# Advances in Visual Concept Detection: Ten Years of TRECVID

Ville Viitaniemi, Mats Sjöberg, Markus Koskela,
Satoru Ishikawa and Jorma Laaksonen

[Id: erkki2014.tex,v 1.37 2014/10/15 23:35:57 jorma Exp ]

**Abstract**

In this article we describe the structure and operation of the visual concept detection subsystem of the PicSOM multimedia retrieval system. We evaluate several alternative techniques used for implementing this component and show the essential results of a series of experiments in the large-scale setups of the TRECVID video retrieval evaluation campaigns in 2005, 2009 and 2014. During these years, the PicSOM system has gone through substantial evolution in both the statistical features and the detection algorithms employed. Transition from global image features to the bag-of-visual-words features and recently further to convolutional deep neural network -based features is also justified in the light of our results. Overall, during the ten years of participation in TRECVID, the PicSOM system has show close to state-of-the performance in this very rapidly developing field of research.

## 1 Introduction

Content-based multimedia information retrieval addresses the problem of finding data relevant to the users' information needs from multimedia databases. In early content-based image and video retrieval systems, the retrieval was usually based solely on querying by examples and measuring the similarity of the database objects (images, video shots) with *low-level features* automatically extracted from the objects. Generic low-level features are often, however, insufficient to discriminate content well on a conceptual level. This "semantic gap" is the fundamental problem in content-based multimedia retrieval.

In recent years, it has become common to build semantic representations of multimedia content by applying machine learning techniques for detecting *mid-level semantic concepts* (events, objects, locations, people, etc.) on basis of the content's low-level visual and aural features [42, 30, 56]. This kind of mid-level representation at least narrows the semantic gap. In recent studies it has been observed that, despite the far-from-perfect accuracy of concept detectors, the representation often is very useful in supporting *high-level indexing and querying* on multimedia data [20]. This is mainly because semantic concept detectors can be trained off-line with computationally more demanding supervised learning algorithms and with considerably more positive and negative training examples than what are typically available at query time. The automatic machine learning based approach is scalable to large numbers of multimedia objects and features. The introduction of large-scale multimedia ontologies, such as LSCOM [41] and ImageNet [13] and large manually annotated data sets (e.g. [2]) have enabled generic analysis of multimedia content as well as an increase in multimedia lexicon sizes by orders of magnitude.

Through years of experimentation and evaluation of concept detection techniques by the multimedia retrieval community, an understanding has emerged that machine learning systems for concept detection should generally be based on fusion of several low-level features extracted from the multimedia content, not just a single well-performing feature. Accepting this boundary condition of feature fusion, there still remain many design choices in implementing a concept detection system. Typically such systems are complex and consist of several sub-modules. The modules themselves can be implemented using a multitude of alternative technologies, and there are alternative ways to combine the modules together.

Given the complex nature of concept detection systems, it is not self-evident which factors and techniques are beneficial for concept detection performance. Some of the techniques applied in systems exhibiting good overall concept detection performance might be essential, whereas some other, attractive-looking techniques might just be parts of otherwise well-functioning systems, without being particularly effective themselves. This situation calls for controlled experiments where just one component of a concept detection system is varied while other system parts are kept constant.

In this article, we describe the development of the concept detection subsystem in our PicSOM multimedia analysis and retrieval framework. We discuss several alternative ways of implementing its components. As one highlight, we propose and study a set of post-processing techniques for taking advantage of correlations that seman-

tic concepts occurring in video material typically exhibit, both in temporal dimension between shots and across different concepts. In particular, the proposed post-processing techniques combine an $N$-gram intra-concept inter-shot temporal modeling technique with a simple clustering approach that takes advantage of temporal and instantaneous inter-concept co-occurrences. Most of the current state-of-the-art multimedia retrieval systems do not include inter-shot temporal analysis.

In the experiments of this paper we extensively compare the concept detection performances the use of the component technologies leads to, employing the large-scale experimental settings of the high-level feature extraction (HLFE) and semantic indexing (SIN) tasks of the annual TRECVID video retrieval evaluation campaign. The name of the task was changed in 2010, but the content of the task remained very much the same, even though also the type and amount of video material employed was changed simultaneously. The TRECVID settings have arguably represented the multimedia research community's best effort to realistically model large-scale multimedia search tasks in a controlled benchmark setting. Yearly dozens of research groups evaluate their techniques and systems using this benchmark.

The succeeding sections of this article are organized as follows. In Section 2 we describe the parts of a generic video retrieval system in order to provide context for the concept detection subsystem, which is the main topic of this paper. We also describe some implementation details of those parts in our PicSOM multimedia retrieval framework to the degree in which they are relevant from the concept-detection point of view. Section 3 contains the essential theoretical and methodological contribution of this paper. There we describe the concept detection techniques implemented in the PicSOM system. Section 4 presents empirical verifications of the proposed concept detection algorithms in the TRECVID evaluations of years 2005, 2009 and 2014. In Section 5 we give our final conclusions from our experiments and experiences.

# 2 Parts of a video retrieval system

Figure 1 schematically shows the architecture of the automatic concept detection and search subsystems in our PicSOM multimedia retrieval system. The implementation of the concept detection system, seen in the center of the figure, is the focus of this paper. In the search phase, the outputs of the concept detection system can be supplemented with outputs of the interactive content-based information retrieval (CBIR) and textual search modules also depicted in the illustration.

[Figure 1 about here.]

The operation of a video search system generally consists of two phases. In the first phase, the system is *prepared* for a video corpus. The corpus is divided into an annotated training part and an unannotated testing part, on which video retrieval is going to be performed in the second *search* phase.

