Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

A Study on the Use of a Binary Local Descriptor and Color Extensions of Local Descriptors for Video Concept Detection

Foteini Markatopoulou[1,2], Nikiforos Pittaras[1], Olga Papadopoulou[1], Vasileios Mezaris[1], and Ioannis Patras[2]

[1] Information Technologies Institute (ITI), CERTH, Thermi 57001, Greece
{markatopoulou,npittaras,olgapapa,bmezaris}@iti.gr
[2] Queen Mary University of London, Mile end Campus, UK, E14NS
i.patras@qmul.ac.uk

Abstract. In this work we deal with the problem of how different local descriptors can be extended, used and combined for improving the effectiveness of video concept detection. The main contributions of this work are: 1) We examine how effectively a binary local descriptor, namely ORB, which was originally proposed for similarity matching between local image patches, can be used in the task of video concept detection. 2) Based on a previously proposed paradigm for introducing color extensions of SIFT, we define in the same way color extensions for two other non-binary or binary local descriptors (SURF, ORB), and we experimentally show that this is a generally applicable paradigm. 3) In order to enable the efficient use and combination of these color extensions within a state-of-the-art concept detection methodology (VLAD), we study and compare two possible approaches for reducing the color descriptor's dimensionality using PCA. We evaluate the proposed techniques on the dataset of the 2013 Semantic Indexing Task of TRECVID.

Keywords: Video feature extraction, concept detection, concept-based video retrieval, binary descriptors.

1 Introduction

Concept-based video annotation and indexing is a very important task for the multimedia analysis field and a significant part of applications such as video retrieval, video event detection and video hyperlinking [27], [17]. A typical video concept detection system consists of three main modules: the video decomposition module, where video sequences are segmented into shots and each shot is represented by e.g. one or more characteristic keyframes/images; the feature extraction module, where features (e.g. local image descriptors, motion descriptors) are extracted from the visual information and encoded into a descriptor vector; and, finally the learning module, which employs machine learning algorithms in order to solve the problem of associating image descriptor vectors and concept labels.

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

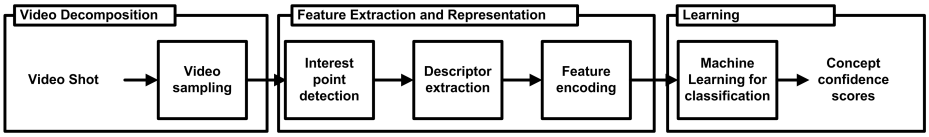

Fig. 1. Block diagram of a typical concept detection system

In this work we focus on the feature extraction process. Scale Invariant Feature Transform (SIFT) [16] and Speeded Up Robust Features (SURF) [2] are probably the two local descriptors that are most-widely used for this task. However, they are non-binary descriptors, which makes them not so suitable for applications requiring the transmission of descriptor vectors. For example, when considering a mobile application where pictures are taken with a mobile device and local descriptors from these pictures need to be sent to a server for further processing, then it is very important that the local descriptors are as compact as possible, to minimize transmission requirements [8]. ORB (Oriented FAST and Rotated BRIEF) [22] is a binary local descriptor, which was originally proposed for similarity matching between local image patches. We examine ORB in the task of video concept detection, and we show that it constitutes a viable alternative to the non-binary descriptors currently used in this task, while its compact size and low storage needs make this descriptor appealing for mobile applications. Subsequently, inspired by two color extensions of SIFT [24], namely RGB-SIFT and OpponentSIFT, we define the corresponding color extensions for the two other local descriptors considered in this work (SURF, ORB), and we show that this relatively straightforward way of introducing color information is in fact a generic methodology that works similarly well for different local descriptors. In addition, we present a different way of performing Principal Component Analysis (PCA) [28] for feature reduction, which improves the results of SIFT/SURF/ORB color extensions when combined with VLAD encoding. Our experiments were performed on the TRECVID 2013 Semantic Indexing (SIN) dataset [19], which consists of a development set and a test set (approximately 800 and 200 hours of internet archive videos for training and testing, respectively).

