Intelligent Content Retrieval using a Visual Vocabulary and Geometric Constraints

Evaggelos Spyrou, Yannis Kalantidis, Giorgos Tolias, Phivos Mylonas and Stefanos Kollias

Abstract— **During the last decades multimedia processing has emerged as an important technology to retrieve content based on similar data. Moreover, recent developments in the fields of high definition (HD) multimedia content and personal content collections (personal camcorders and digital still image cameras) tend to generate a huge volume of multimedia data everyday. Thus, the need for a meaningful, quick organization and access to generated content is now more than necessary; however, it still remains a rather difficult problem to be tackled both by humans and computers. In this paper we propose an intelligent extension of traditional image analysis methodologies towards more efficient digital content retrieval. The main idea is to extend local feature extraction methodologies by introducing additional geometrical constraints in the process. The proposed approach is tested and evaluated on a number of publicly available image datasets and results are very promising.**

I. Introduction

It is true that the end of the last decade has marked an era of ubiquitous connectivity and communication between people, devices and multimedia information. Community multimedia content collections such as Youtube[1] or Flickr[2] emerged to the everyday life of millions of people, allowing them to widespread personal multimedia data across the Internet. Vast amounts of new multimedia content is created daily on the Web and on personal computers, "transforming" ordinary human beings into heavy-duty content consumers. At this point the need for intelligent multimedia search and retrieval capabilities is becoming evident, considering that the produced content collections typically contain high-quality multimedia content, which is becoming harder and harder to access, manage and share.

The popularity of social networks and web-based personal image collections has resulted to a continuously growing volume of publicly available photos and videos. Nowadays, users are constantly uploading, describing, tagging and annotating their personal photos. Consequently, this growth of image collections has created the need for fast, robust and intelligent methodologies, able to analyze large-scale, diverse and rather heterogeneous visual content. As traditional keyword-based search engines give way to image-based, intelligent and context-aware engines, the need for quick, (semi-)automatic organization of content, boosted research efforts towards the direction of intelligent search and retrieval functionalities.

Human annotation or tagging of multimedia content in the form of accompanying metadata -nowadays used widely within social networks- forms a way to represent and handle the underlying knowledge. However, despite this clever human intervention, multimedia content remains highly unstructured and it is rather difficult to quickly extract important semantics from it. Consequently, based solely on additional textual information, it is hard to correlate raw multimedia content to other sources of information. The ultimate research goal remains to develop intelligent, (semi-)automated multimedia content analysis techniques to extract knowledge from the content itself.

The work presented herein forms an integrated approach that aims to retrieve visually similar content in a quick and effective manner. With respect to its technical implementation, we make use of traditional well-esteemed image analysis techniques, such as the construction and utilization of a visual vocabulary and a bag-of-words representation, in order to meaningfully describe the visual properties of the selected content under consideration. Moreover, geometric constraints are applied, in order to extend the bag-of-words model towards more accurate results, in terms of its efficiency and efficacy in the multimedia content retrieval process. It contributes to the current state-of-the-art techniques in single modality content processing for efficient information retrieval and takes a significant step in proposing a novel research methodology in the field.

The structure of this paper is as follows: Section II describes related work in the field of image retrieval, in order to present both the relation and the novelty of the presented approach in comparison to existing techniques. In Section III we discuss the low-level features we used, in order to capture the visual properties of images. In Section IV we describe our approach for creation of a visual vocabulary, that aims to quantize the extracted visual features and the technique we used for the indexing of images. Section V presents the algorithm we used in order to match the points extracted from two images, thus estimating their distance based on their low-level features. Section VI describes a step, that estimates the distance between two images in a more strict way, by exploiting the consistency of the locations of the visual features between the two images in comparison. Finally, experimental results on well-known datasets are presented in Section VII and conclusions are drawn in Section VIII.

The authors are with the Image, Video and Multimedia Systems Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, 9 Iroon Polytechniou str., 15773, Athens, Greece (phone: +30 210 7722491; emails: {espyrou, ykalant, gtolias, fmylonas}@image.ntua.gr, stefanos@cs.ntua.gr).

