four different codebook sizes ranging from a tiny 32 VWs codebook up to a much wider 2048 VWs codebook, the experimental results suggest that the preferred weighting scheme strategy is collection and feature-type independent, and should be selected based on the size of the codebook. An other interesting direction worth exploring, left for future work, is testing the impact on the retrieval performance of different distance metrics. This type of investigation demands an in-depth study of multiple parameters such as the chosen representation (feature generation), the distribution of the data, the representation's dimensionality and the detected variance per dimension.

Currently we are expanding the SIMPLE family by varying the aggregation model and the description methods. More specifically, we employ the VLAD model as a BOVW alternative and test four different global descriptors that are evaluated based on their length, content and type of attributes their description is based upon. Early results confirm that global descriptors that are compact, quantized and carry color information are successfully localized through the SIMPLE scheme while the introduction of VLAD, although not outperforming the respective BOVW implementations achieves directly comparable performances with tiny codebooks of 16 or 64 clusters, eliminating simultaneously the need of applied weighting schemes.

Conclusions

In this paper, we explored, extended and simplified the SIMPLE family of local features' descriptors. We combined four sampling strategies, with four global features' descriptors, in a BOVW architecture and evaluated the produced descriptor in four diverse, popular image collections so as to (i) minimize the case that good achieved performances might have to do with specificities of the database and (ii) allow the comparison of the proposed method to many others from the literature that might have been left out in this work.

The primary scope of this study was to investigate how the parameters of a CBIR system (points-of-interest detection, description mechanisms, codebook sizes and weighting strategies) can best be selected to serve specifically for the needs of retrieval tasks. We built our design strategy keeping in mind the usability of the proposed descriptors in terms of scalability, compactness, efficiency and effectiveness and were rewarded with a set of very promising local feature descriptors that hit the mark on all of them.

We strongly encourage the incorporation of these light-weighted local features into different retrieval systems, the experimentation with collections varying in domain, relevance assumption or scale and overall the expansion of the SIMPLE family, and thus we provide open source implementations in C#, Java and MATLAB (http://tinyurl.com/SIMPLE-Descriptor). Furthermore, all descriptors are part of the LIRE library [48] and can be used under the GNU GPL license.

Acknowledgment

This work was supported by Lakeside Labs GmbH, Klagenfurt, Austria and funding from the European Regional Development Fund and the Carinthian Economic Promotion Fund (KWF) under grant KWF-20214/25557/37319. This research has been also co-financed by the European Union (European Social Fund-ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF)- Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

Author details

[1]Democritus University of Thrace, Department of Electrical and Computer Engineering, Xanthi, Greece.
[2]Klagenfurt University, Institute for Information Technology (ITEC) , Klagenfurt, Austria.