The rest of this paper is organized as follows: Section 2 reviews related work, focusing on local image descriptors. Section 3 discusses how the binary ORB descriptor can be used for video concept detection. Section 4 introduces the color extensions of SURF and ORB, while Section 5 discusses two possible approaches of employing PCA for color descriptors. Section 6 presents our experiments and results, and finally Section 7 summarizes our main conclusions.

2 Related Work

Figure 1 summarizes a typical concept detection system. The video decomposition module uses shot segmentation algorithms in order to divide the initial

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

video sequence into shots, and then possibly also single-out a subset of the visual information (e.g. keyframes, tomographs [26]) to be used for further processing. Then, the feature extraction module deals with the extraction of meaningful feature vectors to represent each piece of visual information. A variety of visual, textual and audio features can be extracted to this end; a review of different types of features can be found in [27]. In large-scale video concept detection, typically local image features are utilized, being extracted from representative keyframes or similar 2D image structures [26]. Two of the most popular local descriptors are SIFT [16] and SURF [2]. Both of them extract features that are invariant to rotation, scale and illumination variations, while SURF extraction is somewhat less computationally-demanding (SURF is two times faster than SIFT according to [2]). SIFT and SURF construct vectors of floating-point values (which are often quantized to integers in the range [0,255]). For many modern applications, though, e.g. concept detection on mobile devices, small-sized yet discriminative descriptors are very important in order to extract, store and transmit them efficiently (e.g. send local descriptors to a server for performing concept detection). Binary local descriptors are an attractive alternative to non-binary descriptors such as SIFT and SURF, generating binary strings which can be computed efficiently while also requiring lower storage space. BRIEF [5], ORB [22], BRISK [15], and FREAK [1] are some examples of binary local descriptors that have been proposed for similarity matching between local image patches. They are all based on calculating the differences between pairs of pixel intensity values within an image patch; what distinguishes them is the pattern they follow in order to perform these pair-wise pixel comparisons. Studies show that ORB is among the most accurate binary descriptors for image matching [6]. The possibility of using ORB in image classification was also briefly examined in [12].

The above mentioned non-binary and binary local descriptors are intensity-based: they are applied to grayscale images (e.g. an RGB image is firstly converted to grayscale), and the extracted features are calculated from the pixel intensity values. Two color variants of SIFT, namely RGB-SIFT and OpponentSIFT, that increase the illumination invariance, the discriminative power and also make the descriptor invariant to light color changes were proposed in [24]. Methods that consider the color information in order to improve the SURF descriptor have also been proposed, but were examined only on the image matching problem [11], [10], [9]. For example, [10] calculates a color local kernel histogram in the neighborhood of each keypoint and concatenates it with the original SURF descriptor that has been extracted from the pixel intensity values of the same neighborhood. In [12], the extraction of ORB from all three color channels of the RGB color space was considered.

For the purpose of visual concept detection, local descriptors extracted from different patches of one image are subsequently aggregated into a global image representation, a process known as feature encoding. The most popular encoding in the last years has been the Bag-of-Words (BoW) [21]. Fisher vector (FV) [20], Super Vector (SV) [30] and VLAD (Vector of Locally Aggregated Descriptors) [13] are three state-of-the-art encodings that significantly outperform the

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

BoW [25] [7]. FV encoding describes the difference between the distribution of features for an image and the distribution fitted to the features of all the training data. VLAD [13] is a fast approximation of FV that performs somewhat worse but is more compact and faster to compute [14], which makes it a good compromise. SV [30] works in the same lines, however requires larger codebooks than VLAD and FV in order to exhibit similar levels of accuracy, which increases the memory and computation requirements. The three latter encodings are high-dimensional and their dimensionality is affected by the dimensionality of the local descriptors they encode, thus dimensionality reduction approaches such as PCA are widely used for making the image representation more compact prior to learning/classification. Dimensionality reduction can be performed at two stages: local descriptors can be reduced prior to the encoding, and then the final encoding can also be further compacted [14].

Finally, for learning the associations between the image representations and concept labels, algorithms such as Logistic Regression (LR) and Support Vector Machines (SVM) are typically trained separately for each concept, on ground-truth annotated corpora. Then, when a new unlabeled video shot arrives, the trained concept detectors will return confidence scores that show the belief of each detector that the corresponding concept appears in the shot. This baseline learning process can be further improved in different ways, e.g. by taking into account concept correlations instead of training each detector independently [18].