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

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

II. Related Work

In this work we try to bring together important elements of the content information retrieval research field with special focus on the visual part. In this section we attempt to shortly describe some important advances and point out relevant ongoing activities for each area. Amongst the basic building blocks of intelligent information retrieval are visual similarity techniques based on the extraction of scale- and/or rotation-invariant feature/interest points, leading to the construction of a suitable visual vocabulary and corresponding indexing of the image representation. In the following we provide a brief overview and discuss related research efforts conducted on the above topic.

A. Visual similarity

The easiest but less effective approach to efficient visual similarity is to extract visual features globally. In this way, the non-trivial task of selecting image parts or regions from which descriptors should be extracted is skipped. In [1], a similarity measure between images is presented. This measure is based on the Kullback–Leibler divergence between multidimensional probability density functions of wavelet coefficients grouped in coherent sets. Also, in [37], the global characteristics of an image are captured along with the local ones, by adaptively computing hierarchical geometric centroids of the image, called neighbourhoods. This method is applied to the problem of near-duplicate image retrieval.

Another approach which has been popular in previous years, is to extract descriptions from regions of the image, in an effort to capture local image characteristics and achieve better performance than global approaches, since in many cases two images may present the same global features, while locally are significantly different. Typically, knowledge for visual properties of regions is encoded in the form of a visual vocabulary. Each region is then assigned to a visual word. Numerous extensions of the bag-of-words approach have been recently proposed. For example, [28] explores techniques to map each visual region to a weighted set of words, allowing the inclusion of features which were lost in the quantization stage of previous systems. The set of visual words is obtained based on proximity in the descriptor space. In [8], images are segmented into regions and regions are classified into visual words, using a variety of features. Then a mapping between visual words and keywords is learned using the Expectation Maximization method. In [19], an approach for the linguistic indexing of images is presented, that uses Wavelets to extract image features and Hidden Markov Models (HMMs) to learn the association of those features to the keywords describing the images. In [4], the authors propose a randomised data mining method that finds clusters of spatially overlapping images. This unsupervised method is applied on large databases, finds clusters of similar regions and also is capable to retrieve near-duplicates of images. Moreover, the approach of [14] uses a visual words' description of images and then tries to create a more accurate description by using Hamming embedding and weak geometric consistency constraints.

B. Visual similarity based on points

While global extraction of features and local from regions presents good results in certain retrieval problems, as in the case of "object"-retrieval based applications, these techniques present serious limitations. Thus, most modern algorithms begin with the determination of interest points within an image. These points carry properties such as invariance to various image transformations, illumination etc. The methods continue by defining regions in the neighbourhood of these points and extract the descriptors within them. We should note here, that while some of the papers presented herein deal solely with object detection, the techniques mentioned are also important in the area of image retrieval, where the goal is to retrieve images based on the objects/places they contain. In [5], a representation of local image structure and a matching scheme, both insensitive to many appearance changes is presented. This method is applied to two-view matching of images from different modalities. Moreover, [9] presents a method to learn and recognise object class models from unlabelled and unsegmented cluttered scenes in a scale invariant manner. In this work, objects are modelled as flexible constellations of parts. A probabilistic representation is used for all aspects of the object: shape, appearance, occlusion and relative scale. An entropy-based feature detector is then applied, for region selection within the image. Also, in [10], object recognition is based on affine invariant regions. Segmentation and recognition are achieved simultaneously.

In [17], the problem of near-duplicate image retrieval is tackled with a parts-based representation of images using distinctive local descriptors extracted from points of interest, which are invariant under several transformations. Moreover, the work presented in [18], uses parts affinely rigid by construction. Object detectors are trained by identifying groups of neighbouring local affine regions whose appearance and spatial configuration remain stable across multiple instances. In [26], a novel feature matching method aims at efficiently tackling high-dimensional problems. The work presented in [29] is a large-scale object retrieval system. Therein, the query is a region of an image and the system retrieves images that contain the same object as the one contained in the user's query. In [32], the target is to identify the same rigid object or 3D location in different shots of a film, using invariant descriptors, that facilitate multiview matching. In [33], the authors suggest the use of local grey value invariants for retrieving images. These features are also computed in invariant points. Finally, in the same manner, in [23], a new method for detecting scale invariant interest points used for image indexing is presented.