References

1. R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, no. 2, 2008.
2. T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: an experimental comparison," *Inf. Retr.*, vol. 11, no. 2, pp. 77–107, 2008.
3. O. A. B. Penatti and R. da Silva Torres, "Color descriptors for web image retrieval: A comparative study," in *SIBGRAPI*, 2008, pp. 163–170.
4. J. Annesley, J. Orwell, and J.-P. Renno, "Evaluation of mpeg7 color descriptors for visual surveillance retrieval," in *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*. IEEE, 2005, pp. 105–112.
5. K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, 2010.
6. S. Loncaric, "A survey of shape analysis techniques," *Pattern Recognition*, vol. 31, no. 8, pp. 983–1001, 1998.
7. D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1 – 19, 2004.
8. P. Howarth and S. M. Rüger, "Evaluation of texture features for content-based image retrieval," in *CIVR*, 2004, pp. 326–334.
9. C. G. Harris and J. Pike, "3d positional integration from image sequences," *Image and Vision Computing*, vol. 6, no. 2, pp. 87–90, 1988.
10. J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*. IEEE, 1994, pp. 593–600.
11. E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *ECCV (1)*, 2006, pp. 430–443.
12. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
13. H. Bay, T. Tuytelaars, and L. J. V. Gool, "Surf: Speeded up robust features," in *ECCV (1)*, 2006, pp. 404–417.
14. D. Scaramuzza, M. Achtelik, L. Doitsidis, F. Fraundorfer, E. B. Kosmatopoulos, A. Martinelli, M. W. Achtelik, M. Chli, S. A. Chatzichristofis, L. Kneip, D. Gurdan, L. Heng, G. H. Lee, S. Lynen, M. Pollefeys, A. Renzaglia, R. Siegwart, J. C. Stumpf, P. Tanskanen, C. Troiani, S. Weiss, and L. Meier, "Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in gps-denied environments," *IEEE Robot. Automat. Mag.*, vol. 21, no. 3, pp. 26–40, 2014. [Online]. Available: http://dx.doi.org/10.1109/MRA.2014.2322295
15. O. G. Cula and K. J. Dana, "Compact representation of bidirectional texture functions," in *CVPR (1)*, 2001, pp. 1041–1047.
16. X. Li and A. Godil, "Investigating the bag-of-words method for 3d shape retrieval," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, p. 5, 2010.
17. Y. Chen, X. Li, A. Dick, and R. Hill, "Ranking consistency for image matching and object retrieval," *Pattern Recognition*, vol. 47, no. 3, pp. 1349–1360, 2014.
18. J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
19. H. Jégou, M. Douze, and C. Schmid, "Improving bag-of-features for large scale image search," *International Journal of Computer Vision*, vol. 87, no. 3, pp. 316–336, 2010.
20. F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 143–156.
21. H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 304–317.
22. H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3304–3311.
23. L. Shao, D. Wu, and X. Li, "Learning deep and wide: A spectral method for learning deep networks," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 12, pp. 2303–2308, 2014.
24. F. Zhu and L. Shao, "Weakly-supervised cross-domain dictionary learning for visual recognition," *International Journal of Computer Vision*, vol. 109, no. 1-2, pp. 42–59, 2014.
25. L. Liu, M. Yu, and L. Shao, "Multiview alignment hashing for efficient image search," *Image Processing, IEEE Transactions on*, vol. 24, no. 3, pp. 956–966, 2015.