3 Using a Binary Local Descriptor for Concept Detection

ORB [22] is a binary local image detector and descriptor that presents similar discriminative power with SIFT and SURF in image matching problems, it has similar properties such as invariance in rotation, scale and illumination, but at the same time is more compact and faster to be computed. A 256-element binary ORB vector requires 256 bits to be stored; in contrast, an integer-quantized 128-element SIFT vector requires 1024 bits. In addition, according to [22], ORB is an order of magnitude faster than SURF to compute, and more than two orders of magnitude faster than SIFT.

There is not a single way for introducing binary descriptors in the visual concept detection pipeline. [12] did so by considering the BoW encoding, and proposed a modified K-means algorithm (the "K-majority" algorithm) for generating the codebook (vocabulary) of BoW, that would result in a binary codebook. To illustrate the modifications, Algorithm 1 presents the steps of the original K-means clustering algorithm. In order to create a binary codebook, [12] used the Hamming distance in Step 2 of the Construction stage and also in the Assignment stage, while in Step 3 they used their "K-majority" voting method in order to calculate a binary cluster center.

In this work we claim that a binary descriptor (ORB) can be used for the video concept detection in the same way as its non-binary counterparts. Specifically, let us assume that I is a set of images and x_i $i = 1, ..., N$ are ORB descriptors extracted from I, where $x_i \in \{0,1\}^d$. N is the total number of ex-

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

Algorithm 1 Steps of K-means algorithm

Codebook construction:
1. Randomly initialize a set of K cluster centers w_k
2. For each descriptor vector x_i, compute index k_i of the cluster centre nearest to x_i
3. Update the cluster centers w_k
4. Repeat steps 2 and 3 until convergence

Word assignment:
Given a new local descriptor vector x', assign it to the nearest cluster w_k

tracted local descriptors and d is the dimension of the ORB descriptor. From these binary descriptors, we generate a floating-point codebook of K visual codewords $w_k \in \mathbb{R}^d$, $k = 1, ..., K$, using a standard K-means. The distances between the binary ORB descriptors and the codewords (Construction: Step 2 and Assignment stage of Algorithm 1) are calculated by the L2 norm. In Step 3 of Algorithm 1, averaging is also performed as in the original K-means (calculating the mean of a set of vectors).

Assigning binary local descriptors to a binary codebook using the hamming distance, as in [12], is faster than assigning them to a floating-point codebook using the L2 distance. However, considering that for the concept detection problem the time needed for the assignment to codebook is negligible compared to other processes of the pipeline (e.g. feature encoding, classification), more important is what leads to a more discriminative codebook that improves the concept detection accuracy. We report results of comparing these two codebook creation strategies (that of [12] and the one described in this section) in Section 6.

4 Color Extensions of Binary and Non-binary Local Descriptors

Based on the good results of two color extensions of SIFT, namely RGB-SIFT and OpponentSIFT [24], we examine the impact of using the same methodology for introducing color information to other descriptors (SURF, ORB). Our objective is to examine if this is a methodology that can benefit different local descriptors and is therefore generally applicable.

Let d denote the dimension of the original local descriptor (typically, d will be equal to 64 or 128 for SURF and 128 or 256 for ORB). Figure 2 summarizes the process of extracting RGB-SURF, RGB-ORB, OpponentSURF and OpponentORB descriptors. An RGB image has three 8-bit channels (for red, green and blue). The original non-color local descriptors are calculated on 8-bit grayscale images, so they first transform the RGB image to grayscale. In contrast to this, our RGB-SURF/ORB (Fig. 2:(a)) apply the original SURF or ORB descriptors directly to each of the three R, G, B channels and for each keypoint extract three d-element feature vectors. These are finally concatenated into one $3 \cdot d$-element feature vector, which is the RGB-SURF or RGB-ORB descriptor vector.