III. Local Feature Extraction

For the representation of the visual content of a given image, a set of interest points is first selected and visual features are extracted locally, from their surrounding area. Since the goal is to choose scale invariant interest points,

their localization is carried out on a gaussian scale-space. In our system, the SURF (Speeded-Up Robust Features) [2] features have been selected to represent the visual properties of the images. These features have been proven to achieve high repeatability and distinctiveness. Apart from that, their extraction speed is high, when compared e.g. with the SIFT features [21]. An example of the extracted SURF features is depicted in Fig. 1.

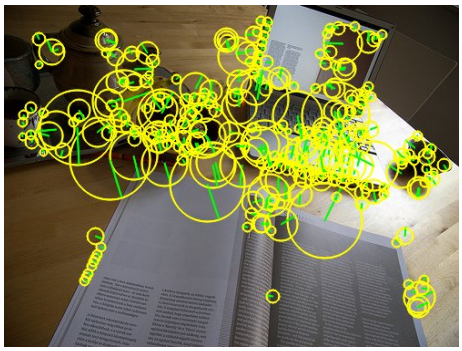


Fig. 1. Interest points extracted with the SURF Fast Hessian detector. The size of the yellow circle and the green line denote the scale and the dominant orientation, respectively.

For the detection of the interest points, a fast approximation of the Hessian matrix, which exploits the use of integral images is adopted. Then the local maxima of the Hessian matrix determinant are chosen as interest points. This blob response maxima process is carried out on several octaves of a Gaussian scale-space. The correct scale is automatically selected also from the Hessian determinant, as introduced in [20]. For exact point localization, an efficient non-maximum suppression algorithm is used at a $3 \times 3 \times 3$ intra-scale neighbourhood [25].

The SURF descriptor captures the intensity content distribution around the points detected with the aforementioned process. The first order Haar wavelet responses are computed with the use of integral images, resulting in a 64-dimensional feature vector. In order to achieve rotation invariance, a dominant orientation is determined. This is selected as the direction that maximizes the sum of the Haar-wavelet responses in a sliding window of size $\pi/3$ around the neighborhood of interest points.

To compute the descriptor, a square area around the interest point with $20 \times s$ side length is selected and divided in 4×4 blocks, with s denoting the interest point scale. Thus, the descriptor is also scale invariant. At each one of the 16 blocks, 4 values that correspond to the sum of the x,y, $|x|$ and $|y|$ first order Haar wavelet responses in a 5×5 grid in the block, are extracted. To make the descriptor robust to contrast changes, the descriptor vector is turned into a unit vector.

It is clear that the selection of the aforementioned low-level feature extraction scheme combines speed with robustness to scale, rotation and contrast changes. The fact that the extraction time is very small compared to other approaches allows the use of this scheme in real time image retrieval systems. Robustness to changes of the image guarantees that the system would be able to match two images depicting the same object under certain visual changes.

IV. Visual Vocabulary and Indexing

In this section we present the method we follow in order to create a visual vocabulary. The words contained in this vocabulary will be used for the representation of the visual properties of a given image. To understand the notion of a visual vocabulary, one should consider it as an equivalent to a typical language vocabulary, with an image corresponding to a part of a text. In the same way that text may be decomposed to a set of words, an image can also be decomposed to a set of *visual* words. Then, in order to compare two images, the sets of their corresponding visual words may be compared instead. Thus, it is interesting to create a visual vocabulary, in such a way that parts of images could be meaningfully assigned to visual words. We should note here that due to their polysemy, visual words cannot be as accurate as natural language words. This means that a given visual word cannot directly be assigned to a specific concept, but it can represent a part of a significantly large number of concepts.

A. Visual Vocabulary Construction

In order to create the visual vocabulary, a clustering approach is adopted. More specifically, the well-known K-means clustering algorithm [22] is applied on the SURF descriptors that correspond to a very large number of points of interest. If this set of points is significantly large, the clustering process using the K-means algorithm becomes a very slow and impractical task. For example, clustering of 5M of points (which is a typical amount of points extracted from 10000 images) requires a few days of processing. Even though clustering is an offline process, one should consider that in order to efficiently deal with large scale retrieval problems, the size of the vocabulary should be in the order of a few tenths of thousands of visual words as described in [15] and [30]. Thus, in order to rapidly create an appropriate vocabulary, the clustering process is performed on a smaller subset, carefully selected to contain the most representative images. After constructing the visual vocabulary, each image has to be represented with a description that captures its relation to all the words of it. We should also emphasize here that in order to create a visual vocabulary able to perform well in more than one domains, the images from which the regions of interest are extracted, have to be diverse and heterogeneous. Moreover the size of the visual dictionary has to be significantly large.