In the preparing phase the whole video corpus is first segmented into shots and the annotations are associated with the shots. A number of low-level visual, audio and textual feature descriptors are extracted from each shot and content-based indices prepared based on the features. In systems that rely on automatic detection of concepts, the annotated part can then be used to train shot-wise detectors for the concepts that have been specified in the annotations. The detectors apply supervised learning techniques to form a mapping between low-level shot features and the annotation concepts, earlier often referred as *high-level features*, and more recently as *visual semantic concepts*. The preparing phase is allowed to be time-consuming as it is intended to be performed off-line prior to the actual on-line use of the retrieval system.

After the preparation phase, the retrieval system is ready to be used for video retrieval in the search phase. In this phase, the system is queried with a textual phrase, combined with image and video examples of the desired query topic. The result of a query is a list of video shots, ranked in the order of decreasing predicted likelihood to match the query. The system operation in the search phase is intended to be sufficiently fast to enable the retrieval needs of a real user to be satisfied while the user is waiting, typically in a couple of seconds. The example images and video shots will require pre-processing, feature extraction and classification that cannot be performed during the preparing phase, but will inevitably need to be done while the user is waiting for the output.

As description and evaluation of concept detection techniques forms the essence of this paper, the detailed discussion of those techniques are postponed to Section 3. In the remainder of this section we discuss the other parts of a video retrieval system. The preparation phase parts are described to the extent in which they are relevant for the subsequent concept detection experiments. The description of the video search phase in turn motivates and emphasizes the need for well-functioning semantic concept detectors.

## 2.1 Shot segmentation and keyframe selection

The first task of the preparing phase for a comprehensive video retrieval system is to segment the video corpus temporally into sequential basic units. The PicSOM multimedia retrieval system implements two shot boundary detection techniques, based on global visual feature evolution [40] and interest point tracking statistics [33]. However, for the experiments of this paper with TRECVID video material, we employ the openly available master definition of shots [46] so that our results are comparable with those by other groups that have performed TRECVID tasks.

Another preparation phase task is the extraction of one or more keyframes from each video shot. The keyframes are needed both for extracting visual features to describe the content of the shot and for presenting them to the users of the system as still replacements for the dynamic video content. The most straightforward keyframe selection method is to use the center-most frame of each shot. Better results can be obtained by selecting the keyframe on the basis of the content of the shot, by comparing the frames with their neighbors and the calculated average of the shot [49]. In recent years, the organizers of the TRECVID semantic indexing task have provided also all i-frames of the MPEG-4 compressed video streams, and it has been computationally feasible to use all of them as keyframes.

## 2.2 Low-level features

Automatic extraction of low-level features is the foundation of large-scale content-based multimedia processing. Using pixel values of video or image data directly in search and retrieval is typically neither sensible nor feasible. Effective features combined with an appropriate distance or similarity measure facilitates the use of the statistical vector space model approach, which is the basis of most current multimedia analysis methods. In many cases a single well-chosen keyframe can compactly express the most central visual characteristics of that shot. Consequently, one can use still-image features, often originally developed for image-only retrieval systems, as a way to compare video shots.

In the following sections we briefly go through different modalities of features that can be extracted to represent different relevant and complementary aspects of the underlying video data. Feature types that are used in the concept detection experiments of this paper are described in more detail as the nature and quality of the extracted features critically determine the maximum level of performance that a concept detection system based on those features can achieve.

### 2.2.1 Global image features

Many of the classical image features are global, i.e. calculated from all pixels of the image, thus representing characteristics of the image as a whole. An increasingly popular alternative has been to calculate features separately for smaller image segments, for example calculating each block in a grid or pyramid structure placed over the image. It is also possible to use automatic segmentation, where the image is split into visually homogeneous segments, for which features are calculated separately [4].

The PicSOM system uses a wide range of image features. In TRECVID 2005, many of PicSOM's global image features were based on the standardized MPEG-7 descriptors [39]. We used both the implementations of the MPEG-7 XM reference software and our own more efficient implementations of the following MPEG-7 features: *Color Layout*, *Color Structure*, *Dominant Color*, *Scalable Color*, *Edge Histogram*, and *Region Shape*. Furthermore, PicSOM implements some non-standard image features developed in-house: *Average Color*, *Color Moments*, *Texture Neighborhood*, *Edge Histogram*, *Edge Co-occurrence* [7] and *Edge Fourier*. These have been calculated either globally or for five spatial zones (center, top, bottom, left, right) of the image. In the case of zoning, the final image-wise feature vector has been obtained as a concatenation of the zone-wise features.

### 2.2.2 BoV image features

Until very recently, the field of image analysis has been dominated by the the approach of characterizing images by describing the statistics of their local feature descriptors. The local descriptors can be calculated for visually salient *interest points* [1]. For instance, the points can be edge or corner points where the the image content changes substantially. Another strategy is to sample image area evenly and calculate local descriptors for the sample of image locations. Histograms of robust, scale-invariant local descriptors—such as Scale-Invariant Feature Transform (SIFT) [37] and the Speeded Up Robust Features (SURF) [5]—and later their Fisher Vector encodings provided the state-of-the-art image descriptors between 2008 and 2013.

Histograms of localized features are also called *bag of visual words* (BoV) in analogy to the traditional bag-of-words approach in textual information retrieval. In this interpretation each histogram bin—representing a specific local pattern—is seen as a "visual word" in the vocabulary of all the histogram bins. The BoV features can be enhanced by calculating the histograms for different subdivisions of the image, in addition to the entire image [35]. Another recent improve-

ment to the BoV methodology is to use soft-assignment in histogram generation as demonstrated e.g. in [59].