Similarly, our OpponentSURF/ORB (Fig. 2:(b)) descriptors firstly transform the initial RGB image to the opponent color space [24]. We refer to the transformed channels as O_1, O_2 and O_3. O_3 is the luminance channel, i.e. the one

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

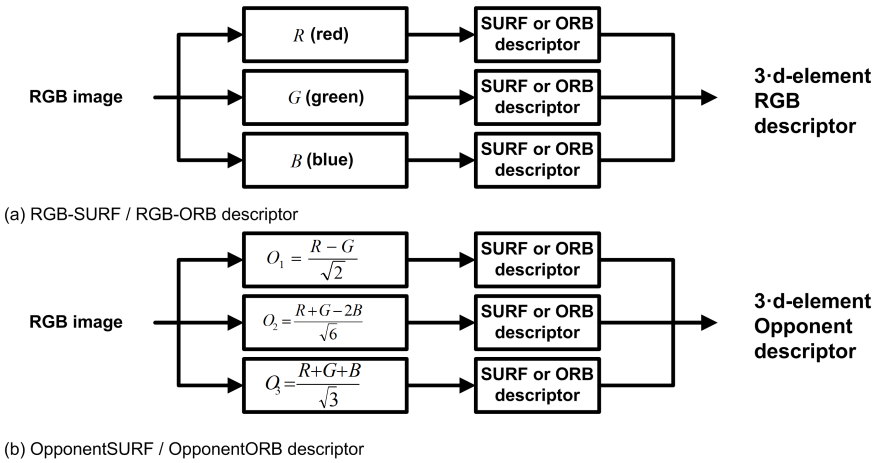

Fig. 2. Block diagram of color SURF/ORB descriptor extraction, where d denotes the dimension of the original local descriptor.

that the original SURF/ORB descriptors use. The other two channels (O_1 and O_2) capture the color information, where O_1 is the red-green component and O_2 is the blue-yellow component. Following the transformation, a normalization step that converts the ranges of each channel within the [0,255] range is employed, as in [24]. Similarly with RGB-SURF/ORB, the original SURF or ORB descriptors are then applied separately to each transformed channel and the final $3 \cdot d$-element feature vectors are the concatenation of the three feature vectors extracted from the three channels.

5 Reducing the Dimensionality of Local Color Descriptors

State-of-the-art encoding methods generate high-dimensional vectors that make difficult the training of machine learning algorithms. For example, while the

Algorithm 2 Algorithm for channel-PCA

Input: The number of color channels c ($c = 3$ in our color descriptors); the dimension d of each channel of the color descriptor (normally 128 or 256); the desired dimension l' of the reduced feature vector (the full feature vector will be reduced from $l = c \cdot d$ to l'); the complete feature matrix A that will be used for learning the projection matrices
Projection Matrix calculation: Calculate c projection matrices of size $d \times p_i$ according to:
for $i = 1$ to c **do**
1. Perform eigenvalue decomposition of the covariance matrix corresponding to the features of the current channel (i.e., corresponding to a part of the features in A)
2. Select the number of principal components p_i to retain for this channel ($\sum_{i=1}^{c} p_i = l'$)
3. Form the channel's projection matrix using only the first p_i principal components
end for
Dimensionality reduction using channel-PCA: Given a new feature vector x', transform the features of each color channel using the corresponding projection matrix, and concatenate the transformed feature vectors for all channels

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

BoW model generates a k-element feature vector, where k equals to the number of visual words, VLAD encoding generates a $k \cdot l$-element feature vector (where l is the dimension of the local descriptor; in the case of the color extensions of descriptors discussed in the previous section, $l = 3 \cdot d$). Thus, it is common to employ dimensionality reduction before the construction of VLAD vectors, on local descriptors, mainly using PCA [28]. In this section we explain that directly applying PCA to the full vector of color descriptors, as implied from previously published works (e.g. [7]; termed "typical-PCA" in the sequel), is not the only possible solution, and we propose a simple modification of this descriptor dimensionality reduction process that it experimentally shown to improve the concept detection results in several cases.

PCA projects linearly l-dimensional features to a lower-dimensional feature space. Given a matrix A with dimension $l \times n$, where n is the number of observations, if we want to perform dimensionality reduction (from l to l') with PCA, the reduced matrix A' will be $A' = E^T \cdot A$, where E is the projection matrix (of dimension $l \times l'$) and T denotes the transpose of a matrix.