B. Nearest Neighbor search using a k-d tree

The goal is to describe a given image based on the set of the visual words it contains. This description will be in a vector form and will be denoted as *model vector*. In particular, for the formulation of a model vector, we need to determine the visual word that is the closest in terms of descriptor vector to each one of the image's points. To do

this fast and efficiently we select the k-d tree structure. The structure of k-d trees has been widely used in information retrieval [12], [3]. This data structure is a binary tree, which stores a finite number of k-dimensional data points and has been widely applied in the fields of computer learning [24] and neural networks [27]. Within the presented work, k-d trees are used in order to find the closest visual word of every point of interest, which is typically a very difficult and time consuming task due to the large dimension of points.

Given N k-dimensional elements, the k-d tree is constructed by partitioning the space iteratively, one dimension at a time. At each iteration, the feature space is divided into two subspaces along the selected dimension. This is repeated until each subspace contains a single point. This process creates a tree which allows a very fast search for all data points. The height of this tree is equal to $log(N)$.

In the case of the presented system, a k-d tree is created by the centroids of the clusters that are created by the clustering process. The dimension of the centroids is equal to 64. These centroids comprise the visual words of the visual vocabulary. This tree is created once and for all the images that we would like to index. Then, within the process of formulating the model vector for each point of the given image, its nearest neighbor is determined using the k-d tree.

C. Model Vector Formulation

After constructing the visual vocabulary, a given image is then represented in terms of it using a model vector. We assign the most similar visual word of the vocabulary to each descriptor of an image. Then, a histogram is constructed for each image, which counts the occurrences of the visual words of the vocabulary within it. If N_{vw} is the size of the visual vocabulary, the model vector mv_I that describes the visual content of a given image I is denoted by

$$mv_I = \left[tf_I(0), tf_I(1), \ldots, tf_I(N_{vw})\right] \ , \qquad (1)$$

where $tf_{I,i}$ denotes the number of times that the visual word i was selected as a nearest neighbor of one of the interest points extracted from image I. In order to find the closest visual word to a point, the aforementioned k-d tree structure is used.

The histogram of visual word appearance frequencies is then normalized and its non-zero values are stored in an index which resembles to the technique of inverted files, widely used in fast text retrieval [35],[36]. Each image is then represented by its corresponding visual words and the frequencies these occur. From this point, when it is mentioned that a visual word appears within an image, this would mean that this visual word is the nearest neighbor of one or more of the image's interest points.

When it is formed, based on large vocabularies of over $100K$ of visual words, the model vector is very sparse. The maximum number of non-zero values is at most equal to the number of the image's interest points in the extreme case when each of them is assigned to a different visual word.

Since this indexing process is inspired by techniques applied in the task of textual search, in addition to the term

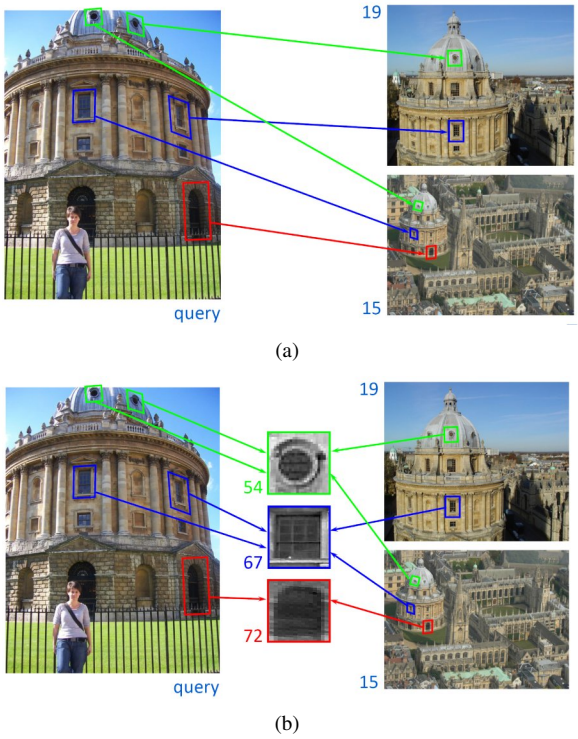


Fig. 2. Matching of two images without a visual vocabulary (2(a)) and with a visual vocabulary (2(b)).