In addition to the BoV encoding, other approaches include sparse coding of the local descriptors [66], supervector encoding [71], vector of locally aggregated descriptors (VLAD) [26], and the Fisher vector [45]. The Fisher vector encoding can arguably be considered as the current state-of-the-art in local feature based image classification. By measuring the deviation of a sample from a GMM-based generative model in the SIFT descriptor space, one ends up, however, with very high-dimensional image signatures.

The BoV features used in the PicSOM system in TRECVID evaluation of year 2009 were based on the *SIFT* local descriptors and the opponent color space version of the *Color SIFT* descriptor [58]. We have employed two different strategies for selecting the points from which the local descriptors are extracted: the Harris-Laplace interest point detector and dense sampling of images. The codebooks have been generated with k-means and Self-Organizing Map (SOM) clustering algorithms.

In 2014, the PicSOM system used also densely-sampled SIFT descriptors encoded with VLAD and Fisher vectors. The codebooks were generated using k-means with 512 cluster centers and a 128-component GMM, respectively.

### 2.2.3 Deep convolutional network features

A recent major development in image classification has been the use of deep convolutional neural networks (CNNs), with excellent results [34, 70, 19]. The convolutional networks based on the structure of Krizhevsky et al [34] typically contain five or more convolutional layers, followed by two fully-connected layers, and the output layer. Still, one drawback with CNNs is that they require huge amounts of training data and delicate tuning of the training parameters. It has, however, been observed that CNNs trained with one visual dataset can function as highly discriminative features even for considerably different data domains and tasks [15, 31]. We can therefore employ CNNs trained with external data as feature extractors in a standard concept detection framework.

In 2014 the PicSOM system included a total of 24 CNN features extracted with four different CNN networks [31]. We use the activations of the first fully-connected layers of each network as our features, which results in 4096-dimensional feature vectors. We also use the reverse spatial pyramid pooling proposed in [19] with two scale levels. The first level corresponds to the full image, and the second level consists of nine regions with scale of two. The CNN activations of the

regions are then pooled using average pooling, and the activations of the different scales are concatenated. The resulting spatial pyramid features are therefore 8192-dimensional.

### 2.2.4 Video features

In many cases the static visual properties of a video keyframe are not enough to describe the salient features of the full scene. The dynamic properties may also make the computational learning problem easier. It has been reported in various recent publications that using video features beyond the single keyframe approach can improve the results [50, 55, 22]. For the experiments reported in this paper for the TRECVID 2005 and 2009 evaluations, we extracted video features by temporally extending some of the still-image features described in the previous sections. When calculating these features, the video shot is first divided into non-overlapping temporal subshots of equal lengths. A feature vector is calculated separately for each frame and all the frame feature vectors averaged within the subshots to form feature vectors which are finally concatenated to form one shot-wise feature vector.

### 2.2.5 Audio features

Most video shots include a sound track, containing for example human speech, music or different environment sounds. The general level characteristics of the sound track can be described either globally or the track can be segmented into separately described parts. A popular approach for the description is to calculate the mel-scaled cepstral coefficients (MFCC) [11]. Besides coarse general level description of audio, speech can often be automatically recognized and thus handled as text as will be described in the following section. Depending on the video analysis and retrieval task at hand, analyzing music and environment sounds may or may not be beneficial. For the experiments of this paper audio features were used only in the TRECVID 2005 evaluation.

### 2.2.6 Textual features

Video material often includes textual data or meta-data that can facilitate text-based indexing and retrieval. Textual data for video shots may originate e.g. from speech recognition, closed captions, subtitles, or video OCR. As text-based information retrieval methodology is very mature and text indices can provide fast and accurate results [47, 3], an effective video retrieval system will generally benefit from a text

search component when responding to the high semantic level queries from the user. For detecting mid-level semantic concepts, however, textual features are not always useful. The textual information in the TRECVID corpora used before year 2010 was obtained through an automatic speech recognition and machine translation process. Experiences had proven that textual features extracted from that material performed poorly in detecting visual concepts.

In the experiments of this paper, textual features were used in the TRECVID evaluation of year 2005, but not in the evaluation of year 2009. Since year 2010, textual data have not been provided in the TRECVID semantic indexing task.

## 2.3 Search phase

The ultimate goal of video retrieval is to find relevant video content for a specific information need of the user. The conventional approach has been to rely on textual descriptions, keywords, and other meta-data to achieve this functionality, but this requires manual annotation and does not usually scale well to large and dynamic video collections. In some applications, such as YouTube, the text-based approach works reasonably well, but it fails when there is no meta-data available or when the meta-data cannot adequately capture the essential content of the video material.

Content-based video retrieval, on the other hand, utilizes techniques from related research fields, such as image and audio processing, computer vision, and machine learning, to automatically index the video material. Content-based queries are typically based on a small number of provided examples (i.e. *query-by-example*). The material of a video collection is ranked based on its similarity to the examples according to low-level features [54, 14, 53]. In recent works, the content-based techniques are commonly combined with separately pre-trained detectors for various semantic concepts (*query-by-concepts*) [20, 56]. It has been empirically observed that visual concept lexicons or ontologies are an integral part of effective content-based video retrieval systems.