PCA aims to find those directions in the data space that present high variance. When PCA is applied directly to the entire vector of one of the color extensions of (binary or non-binary) local descriptors, if one or two of the three color channels of the descriptor exhibit lower diversity than the others, then these risk being under-represented in the reduced dimensionality space. To avoid this, we propose performing PCA separately for each color channel and consider an equal number of principal components from each of them, to create three projection matrices that correspond to each of the three channels, instead of one projection matrix that corresponds to the complete descriptor vector. The three reduced single-channel descriptor vectors that can be obtained for a color descriptor using the aforementioned projection matrices are finally concatenated in a reduced color-descriptor vector. Algorithm 2 summarizes the proposed channel-PCA algorithm.

6 Experiments

6.1 Experimental Setup

Our experiments were performed on the TRECVID 2013 Semantic Indexing (SIN) dataset [19], which consists of a development set and a test set (approximately 800 and 200 hours of internet archive videos for training and testing, respectively). We evaluate our system on the test set using the 38 concepts that were evaluated as part of the TRECVID 2013 SIN Task, and we follow the TRECVID methodology for the evaluation of the results [19].

For experimenting with all methods, one keyframe was initially extracted for each video shot and was scaled to 320×240 pixels prior to feature extraction. For some of our final experiments, we also extracted two visual tomographs [26] from each shot. Regarding feature extraction, we followed the experimental setup of [7] and we used the toolbox that its authors have published. More specifically, we

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

used the dense SIFT descriptor, that accelerates the original SIFT descriptor, in combination with the Pyramid Histogram Of visual Words (PHOW) approach [4]. PHOW is a simple modification of dense SIFT that uses more than one square regions at different scale levels in order to extract features. For SURF and ORB we used their implementations included in OpenCV, and further extended these implementations with the corresponding color variants that we introduced in Section 4. The same square regions at different scale levels of the PHOW approach were used as the image patches that were described by ORB and SURF. We calculated 128-SIFT, 128-SURF and 256-ORB grayscale descriptors; then, each color extension of a descriptor resulted in a color descriptor vector three times larger than that of the corresponding original descriptor, as explained in Section 4. All the local descriptors were compacted (to 80 dimensions for SIFT, SURF and their color extensions, following the recommendations of [7] and [14]; to 80 dimensions for grayscale ORB and to 256 dimensions for ORB color extensions) using PCA and were subsequently aggregated using the VLAD encoding. Similarly with the authors of [7], we divided each image into the same 8 regions using spatial binning and we used sum pooling to combine the encodings from different regions. As a result of the above process, a VLAD vector of 163840 elements for SIFT, SURF or grayscale ORB and of 524288 elements for ORB color extensions was extracted for each image (by image we mean here either a keyframe or a visual tomograph). These VLAD vectors were compressed into 4000-element vectors by applying a modification of the random projection matrix [3]. These reduced VLAD vectors served as input to the Logistic Regression (LR) classifiers that we used. Following the *cross validated committees* methodology of [17], we trained five LR classifiers per concept and per local descriptor (SIFT, ORB, RGB-ORB etc.), and combined the output of these five by means of late fusion (averaging). When different descriptors were combined, again late fusion was performed by averaging of the classifier output scores. In all cases, the final step of concept detection was to refine the calculated detection scores by employing the re-ranking method proposed in [23].

6.2 Results and Discussion

Tables 1, 2 and 3 present the results of our experiments in terms of Mean Extended Inferred Average Precision (MXinfAP) [29], which is an approximation of the Mean Average Precision (MAP) suitable for the partial ground truth that accompanies the TRECVID dataset [19].

Table 1. Performance (MXinfAP, %) for ORB, when the binary codebook proposed in [12] and when a floating-point codebook is used. In parenthesis we show the relative improvement w.r.t. the binary codebook.