frequency (tf), which is the frequency of a given term in a document, inverse document frequency (idf), can also be used. This case is studied in Section IV-D.

The process of querying an image database without and with a visual vocabulary is depicted in Fig. 2. In the first case the comparison of local descriptors is performed immediately for two images and after exhaustive comparisons in the whole database, the closest regions are found. In the latter case, for every image of the database all points have been assigned to appropriate visual words of the visual vocabulary. Thus, for a new query, its points have to be assigned to the closest visual words of the vocabulary. After this process, two images are considered to be similar if their points are assigned to similar visual words.

D. Inverse Document Frequency and Stop List

Inverse Document Frequency is another technique used in information and text retrieval [31], which during the last few years has been used for image retrieval, either along with language processing [16], or using visual dictionaries [34] [7].

The model vector of an image has taken, up to now, only the frequency of the appearance of a visual word as a nearest neighbor to any of the interest points into account. We define the *Inverse Document Frequency* or *idf* as

$$idf_k = log\frac{N}{n_k} \ , \qquad (2)$$

where N denotes the total number of the images of the given collection (the size of the database) and n_k the number of the

appearences of the visual word VW_k as the nearest neighbor to all points of all images in the database. Thus, idf acts as a weighting scheme, which identifies the most and less frequent visual words of the entire collection. The model vector can now be formulated as

$$v_I = \left[idf_0 \cdot f_I(0), \ldots, idf_K \cdot f_I(K) \right] , \tag{3}$$

with idf_k being the idf value of the visual word k.

It is obvious that the most common visual words, i.e. those with the smallest idf values, are not discriminative and their absence would facilitate the retrieval process. On the other hand, the rarest visual words are in most of the cases a result of noise and may distract the retrieval process. To overcome these problems, a stop list is created that includes the most and the least frequent visual words of the image collection. Using this list, the presence of its visual words is ignored, resulting to even sparser model vectors and thus to smaller image representations.

V. Matching

In order to compute the similarity between two given images, two similarity measures are used. The first is the inner product between the model vectors, which is a commonly used measure. Let v_Q and v_I be the model vectors for the query image Q and a database image I respectively. Then, their matching score can be computed as

$$s_2(Q, I) = \langle v_Q, v_I \rangle = \sum_{i=0}^{K} v_Q(i) v_I(i) , \tag{4}$$

where N_{vw} denotes the size of the visual vocabulary and $v_x(i)$ denotes the term frequency of the visual word i in image x.

The second similarity measure, that also proved to yield better results in practice, is the histogram intersection, discussed in [7]. Since the model vectors are histograms of visual words, the similarity between the model vector that corresponds to the query image and the one that corresponds to the database image is computed as

$$s_1(Q, I) = \sum_{i=0}^{K} \min(v_Q(i), v_I(i)) . \tag{5}$$

For both matching schemes and since the vectors are practically very sparse, the inverted file scheme is used in order to decrease matching time.

When a query is performed, first the local low-level features are extracted from the query image and its model vector is computed. Then the similarity of the query model vector with all database model vectors is computed, and the N most similar images, that is the images with the highest similarity values, are either returned to the user as similar, or become candidates for geometric consistency checking, as explained in Section VI.

VI. Geometric Consistency Checking

When the retrieval process considers only the model vectors that represent the visual content of images, it sometimes fails to produce accurate results, because the bag-of-features approach totally ignores the geometry of the extracted interest points. That is, because two images can contain similar visual words, which appear in a totally different spatial layout one from the other. Thus, the inclusion of a geometry consistence checking would be very useful. The method usually adopted is the RANSAC algorithm [11]. This method can determine the geometric transformation between two images given a set of tentative point-to-point correspondences, in presence of many false such correspondences that are also called *outliers*. In fact the RANSAC algorithm estimates the transformation that maximizes the number of *inliers* that is the set of correspondences that support the model. A modification of the RANSAC algorithm called Fast Spatial Matching[29] has been applied in this paper and is described in detail in section VI-A.