Concept-based video retrieval can operate in either automatic or interactive mode. In *automatic concept-based video retrieval*, no user interaction is needed after a query has been presented to the retrieval system. In the automatic mode, the fundamental challenge is mapping the user's information need into the space of available concepts in the used concept ontology [43]. The basic approach is to select a small number of concept detectors as active and weight them based either on the performance of the detectors or their estimated suitability for

the current query. Negative or complementary concepts are not typically used. In [43], the methods for automatic selection of concepts were divided into three categories: *text-based*, *visual-example-based*, and *results-based methods*. Text-based methods use lexical analysis of the textual query and resources such as WordNet [17] to map query words into concepts. Methods based on visual examples measure the similarity between the provided example objects and the concept detectors to identify suitable concepts. Results-based methods perform an initial retrieval step and analyze the results to determine the concepts that are then incorporated into the actual retrieval algorithm.

In addition to automatic retrieval, *interactive concept-based retrieval* constitutes a parallel paradigm. Interactive video retrieval systems include the user in the loop at all stages of the retrieval session and therefore call for sophisticated and flexible user interfaces. A global database visualization tool providing an overview of the database as well as a localized point-of-interest with increased level of detail are typically needed. Relevance feedback can also be used to steer the interactive query toward video material the user considers relevant [29]. Semantic concept detection has generally been recognized as an important component also in interactive video retrieval [20], and current state-of-the-art interactive video retrieval systems (e.g. [12]) typically use concept detectors as a starting point for the interactive search functionality.

# 3 Concept detection in PicSOM

After having extracted low-level video features from each shot, supervised learning techniques can be applied in order to learn the associations between the low-level features and the concepts in the annotations of the video corpus. The PicSOM multimedia retrieval system includes a supervised concept detection subsystem trained in the preparing phase of the video corpus. Figure 2 illustrates the overall architecture of this system. All the $K$ concepts are first detected from each shot, based on the shot's low-level features, $K$ being the number of concepts that have been annotated in the training part of the video corpus. This step results in a $K$-dimensional vector of detection scores. After the shot-level concept detection, the scores are re-adjusted in a post-processing step according to the score vectors of temporally neighboring shots, based on the estimated likelihood of observing particular temporal concept patterns.

## 3.1 Shot-level concept detection

The shot-level concept detection task is in the PicSOM system addressed with a well-established fusion-based architecture. The fusion-based approach is common also in other well-performing state-of-the-art image and video analysis systems (e.g. [55, 44]). In our approach, dozens of supervised probabilistic detectors are first trained for each concept, based on the different shot-wise low-level features, detailed in Section 2.2, and their early-fusion combinations. The feature-wise detector outcomes are then fused in a post-classifier fusion (also called late fusion) step. The outlined shot-level detection architecture contains a number of components that can be implemented in several alternative ways. In the following we describe the techniques implemented in the PicSOM system during the various stages of its development.

[Figure 2 about here.]

Given the extracted shot-wise features, the first stage in our fusion algorithm is the feature-wise supervised detection of concepts. Each concept and feature is treated symmetrically, i.e. every concept is detected with the same algorithms.

### 3.1.1 Self-Organizing Maps

Historically, the Self-Organizing Map (SOM)-based detectors have always been part of the PicSOM throughout the system's existence. The early emphasis was on interactive CBIR, where the rapidness of the SOM approach in learning new category definitions is essential for a satisfactory user experience. Also much of the early work where the PicSOM system has been used in off-line category detection tasks used SOM-based detectors. One of the first evaluations of this approach took place in the TRECVID 2005 evaluation as will be described in Section 4.

The construction of the SOM-based detectors begins with quantizing the feature spaces using the TS-SOM [28] algorithm, a tree-structured variant of the SOM [27]. In the subsequent learning algorithm, the bottom levels of TS-SOMs define the quantization and the upper levels act as an index structure for rapid search. Typically, TS-SOMs from two to four stacked levels have been used, the bottom levels measuring from $16 \times 16$ to $256 \times 256$ map units, respectively. Figure 3 shows an example of a TS-SOM quantization of a feature space based on the color and texture distribution of image segments.

[Figure 3 about here.]

The TS-SOM preparation step needs to be performed only once for each feature type in an image collection. After that, generating a classifier for any binary partitioning of the training images is very fast. Any partitioning is characterized by the division of the training images into positive and negative examples. The classifier for the partitioning is created by subtracting the proportion of negative examples that fall into each bottom-level TS-SOM unit, i.e. quantization bin, from the corresponding proportion of positive examples. This way a classification score is assigned to each quantization bin. After this initial scoring, the scores are low-pass filtered on the two-dimensional TS-SOM grid surface, taking advantage of the topology-preserving characteristic of the SOM clustering and efficiently emphasizing the differences between the feature space regions where positive and negative examples are well separated, or occur mixed with each other.

When the preparation step is complete, a detection score is associated with each quantization bin of the feature space. Assigning a feature-wise detection score to an independent test image is then simple: the extracted feature vector of the image is quantized using the same quantization scheme and the image receives the detection score of the quantization bin into which its feature vector is mapped.

### 3.1.2 Non-linear Support Vector Machines

After the Self-Organizing Map, non-linear Support Vector Machine (SVM) [10] algorithm was used as the supervised detection algorithm in PicSOM. The SVM implementation used is an adaptation of the C-SVC classifier of the LIBSVM software library [9].

We have used the radial basis function (RBF) SVM kernel

$$g_{\mathrm{RBF}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2\right) \tag{1}$$

for all the shot-wise features and also have the option to use the $\chi^2$ kernel

$$g_{\chi^2}(\mathbf{x}, \mathbf{x}') = \exp\left(-\gamma \sum_{i=1}^{d} \frac{(x_i - x_i')^2}{x_i + x_i'}\right) \tag{2}$$

for $d$-dimensional histogram-like visual features.