Descriptor	Binary codebook [12]	Floating-point codebook (no PCA)	Floating-point codebook (PCA 80)
ORB	4.52	10.36 (+129.2%)	11.43 (+152.9%)

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

In Table 1 we examine the performance of the original grayscale ORB descriptor in concept detection, when used in conjunction with a binary codebook (as in [12]) and a floating-point one (as in Section 3). In both cases, VLAD encoding is employed. We can see that the binary codebook proves ineffective; the floating-point one outperforms it by more than 129%. We also compacted the ORB descriptors to 80 dimensions using PCA before encoding them, which further increased MXinfAP to 11.43%. Based on this result, in all subsequent experiments with ORB and its extensions a floating-point codebook was used. In addition, grayscale ORB was compacted to 80 dimensions.

In Table 2 we evaluate the different local descriptors and their color extensions considered in this work, as well as combinations of them. First, comparing the original ORB descriptor with the other two non-binary descriptors (SIFT, SURF), we can see that ORB performs rather similarly to its non-binary counterparts (more precisely, its MXinfAP is a bit worse). This performance is achieved despite ORB and its extensions being much more compact than SIFT and SURF, as seen in the second column of Table 2. Second, concerning the methodology for introducing color information to local descriptors, we can see that the com-

Table 2. Performance (MXinfAP, %) for the different descriptors, when typical and channel-PCA for dimensionality reduction is used. In parenthesis we show the relative improvement w.r.t. the corresponding original grayscale local descriptor for each of the SIFT, SURF and ORB color variants.

Descriptor	Descriptor size in bits	Keyframes, typical-PCA	Keyframes, channel-PCA	Boost(%) w.r.t typical-PCA
SIFT	1024	14.22	14.22	-
RGB-SIFT	3072	14.97 (+5.3%)	14.5 (+2.0%)	-3.1%
OpponentSIFT	3072	14.23 (+0.1%)	14.34 (+0.8%)	+0.8%
SIFT combination	-	**19.11 (+34.4%)**	**19.24 (+35.3%)**	+0.7%
SURF	1024	14.68	14.68	-
RGB-SURF	3072	15.71 (+7.0%)	15.99 (+8.9%)	+1.8%
OpponentSURF	3072	14.7 (+0.1%)	15.26 (+4.0%)	+3.8%
SURF combination	-	**19.4 (+32.2%)**	**19.48 (+32.7%)**	+0.4%
ORB	256	11.43	11.43	-
RGB-ORB	768	13.02 (+13.9%)	13.58 (+18.8%)	+4.3%
OpponentORB	768	12.61 (+10.3%)	12.73 (+11.4%)	+1.0%
ORB combination	-	**17.38 (+52.1%)**	**17.45 (+52.7%)**	+0.4%
SIFT/SURF combination	-	**22.4**	**22.35**	-0.2%
SIFT/ORB combination	-	**21.32**	**21.46**	+0.7%
SURF/ORB combination	-	**21.56**	**21.74**	+0.8%
SIFT/SURF/ORB combination	-	**23.00**	**23.01**	0.0%

Table 3. Performance (MXinfAP, %) for different combinations of descriptors, (a) when features are extracted only from keyframes, (b) when horizontal and vertical tomographs described by SIFT, RGB-SIFT and OpponentSIFT are also examined, (c) when the Label Powerset algorithm is also applied [18].

Descriptor	(a) Keyframes (channel-PCA)	(b) Keyframes+ Tomographs	(c) Keyframes+ Tomographs+LP
SIFT combination	19.24	20.28	21.35
SURF combination	19.48	19.74	20.92
ORB combination	17.45	17.83	19.92
SIFT/SURF/ORB combination	**23.01**	**24.49**	**25.58**

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

bination of the original SIFT descriptor and the two known color SIFT variants that we examine ("SIFT combination" in Table 2) outperforms using the original SIFT descriptor alone by 34.4% (35.3% for channel-PCA). The similar combinations of the SURF color variants with the original SURF descriptor, and of the color variants of ORB with the original ORB descriptor, are shown in Table 2 to outperform the original SURF and ORB by 32.2% and 52.1%, respectively (which increase to 32.7% and 52.7% for channel-PCA). These results show that the relatively straightforward way we used for introducing color information to SURF and ORB, based on the similar SIFT extensions, is in fact generally applicable to heterogeneous local descriptors.