It is obvious that RANSAC relies a lot in the correspondences among points, which will be provided initially. These correspondences are not available, thus need to be calculated each time. A method that determines the nearest neighbors is not efficient, since it is a very time-consuming procedure that also needs to be computed online. However, we can exploit the correspondences between points and visual words, in order to create tentative point correspondences between two images. This requires an additional indexing process. Within this procedure, for every given image and each of its visual words, we store the locations of the corresponding points that yield these visual words as their nearest neighbors. We should note here that this process is very fast.

This procedure, however, introduces many false correspondences, due to the quantization effects of the bag-of-words approach. So, if, for example, a visual word appears 4 times in an image and 5 in another, then for this pair and with our method $4 \times 5 = 20$ correspondences will be formed, instead of the 4 correct ones. Taking this into account, we follow a rejection procedure, called *neighbor checking*. This method rejects correspondences between points whose neighborhoods do not match [34]. This means that in order for a tentative correspondence to be considered as valid, we require some of its spatially neighboring points to also have a valid correspondence between them. An example of RANSAC inliers between two images in the presence of partial occlusion is depicted in Fig. 3.

We choose not to apply this method in the whole database, but rather in the most similar images, in terms of their model vectors. The outcome of RANSAC, which is the number of inliers, is used to filter the most similar images and *re-rank* them. This approach appears to decrease the number of false positives. Using an appropriate threshold on the retrieved results, a higher precision can be easily achieved.

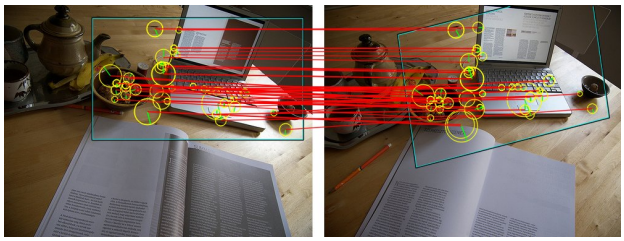


Fig. 3. The inliers found by RANSAC between two images.

A. Spatial verification using Fast Spatial Matching

The well known RANSAC algorithm is applied between two images in order to determinate the geometric transformation between them, in presence of outliers. We use a deterministic modification of RANSAC to determine the affine transformation between a pair of two images. Given a set of correspondences between two images, i.e., the pairs $x_i \leftrightarrow x_i'$, we can define the affine transformation matrix H by

$$x_i' = Hx_i \ . \tag{6}$$

While in the original RANSAC algorithm 3 correspondences are needed to create an initial hypothesis for affine transformation, in Fast Spatial Matching an initial hypothesis is created by single correspondences of SURF features leading to an initial similarity transformation between the two images. Using single correspondences for hypothesis generation makes it feasible to evaluate all possible transformations based on the tentative correspondences, thus removing the randomness of the algorithm [29]. The geometric model with the highest number of inliers so far is kept after each iteration. When a better model is found, then using the inliers of the similarity transformation, the Local-Optimization step [6] is executed in order to determine an affine transformation in the least-squares sense.

Given the correspondences of the points between two images, the algorithm is summarized as follows:

- Randomly select a tentative correspondence, from the set of the ones not selected so far. This is a correspondence between two circular regions found by the SURF detector $C \longleftrightarrow C'$.
- Based on $C \longleftrightarrow C'$ and on the transforms $H1$, H_2 which will transform C, C' correspondingly to the unit circle, the overall similarity transformation is $H = H_2^{-1}H_1$.
- Using H, calculate the symmetric transfer error E_i[13] for each correspondence and find the set I for which $E_i < \theta$. This is the set of inliers.
- If $|I|$ is the highest so far, then use the Local-Optimization, solve for affine transformation and use it to recalculate the set of inliers.

It is appropriate to notice that although our method was initially based on scale and rotation invariant regions for an initial transformation hypothesis (which is a similarity transformation), we finally calculated the affine transformation using Local-Optimization.