The free parameters of the SVMs are selected with an approximate 10-fold cross-validation search procedure that consists of a heuristic line search to identify a promising parameter region, followed by a grid search in that region. To speed up the computation, the data set is radically downsampled for the parameter search phase. Further speed-up is gained by optimizing the C-SVC cost function only very approximately during the search. For the final detectors we also downsample the data set, but less radically than in the parameter

search phase. Usually there are much fewer annotated example shots of a concept (positive examples) than there are example shots not exhibiting that concept (negative examples). In these cases we usually retain all the positive examples and just limit the number of negative examples.

### 3.1.3 Linear Support Vector Machines

There have been numerous approaches to reduce the computational complexity from the level of standard non-linear SVMs. Such approaches include using approximate SVM solvers [6, 65], reducing the number of support vectors [8, 16], and replacing the non-linear SVMs with linear classifiers [69]. It is also possible to speed up SVMs by using GPUs [57]. Using linear classifiers is particularly appealing, as both the training and classification time requirements can be several orders of magnitude smaller than with non-linear SVMs. Recent algorithms for training large-scale linear classifiers include the stochastic sub-gradient descent in Pegasos [48] and the dual coordinate descent algorithm in LIBLINEAR [21]. As a practical example, in our current implementation and the TRECVID data used in experiments reported in this paper, evaluating a linear classifier for a single image (excluding feature extraction) takes only a fraction of a millisecond whereas non-linear SVMs require 100–200 ms per image. In PicSOM we have focused on two approaches, homogeneous kernel maps, and power mean SVM.

Non-linear kernel classifiers can be considered as linear classifiers in a feature space for which there exists a corresponding implicit feature map $\Psi : \mathbb{R}^d \to \mathbb{R}^D$. Therefore, one approach is to perform an explicit (either exact or approximate) feature mapping to convert the non-linear problem into a linear one and use a standard linear solver. With an exact feature map this is straightforward:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Psi(\mathbf{x}_i), \Psi(\mathbf{x}_j) \rangle \ . \tag{3}$$

The exact mapping approach can work in certain cases, but, in general, the dimensionality $D$ of the feature map $\Psi$ can be high or even infinite, as is the case e.g. with the RBF kernel. Therefore, a more practical approach is to approximate the non-linear kernel. One approach is to try to find a mapping function $\hat{\Psi} : \mathbb{R}^d \to \mathbb{R}^r$ so that

$$\langle \hat{\Psi}(\mathbf{x}_i), \hat{\Psi}(\mathbf{x}_j) \rangle \approx K(\mathbf{x}_i, \mathbf{x}_j) \ . \tag{4}$$

In the general case, finding such mappings is difficult, but it has turned out that with additive kernels this is possible. A kernel is

*additive* if it can be represented as a sum of feature-component-wise one-dimensional functions, i.e. if it can be written as

$$K(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{d} k_i(x_i, z_i) \ , \tag{5}$$

where $\mathbf{x} = [x_1, \ldots, x_d]^T, \mathbf{z} = [z_1, \ldots, z_d]^T \in \mathbb{R}_+^d$. Common additive kernels include the intersection kernel

$$k_{\text{int}}(x_i, z_i) = \min(x_i, z_i) \ , \tag{6}$$

the $\chi^2$ kernel

$$k_{\chi^2}(x_i, z_i) = -\frac{(x_i - z_i)^2}{x_i + z_i} \ , \tag{7}$$

the Bhattacharyya kernel

$$k_{\text{bha}}(x_i, z_i) = \sqrt{x_i z_i} \ , \tag{8}$$

and the Jensen-Shannon kernel

$$k_{\text{js}}(x_i, z_i) = \frac{x_i}{2} \log_2 \frac{(x_i + z_i)}{x_i} + \frac{z_i}{2} \log_2 \frac{(x_i + z_i)}{z_i} \ . \tag{9}$$

In [38], Maji et al proposed a sparse feature map for the intersection kernel, and subsequently Vedaldi and Zisserman proposed *homogeneous kernel maps* [60, 61] for any additive homogeneous kernel. Such explicit kernel maps are convenient to use as they do not require any changes to the linear classification algorithm and are data independent. As a result, no learning is required and the kernel map can be computed on-the-fly using a look-up table.

The homogeneous kernel map of order $n$ is a $(2n+1)$-dimensional linear approximation of an additive kernel for a scalar feature, $\hat{\Psi}_n : \mathbb{R} \to \mathbb{R}^{2n+1}$. Due to the additivity property (Eq. (5)), one can then encode a $d$-dimensional feature vector as a $d(2n+1)$-dimensional linear problem using the kernel map and use any standard linear solver with it to approximate the corresponding non-linear kernel. The complexity of evaluating the classifier is thus $O(d)$. In [60, 61], homogeneous kernel maps are provided for many common additive kernels used in computer vision. Among them, an implementation of the homogeneous kernel map of order $n = 2$ has been adopted in the PicSOM system for the experiments described in this paper.

## 3.2 Fusion algorithms

The PicSOM system includes several alternative algorithms for the fusion of feature-wise concept detectors. As a baseline approach we

form the geometric mean of all the detector outcomes for each processed video frame. Besides this unsupervised fusion approach, we also implement several supervised fusion methods that make use of the cross-validated detector outcomes for the training set.

One supervised technique is SVM-based fusion employing RBF kernels, another Bayesian Binary Regression (BBR) [18]. The other implemented alternatives are variations of the scheme where the basic fusion mechanism is still the geometric mean, but the mean is calculated only over a subset of the detector outcomes, selected by a sequential forward-backward search (SFBS).

In addition to basic SFBS, we implement the idea of partitioning the training set into multiple folds. In our implementation we have used a fixed number of six folds. The SFBS algorithm is run several times, each time leaving one fold outside the training set. The final fusion outcome is the geometric mean of the fold-wise geometric means. For later reference, we denote this fusion algorithm multifold-SFBS.