To analyse the influence of PCA on the vectors of local color descriptors, we also compared in Table 2 the channel-PCA of section 5 with the typical approach of applying PCA directly on the entire color descriptor vector. In both cases PCA was applied before the VLAD encoding, and in applying channel-PCA we kept the same number of principal components from each color channel (e.g. for RGB-SIFT, which is reduced to $l' = 80$ using typical-PCA, we set $p_1 = p_2 = 27$ for the first two channels and $p_3 = 26$ for the third color channel; $p_1 + p_2 + p_3 = l'$). According to the relative improvement figures reported in the last column of Table 2, performing the proposed channel-PCA in several cases improves the concept detection results, compared to the typical-PCA alternative, without introducing any additional computational overhead.

Another observation from Table 2 is that the concept detection performance increases when pairs of local descriptors (including their color extensions) are combined (i.e., SIFT/SURF, SIFT/ORB and SURF/ORB combinations), which shows a complementarity in the information that the different local descriptors capture. The best overall results among the experiments of Table 2 are achieved when all the local descriptors and their color variants are combined (last row of this table), reaching a MXinfAP of 23.01%.

Finally, in Table 3 we report experiments with two literature techniques that can further benefit the combination of SIFT, SURF and ORB. Specifically, we experiment with video tomographs [26] (for simplicity these are described using only SIFT and its two color extensions) and a two-layer stacking architecture that captures concept correlations using the Label Powerset (LP) algorithm in the second layer [18]. In all experiments of this table, for the color variants of SIFT, SURF and ORB, channel-PCA was used. The results of Table 3 indicate that introducing some form of motion information (through tomographs) and considering the correlations among concepts (through LP) can give an additional 11.2% relative improvement to the best results reported in Table 2 (MXinfAP increased from 23.01% to 25.58%).

7 Conclusions

In this work we showed that a binary local descriptor (ORB) can perform sufficiently well, compared to its non-binary counterparts, in the video concept detection task. We also showed that a methodology previously used for defining

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

two color variants of SIFT is a generic one that is also applicable to descriptors such as ORB and SURF. We proposed a different way of employing PCA for dimensionality reduction of color descriptors that are used in combination with VLAD (channel-PCA). Finally, we quantified the impact of combining the above techniques (e.g. combination of binary and non-binary color descriptors) and other previously proposed methods (tomographs, LP) to a concept detection system.

Acknowledgements This work was supported by the European Commission under contracts FP7-287911 LinkedTV and FP7-600826 ForgetIT.

References

1. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: Fast retina keypoint. In: IEEE Int. Conf. CVPR 2012. pp. 510–517 (2012)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision and Image Understing 110(3), 346–359 (2008)
3. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: Applications to image and text data. In: 7th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. pp. 245–250. ACM, NY (2001)
4. Bosch, A., Zisserman, A., Muoz, X.: Image classification using random forests and ferns. In: IEEE Int. Conf. ICCV 2007. pp. 1–8. Rio de Janeiro (2007)
5. Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., Fua, P.: BRIEF: Computing a Local Binary Descriptor Very Fast. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(7), 1281–1298 (2012)
6. Canclini, A., Cesana, M., Redondi, A., Tagliasacchi, M., Ascenso, J., Cilla, R.: Evaluation of low-complexity visual feature detectors and descriptors. In: 18th Int. Conf. on Digital Signal Processing (DSP), 2013. pp. 1–7 (2013)
7. Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: British Machine Vision Conference. pp. 76.1–76.12. British Machine Vision Association (2011)
8. Chen, D.M., Makar, M., de Araújo, A.F., Girod, B.: Interframe coding of global image signatures for mobile augmented reality. In: DCC. pp. 33–42 (2014)
9. Chu, D., Smeulders, A.: Color invariant surf in discriminative object tracking. In: Kutulakos, K. (ed.) Trends and Topics in Computer Vision, LNCS, vol. 6554, pp. 62–75. Springer (2012)
10. Fan, P., Men, A., Chen, M., Yang, B.: Color-SURF: A surf descriptor with local kernel color histograms. In: IEEE Int. Conf. on Network Infrastructure and Digital Content. pp. 726–730 (2009)
11. Fu, J., Jing, X., Sun, S., Lu, Y., Wang, Y.: C-surf: Colored speeded up robust features. In: Yuan, Y., Wu, X., Lu, Y. (eds.) Trustworthy Computing and Services, Communications in Computer and Information Science, vol. 320, pp. 203–210. Springer (2013)
12. Grana, C., Borghesani, D., Manfredi, M., Cucchiara, R.: A fast approach for integrating orb descriptors in the bag of words model. In: SPIE. vol. 8667, pp. 866709–866709–8 (2013)
13. Jegou, H., Douze, M., Schmid, C., Perez, P.: Aggregating local descriptors into a compact image representation. In: IEEE on Computer Vision and Pattern Recognition (CVRP 2010). pp. 3304–3311. San Francisco, CA (2010)