VII. Experimental Results

First we present the datasets we used in order to evaluate the proposed method. All of the datasets are annotated and publicly available, namely:

- UK Bench: This dataset contains 10000 images of 2500 objects. 4 pictures correspond to each object, taken from different angles. A sample of this dataset is depicted in Fig. 4(a).
- Zurich Buildings (Zurich1): This dataset is a collection of images of buildings in Zurich. It contains 1005 images from 201 different buildings. A sample of this dataset is depicted in Fig. 4(b).
- Zurich Buildings with distractors (Zurich2): The Zurich Bulidings Dataset, augmented with 5000 "distractor" images. i.e. images that should not be returned by any query.
- Caltech: We used a subset from the Caltech101 database which contained 1025 images from 8 categories (zebra, car, racket, tennis, insects, cows, airplanes, motorcycles and Eiffel tower). A sample of this images is depicted in Fig. 4(c)

We constructed a visual vocabulary of 10000 visual words and for the evaluation of the presented work, we utilized the *mean Average Precision* (*mAP*) metric. Results are summarized in Table I. In general, we observe robust retrieval results, with high precision rates on the top-ranked images' list.

Best results were achieved at the Zurich Buildings datasets. There, we observe that although a large amount of distractors were added, the performance remained significantly high. We may notice that the performance in the Caltech dataset appears poor, in comparison to the other datasets. However, we should emphasize that images of each category of this dataset belong to the same object class and not to the same object, while in the other datasets they belong to multiple views of the same object.

Overall, the performance of the proposed technique appears very promising and the experimental results indicate its robustness in diverse data sets.

TABLE I

Mean Average Precision values for the aforementioned datasets. mAP(GC) and mAP are estimated with and without Fast Spatial Matching, respectively.

Dataset	**Size** (images)	**mAP(GC)**	**mAP**
UK Bench	10500	0.55	0.33
Zurich1	1005	0.80	0.57
Zurich2	6005	0.73	0.42
Caltech	1025	0.42	0.35

VIII. Conclusions

The work presented in this paper is an approach that manages to efficiently retrieve visually similar multimedia

content. We utilized traditional analysis techniques, like construction and utilization of a visual vocabulary and a bag-of-words representation, in order to meaningfully describe the visual properties of the selected content. We then applied geometric constraints in order to extend the model and obtained more accurate results in the retrieval process. Future work will include application to larger image datasets and tackling of computational issues and limitations of the proposed representation model, so that this is used for efficient object recognition.

ACKNOWLEDGMENTS

This research was partially supported by the FP7 ICT Integrated Project WeKnowIt (contract FP7-215453).

REFERENCES

[1] S. Anthoine, E. Debreuve, P. Piro, and M. Barlaud. Using neighborhood distributions of wavelet coefficients for on-the-fly, multiscale-based image retrieval. In *Image Analysis for Multimedia Interactive Services, 2008. WIAMIS'08. Ninth International Workshop on*, pages 28–31, 2008.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, 2008.

[3] J.L. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[4] O. Chum and J. Matas. Web scale image clustering: Large scale discovery of spatially related images. Technical report, Technical Report CTU-CMP-2008-15, Czech Technical University in Prague, 2008.

[5] O. Chum and J. Matas. Geometric hashing with local affine frames. In *Computer Vision and Pattern Recognition*, volume 1, pages 879–884, 2006.

[6] O. Chum, J. Matas, and J. Kittler. Locally optimized ransac. *Lecture Notes in Computer science*, pages 236–243, 2003.

[7] O. Chum, J. Philbin, and A. Zisserman. Near duplicate image detection: min-hash and tf-idf weighting. In *Proceedings of the British Machine Vision Conference*, 2008.

[8] P. Duygulu, K. Barnard, JFG De Freitas, and D.A Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. *Lecture Notes in Computer science*, pages 97–112, 2002.

[9] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Computer Vision and Pattern recognition*, volume 2, pages 1575–1589. IEEE Computer Society; 1999, 2003.

[10] V. Ferrari, T. Tuytelaars, and L. Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *International Journal of Computer Vision*, 67(2):159–188, 2006.