We also consider reserving a part of the training set for validation and early-stopping the search based on the performance in this validation set. This early-stopping can be combined with both the basic SFBS and multifold-SFBS algorithms. For the basic SFBS, one sixth of the training data is used as a validation set. In the case of multifold-SFBS, the left-out fold for each fold-wise run is re-used as the validation set.

In addition to the fusion fusion mechanisms used to combine the outputs of multiple detectors for a single video frame, one needs to fuse the frame-wise results to shot-wise detection scores, provided that there are more than one keyframe in a shot. In PicSOM, we have employed for this purpose a simple maximum pooling approach which we have found to perform better than e.g. arithmetic and geometric average pooling techniques.

## 3.3 Temporal post-processing

For temporal post-processing of the fusion outcomes, the PicSOM system implements techniques first described in [64]. The techniques operate on a stream of $K$-tuples corresponding the concept detector outputs for the sequential video shots, where $K$ is the number of the detected concepts. The methods thus ignore the absolute timing and duration of the video shots, preserving only their ordering.

Methodologically, our temporal post-processing is based on $N$-gram modeling performed for each concept individually. In the following, $c_n \in \{0, 1\}$ is an indicator variable of the occurrence of the concept to be detected at time instant $n$ and $s_n \in \mathbf{R}$ is the output of

the corresponding concept detector. $H_n$ denotes the recursive prediction history known at time instant $n$, extending $N-1$ steps backwards in time:

$$H_n = \{\hat{p}(c_{n-i}|s_{n-i}, H_{n-i})\}_{i=1}^{N-1} \ . \tag{10}$$

Using this notation, we can write the recursive $N$-gram model as

$$\hat{p}\left(c_n|s_n, H_n\right) \propto \hat{p}(s_n|c_n)\hat{p}\left(c_n|H_n\right) \tag{11}$$

if we assume the conditional independence of $s_n$ and $H_n$ given $c_n$, i.e.

$$\hat{p}(s_n|c_n, H_n) = \hat{p}(s_n|c_n) \ . \tag{12}$$

Then the recursive model can be written as

$$\hat{p}\left(c_n|H_n\right) \ = \ \sum_{c_{n-1}} \cdots \sum_{c_{n-N+1}} p_0(c_n|c_{n-1}, \ldots, c_{n-N+1}) \prod_{i=1}^{N-1} \hat{p}\left(c_{n-i}|s_{n-i}, H_{n-i}\right) \tag{13}$$

Here $p_0$ is the marginalized $N$-gram probability that is estimated from the training data. The $N$-gram model is initialized in the beginning of each video by using models of lower order, e.g. a bigram model is used on the second time instant. The conditional distributions of detector outputs $\hat{p}(s_n|c_n)$ are modeled as exponential distributions

$$\hat{p}(s_n|c_n = i) = \frac{1}{\lambda_i} e^{-s_n/\lambda_i}, \quad i \in \{0, 1\} \ . \tag{14}$$

For concept-wise parameters $\lambda$ we use the maximum likelihood estimates

$$\hat{\lambda_i} = \frac{\sum_{n|c_n=i} s_n}{\sum_{n|c_n=i} 1}, \quad i \in \{0, 1\} \ , \tag{15}$$

where the summation is over the shots of the training set.

In addition to this causal model, we also form the corresponding anticausal model that is obtained by reversing the time flow. The causal and anticausal models are then combined by logarithmic averaging of the model outcomes.

# 4 Experiments

In this section, we describe the experiments we have performed in the high-level feature extraction and semantic indexing tasks of the TRECVID evaluation campaigns in 2005, 2009 and 2014, and present an analysis of the results. The experiments are based on our submissions to corresponding TRECVID evaluations [32, 51, 23], but for this paper we have complemented the submitted results with additional experiments based on retrospective analysis of the annual results.

## 4.1 TRECVID evaluation campaign

TRECVID [52] is an annual workshop series organized by the National Institute of Standards and Technology (NIST) and arguably the leading venue for evaluating research on content-based video analysis and retrieval. It started in 2001 as TREC workshop's Video Track and since 2003 it has been organized as a workshop of its own. TRECVID provides the participating organizations large test collections, uniform scoring procedures, and a forum for comparing the results. Each year the TRECVID evaluation contains a set of video analysis tasks, such as high-level feature extraction or semantic indexing, video search, video summarization, event detection, and content-based copy detection.

In the experiments of this paper, we focus on the high-level feature extraction (HLFE) task of TRECVID 2005 and 2009, and the semantic indexing (SIN) task of TRECVID 2014. As already stated, these tasks are basically the same from the point of view of visual analysis, only the name was changed in 2010.

The video material used in TRECVID has consisted of television news broadcasts (until 2006), documentaries, news reports, and educational programs (2007–2009), and consumer videos from the Internet Archive[1] (since 2010). The video material is divided into shots in advance and these reference shots are used as the unit of concept detection [46]. To obtain training data for the HLFE/SIN task, an annual collaborative annotation effort has been organized [2].

Due to the size of the test corpora, it has been infeasible within the resources of the TRECVID initiative to perform an exhaustive examination in order to determine the topic-wise ground truth. Therefore, a pooling technique has been used instead. First, a pool of possibly relevant shots is obtained by gathering the sets of shots returned by the participating teams. These sets are then merged, duplicate shots are removed, and the relevance of only this subset of shots is assessed manually.