Proc. 21st Int. Conf. on MultiMedia Modeling (MMM'15), Sydney, Australia, Springer LNCS vol. 8935, pp. 282-293, Jan. 2015.
The final publication is available at link.springer.com

14. Jegou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(9), 1704–1716 (2012)
15. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: IEEE Int. Conf. ICCV 2011. pp. 2548–2555 (2011)
16. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. Int. Journal of Computer Vision 60(2), 91–110 (2004)
17. Markatopoulou, F., Moumtzidou, A., Tzelepis, C., Avgerinakis, K., Gkalelis, N., Vrochidis, S., Mezaris, V., Kompatsiaris, I.: ITI-CERTH participation to TRECVID 2013. In: TRECVID 2013 Workshop. Gaithersburg, MD, USA (2013)
18. Markatopoulou, F., Mezaris, V., Kompatsiaris, I.: A comparative study on the use of multi-label classification techniques for concept-based video indexing and annotation. In: Gurrin, C., Hopfgartner, F., Hurst, W., Johansen, H., Lee, H., OConnor, N. (eds.) MultiMedia Modeling. LNCS, vol. 8325, pp. 1–12. Springer (2014)
19. Over, P., Awad, G., Michel, M., Fiscus, J., Sanders, G., Kraaij, W., Smeaton, A.F., Quenot, G.: Trecvid 2013 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In: Proceedings of TRECVID 2013. NIST, USA (2013)
20. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: 11th Eur. Conf. on Computer Vision: Part IV. pp. 143–156. Springer-Verlag (2010)
21. Qiu, G.: Indexing chromatic and achromatic patterns for content-based colour image retrieval. Pattern Recognition 35, 1675–1686 (2002)
22. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. In: IEEE Int. Conf. on Computer Vision. pp. 2564–2571 (2011)
23. Safadi, B., Quénot, G.: Re-ranking by local re-scoring for video indexing and retrieval. In: 20th ACM Int. Conf. on Information and Knowledge Management. pp. 2081–2084. ACM, NY (2011)
24. Van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. IEEE transactions on pattern analysis and machine intelligence 32(9), 1582–1596 (2010)
25. Van de Sande, K.E.A., Snoek, C.G.M., Smeulders, A.W.M.: Fisher and vlad with flair. In: IEEE Conference on Computer Vision and Pattern Recognition (2014)
26. Sidiropoulos, P., Mezaris, V., Kompatsiaris, I.: Video tomographs and a base detector selection strategy for improving large-scale video concept detection. IEEE Transactions on Circuits and Systems for Video Technology 24(7), 1251–1264 (2014)
27. Snoek, C.G.M., Worring, M.: Concept-Based Video Retrieval. Foundations and Trends in Information Retrieval 2(4), 215–322 (2009)
28. Witten, I., Frank, E.: Data Mining Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco, second edn. (2005)
29. Yilmaz, E., Kanoulas, E., Aslam, J.A.: A simple and efficient sampling method for estimating ap and ndcg. In: 31st ACM SIGIR Int. Conf. on Research and Development in Information Retrieval. pp. 603–610. ACM, USA (2008)
30. Zhou, X., Yu, K., Zhang, T., Huang, T.S.: Image classification using super-vector coding of local image descriptors. In: 11th European Conf. on Computer Vision: Part V. pp. 141–154. ECCV 2010, Springer-Verlag (2010)