[11] M.A. Fischler and R.C. Bolles. Random sample consensus - a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[12] J.H. Freidman, J.L. Bentley, and R.A. Finkel. An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Mathematical Software (TOMS)*, 3(3):209–226, 1977.

[13] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[14] H. Jegou, M. Douze, and C. Schmid. Hamming embedding and weak geometric consistency for large scale image search. *ECCV, Oct*, 2008.

[15] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In Andrew Zisserman David Forsyth, Philip Torr, editor, *European Conference on Computer Vision*, LNCS. Springer, oct 2008. to appear.

[16] M.A.S.T. Kanade. Video Skimming and Characterization through the Combination of Image and Language Understanding. In *Proceedings of the 1998 International Workshop on Content-Based Access of Image and Video Databases (CAIVD'98)*, page 61. IEEE Computer Society Washington, DC, USA, 1998.

[17] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate detection and sub-image retrieval. In *ACMA C Multimedia*, volume 9, pages 869–876. IEEE Computer Society; 1999, 2004.

[18] S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. In *British Machine Vision Conference*, volume 2, pages 959–968, 2004.

[19] J. Li and J.Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 2:1075–1088, 2003.

[20] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Print on Demand, 1994.

[21] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[22] J. MacQueen. B. 1967.Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 1–297.

[23] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Proc. ICCV*, volume 1, pages 525–531, 2001.

[24] A.W. Moore. An intoductory tutorial on kd-trees. Technical report, Technical Report.

[25] A. Neubeck and L. Van Gool. Efficient Non-Maximum Suppression. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, 2006.

[26] D. Omercevic, O. Drbohlav, and A. Leonardis. High-dimensional feature matching: Employing the concept of meaningful nearest neighbors. In *Ieee 11Th International Conference on Computer Vision*, pages 1–8, 2007.

[27] S.M. Omohundro. Efficient algorithms with neural network behavior. *Complex Systems*, 1(2):273–347, 1987.

[28] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization improving particular object retrieval in large scale image databases. *Image*, 9(14):15–17.

[29] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. C V P R*, volume 3613, pages 1575–1589, 2007.

[30] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007.

[31] Y. Rui, TS Huang, and S. Mehrotra. Content-based image retrieval with relevance feedback in MARS. In *Image Processing, 1997. Proceedings., International Conference on*, volume 2, 1997.

[32] F. Schaffalitzky and A. Zisserman. Automated location matching in movies. *Computer Vision and Image Understanding*, 92(2-3):236–264, 2003.

[33] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *PATTERN ANALYSIS and MACHINE INTELLIGENCE*, 1(6):530–535, 1997.

[34] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 1470, Washington, DC, USA, 2003. IEEE Computer Society.

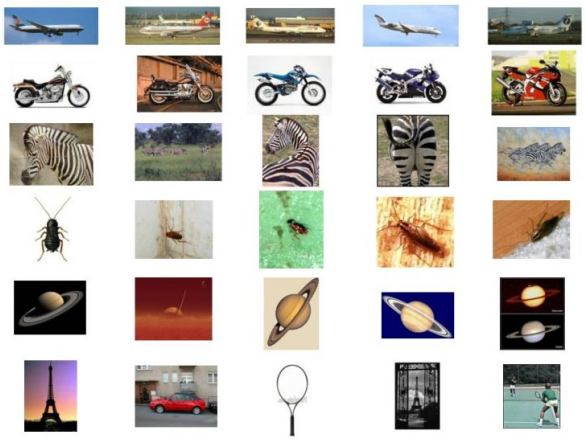
[35] D.M.G. Squire, W. Müller, H. Müller, and T. Pun. Content-based query of image databases: inspirations from text retrieval. *Pattern Recognition Letters*, 21(13-14):1193–1198, 2000.

[36] I.H. Witten, A. Moffat, and T.C. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann, 1999.

[37] M. Yang, G. Qiu, J. Huang, and D. Elliman. Near-duplicate image recognition and content-based image retrieval using adaptive hierarchical geometric centroids. In *Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)-Volume 02*, pages 958–961. IEEE Computer Society Washington, DC, USA, 2006.

(a)

(b)



(c)

Fig. 4. Sample images of the UK Bench, Zurich Buildings and Caltech datasets.