The main performance measure in the HLFE/SIN task has been first the *inferred average precision* (infAP) [67] and later the *extended inferred average precision* (xinfAP) [68], which approximate the standard *average precision* very closely, but require only a subset of the pooled results to be evaluated manually. The mean of the concept-wise precision values over a set of queries, *mean (extended) inferred average precision* (MIAP and MXIAP), is then used to provide an overview of the results.

[1] www.archive.org

## 4.2 Experiments 2005

In our TRECVID experiments in 2005 [32], we were using the Self-Organizing Map as the shot-level concept detection method. The main emphasis of our experiments was on evaluating the performance of the features, both visual and textual, for each concept. This was feasible, because only ten concepts were used. The visual features that were available that time were mostly global features, but some histogram-based texture features were also already available.

### 4.2.1 Data

In 2005, the TRECVID high-level feature extraction task data consisted of 170 hours of video data recorded from TV news broadcasts in Lebanon, China and the United States. Automatic speech recognition (ASR) and machine translation (MT) results in English were provided by the organizers. One half of the data was given for the development of the systems and the other half was reserved for testing.

### 4.2.2 Video features

On the video shot level, we used the MPEG-7 [25] *Motion Activity* descriptor (MA) and temporal versions of three still image features. The temporal image features were calculated by dividing each video first into five non-overlapping parts with equal lengths. All the frames of these five subshots were then extracted, and each frame divided spatially into five separate zones: the upper, the lower, the left hand side, the right hand side and the central zone. A feature vectors were calculated separately for each zone, and were then concatenated to form a vector depicting the whole frame. All the frame-wise feature vectors of a subshot were then averaged and these average vectors were concatenated to form the feature vector of the video clip. Several different video features were calculated using this method by varying the feature that is calculated for the zones of the frames. *Average Color* (AC), *Color Moments* (CM) and *Texture Neighborhood* (TN) features were the three zone features that were used.

The Average Color feature vector is a three element vector that contains the average RGB values of all the pixels within the zone. The Color Moments feature is calculated by separating the HSV color channels from the zone. Then the values of the color channels are treated as probability distributions, and the first three moments (mean, variance and skewness) are calculated for each distribution. The feature vector contains the three moment values for the three color channels.

The Texture Neighborhood feature is calculated from the Y (luminance) component of the YIQ color representation of the zone pixels. The 8-neighborhood of each inner pixel is examined, and a probability estimate is calculated for the probabilities that the neighbor pixel in each surrounding relative position is brighter than the central pixel. The feature vector contains these eight probability estimates.

#### 4.2.3 Image features

For the keyframe indices we used a set of six standard MPEG-7 [25] descriptors, viz. *Color Layout* (CL), *Color Structure* (CS), *Dominant Color* (DC), *Scalable Color* (SC), *Edge Histogram* (EH), and *Homogeneous Texture* (HT). The descriptors were extracted globally from every keyframe in the collection, i.e. no segmentation or zoning was used.

#### 4.2.4 Audio features

The mel-scaled cepstral coefficient, or shortly *Mel Cepstrum* (CE) is the discrete cosine transform (DCT) applied to the logarithm of the mel-scaled filter bank energies. The number of coefficients taken is 12, and these are organized as vector. Finally the total power of the signal is appended to the vector giving a feature vector of length 13.

#### 4.2.5 Text features

Unlike the other features, an inverted file instead of a SOM index was used for the ASR/MT output. For the high-level feature extraction task, the text features were constructed by gathering concept-dependent lists of most informative terms. Let us denote the number of shots in the development set associated with concept $c$ as $N_c$ and assume that of these shots, $n_{c,t}$ contain the term $t$ in the ASR/MT output. After pre-processing and stemming, the following measure is applied for term $t$ regarding the concept $c$:

$$S_c(t) = \frac{n_{c,t}}{N_c} - \frac{n_{all,t}}{N_{all}} \ . \tag{16}$$

For every concept, we recorded the 10 and 100 most informative terms and use them as alternative text features.

#### 4.2.6 Feature selection

The set of used features was selected for each concept separately. For this purpose, we applied a SFS-type feature selection scheme, in which we began with an empty set of features and computed a criterion value

for each of the potential features. If adding the feature with the highest value improved the overall result, the feature was used in the task and the process continued. Otherwise the selection process was stopped. As the optimization criterion we used the average precision at 2000 returned items with two-fold cross validation on the development set.

### 4.2.7 Results

Table 1 lists the sets of selected features for each of the ten studied concepts. As can be seen, the selection process typically resulted in 4–7 features to be selected and fused. The *Prisoner* concept was a notable exception as adding any second feature, including the text features, beside Homogeneous Texture resulted in performance degradation. In general, video features seem to be included in the feature set more often than still image features. Among the image features, Edge Histogram and Homogeneous Texture were used more than the color-based features. Audio and text-features were beneficial only for a subset of the concepts.

[Table 1 about here.]

Figure 4 shows the PicSOM Team's placement among the submissions evaluated by NIST. As this was our first participation in the evaluation, we able to submit only one result. Our performance can be regarded as mediocre. In 2005, most of the other participants were already using Support Vector Machines in their systems and it was therefore decided that also the PicSOM system should start to use them as the principal classifier technology for concept detection.

[Figure 4 about here.]

## 4.3 Experiments 2009

In 2009, we had followed the example given by other successful groups in the TRECVID evaluations and started to use bag-of-visual-words (BoV) features and non-linear Support Vector Machines (SVMs) in the PicSOM system. The aim of our experiments in TRECVID 2009 was to evaluate the advantage that could be obtained with the SIFT and Color SIFT BoV features compared to the global features used earlier. Other research questions were the benefits that could results from late fusion of the detector outputs and from applying temporal post-processing to the shot-wise detection results.