26. L. Shao, L. Liu, and X. Li, "Feature learning for image classification via multiobjective genetic programming," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 7, pp. 1359–1371, 2014.
27. C. Iakovidou, N. Anagnostopoulos, A. Kapoutsis, Y. Boutalis, and S. Chatzichristofis, "Searching images with mpeg-7 (and mpeg-7-like) powered localized descriptors: The simple answer to effective content based image retrieval," in *Content-Based Multimedia Indexing (CBMI), 2014 12th International Workshop*, June 2014, pp. 1–6.
28. B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 11, no. 6, pp. 703–715, 2001.
29. S. A. Chatzichristofis and Y. S. Boutalis, "Cedd: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval," in *ICVS*, 2008, pp. 312–322.
30. Savvas A. Chatzichristofis, Yiannis S. Boutalis, *Compact Composite Descriptors for Content Based Image Retrieval: Basics, Concepts, Tools*. VDM Verlag Dr. Muller, 2011.
31. E. Spyrou, H. L. Borgne, T. P. Mailis, E. Cooke, Y. S. Avrithis, and N. E. O'Connor, "Fusing mpeg-7 visual descriptors for image classification," in *ICANN (2)*, 2005, pp. 847–852.
32. A. R. Doherty, C. O. Conaire, M. Blighe, A. F. Smeaton, and N. E. O'Connor, "Combining image descriptors to effectively retrieve events from visual lifelogs," in *Multimedia Information Retrieval*, 2008, pp. 10–17.
33. M. M. Rahman, S. Antani, and G. R. Thoma, "A classification-driven similarity matching framework for retrieval of biomedical images," in *Multimedia Information Retrieval*, 2010, pp. 147–154.
34. S. A. Chatzichristofis and Y. S. Boutalis, "Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval," in *WIAMIS*, 2008, pp. 191–196.
35. M. M. Rahman, S. Antani, and G. R. Thoma, "A medical image retrieval framework in correlation enhanced visual concept feature space," in *CBMS*, 2009, pp. 1–4.
36. C. K. Dagli and T. S. Huang, "A framework for grid-based image retrieval," in *ICPR (2)*, 2004, pp. 1021–1024.
37. G. Amato, P. Bolettieri, F. Falchi, C. Gennaro, and F. Rabitti, "Combining local and global visual feature similarity using a text search engine," in *CBMI*, 2011, pp. 49–54.
38. A. Abdullah, R. C. Veltkamp, and M. A. Wiering, "Fixed partitioning and salient points with mpeg-7 cluster correlograms for image categorization," *Pattern Recognition*, vol. 43, no. 3, pp. 650–662, 2010.
39. D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *CVPR (2)*, 2006, pp. 2161–2168.
40. G. Schaefer and M. Stich, "Ucid: an uncompressed color image database," in *Storage and Retrieval Methods and Applications for Multimedia*, 2004, pp. 472–480.
41. S. A. Chatzichristofis, C. Iakovidou, Y. S. Boutalis, and O. Marques, "Co.vi.wo.: Color visual words based on non-predefined size codebooks," *IEEE T. Cybernetics*, vol. 43, no. 1, pp. 192–205, 2013.
42. H. Shao, T. Svoboda, and L. Van Gool, "Zubud-zurich buildings database for image based recognition," *Computer Vision Lab, Swiss Federal Institute of Technology, Switzerland, Tech. Rep*, vol. 260, 2003.
43. G. Zajić, N. Kojić, V. Radosavljević, M. Rudinac, S. Rudinac, N. Reljin, I. Reljin, and B. Reljin, "Accelerating of image retrieval in cbir system with relevance feedback," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, p. 062678, 2007.
44. E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "Orb: An efficient alternative to sift or surf," in *ICCV*, 2011, pp. 2564–2571.
45. S. Leutenegger, M. Chli, and R. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *ICCV*, 2011, pp. 2548–2555.
46. L. T. Tsochatzidis, C. Iakovidou, S. A. Chatzichristofis, and Y. S. Boutalis, "Golden retriever: a java based open source image retrieval engine," in *ACM Multimedia*, 2013, pp. 847–850.
47. S. A. Chatzichristofis, Y. S. Boutalis, and M. Lux, "Img(rummager): An interactive content based image retrieval system," in *SISAP*, 2009, pp. 151–153.
48. M. Lux and S. A. Chatzichristofis, "Lire: lucene image retrieval: an extensible java cbir library," in *ACM Multimedia*, 2008, pp. 1085–1088.
49. S. A. Chatzichristofis, C. Iakovidou, Y. S. Boutalis, and E. Angelopoulou, "Mean normalized retrieval order (mnro): a new content-based image retrieval performance measure," *Multimedia Tools and Applications*, pp. 1–32, 2012.
50. S. A. Chatzichristofis and Y. S. Boutalis, "Content based radiology image retrieval using a fuzzy rule based scalable composite descriptor," *Multimedia Tools Appl.*, vol. 46, no. 2-3, pp. 493–519, 2010.
51. H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 8, no. 6, pp. 460–473, 1978.
52. H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1169–1176.
53. X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han, "Contextual weighting for vocabulary tree based image retrieval," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 209–216.
54. S. Zhang, M. Yang, X. Wang, Y. Lin, and Q. Tian, "Semantic-aware co-indexing for image retrieval," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1673–1680.
55. M. Jain, H. Jégou, and P. Gros, "Asymmetric hamming embedding: taking the best of our bits for large scale image search," in *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 2011, pp. 1441–1444.
56. L. Zheng, S. Wang, and Q. Tian, "Coupled binary embedding for large-scale image retrieval," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 23, no. 8, 2014.
57. S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian, "Building contextual visual vocabulary for large-scale image applications," in *Proceedings of the international conference on Multimedia*. ACM, 2010, pp. 501–510.
58. M. J. Huiskes, B. Thomee, and M. S. Lew, "New trends and ideas in visual concept detection: the mir flickr retrieval evaluation initiative," in *Proceedings of the international conference on Multimedia information retrieval*. ACM, 2010, pp. 527–536.