### 4.3.1 Data

The video data for the system development in TRECVID 2009 consisted of approximately 100 hours of documentaries, news reports, and educational programs from the Dutch TV. 280 hours of similar video data were used for evaluation. Table 2 lists all the concepts detected in TRECVID 2009.

[Table 2 about here.]

In the experiments reported in the following subsections the shot-wise feature sets that we have used as a starting point consist solely of various combinations of visual features, i.e. keyframe image and video features. Audio and text features have not been used.

### 4.3.2 Shot-wise features

As a preparation for the post-classifier fusion, we trained a number of individual SVM detectors, each based on a single shot-level feature. This lets us compare different shot-level features in terms of their detection accuracies, although the individual detectors are only used as components of the final fusion-based detection subsystem.

The best individual feature performances we observed resulted from histograms of local image features collected according to the bag-of-visual-words (BoV) paradigm, i.e. variants of SIFT and Color SIFT features. Table 3 compares different BoV feature variants in terms of MIAP [67]. As expected, Color SIFT outperforms normal SIFT. Dense sampling is a more effective approach than interest point detection. The soft histogram technique and spatial pyramids improve the performance of the BoV features as well. These results hold on average, but concept-wise differences are large. It does not seem likely that all the differences would result from statistical fluctuations. Table 4 lists the most accurate non-BoV features, which can be seen clearly inferior in performance to the BoV features.

[Table 3 about here.]

[Table 4 about here.]

Combining the features with with early fusion did produce more accurate detectors than the individual image features alone, the best early fusion combination having MIAP 0.0601. In the whole concept detection system, the detector results based on single features and their early fusion combinations are further processed using late fusion. In some of our previous investigations, also the overall system

performance has benefited from early fusion [63]. However, in the experiments with the 2009 data, the use of early fusion did not improve the overall system performance when also late fusion stage was included.

### 4.3.3 Fusion algorithms

We performed a preliminary evaluation of the various post-classifier fusion algorithms in a setting where the annotated part of the video corpus was further partitioned to a training and validation part in 2:1 proportions. In this preliminary experiment SVM and BBR based fusion algorithms were significantly and consistently outperformed by geometric mean based fusion algorithms, both by the unsupervised basic version and by the supervised SFBS variants. Moreover, the SVM and BBR fusion mechanisms are computationally much more costly. Consequently, the remaining evaluation with the full data set was constrained to the variants of geometric mean fusion.

Figure 5(a) compares different geometric mean based fusion algorithms with the whole video corpus and four different sets D1–D4 of detectors to be fused. These sets result from different sets of shot-wise features, different SVM training parameters and different cross-concept strategies. The number of fused detectors ranges between 77 (D1) and 26 (D4). We can see that the geometric mean of all detectors (the leftmost bar) is always inferior to methods where the set of detectors is selected with sequential forward-backward search (SFBS). This has not always been the case in our earlier experiments as SFBS easily overfits to the training data. The figure also shows that multifold-SFBS performs better than the basic SFBS. Early stopping has no essential effect on the average performance. It, however, seems to increase the variance of the results. These experiments thus confirm that early stopping is not a suitable way of regularizing SFBS.

The results of this section—when compared with the MIAP values of the best individual features in Section 4.3.2—can be used to confirm the observation that fusion of features usually outperforms individual features, even if the best individual features are clearly better than the worst-performing fused features. With a good fusion algorithm, benefit can be obtained from individually rather badly-performing features. In one experiment we picked approximately 75% of the best features for fusion, thus leaving just the worst performing 25% of the features outside. Still, with the multifold-SFBS fusion algorithm the fusion accuracy improved when the worst 25% were returned to the feature set. With a less-developed fusion algorithm, the saturation point is reached earlier where further addition of features no longer improves the fusion result. An example of this behavior can be seen in Fig-

ure 5(a) when comparing sets of detectors D3 and D4. Here set D3 is a superset of D4 having almost three times as many detectors. When the geometric mean fusion is used, better performance is obtained by using the smaller set D4, whereas with the more advanced SFBS fusion algorithms the situation reverses: benefit can be obtained from the extra detectors in D3.

[Figure 5 about here.]

#### 4.3.4 Temporal post-processing

Figure 5(b) shows the effect of temporal post-processing for a selection of shot-wise fusion-based detectors F1–F4. The detectors employ different sets of shot-wise features and fusion algorithms. From the figure we can observe that the $N$-gram post-processing (bars with diagonal hatching) improves MIAP markedly over the baseline with no post-processing (white bars). We evaluated two strategies for choosing the order of $N$-gram models. In one strategy, the $N$-gram order was selected for each concept separately based on a validation experiment performed with 2:1 split of the training data. The other strategy was to choose the order globally, i.e. select the order of $N$-grams that resulted in the best mean performance over all the concepts in the validation experiment. As the results show, the global order-selection approach works somewhat better. In almost all the cases the global selection resulted in the selection of order eight, the maximum order that was considered. Generally, the mean performance seems to increase rapidly with increasing $N$-gram order at first. Gradually the performance starts to saturate and eventually begins to degrade slowly when the order is further increased.

The post-processing methods marked with identifier "any" (solid dark bars) refer to the concept-wise selection of the post-processing method from a larger pool of methods according to the best performance in the 2:1 validation experiment. In addition to the $N$-gram methods, this pool included clustering-based techniques that take advantage of temporal and instantaneous inter-concept correlations. Those techniques turned out to be useful in an experiment with data sets of TRECVID 2005–2007 [64], although in that case the baseline detectors were based on less powerful SOM detectors instead of non-linear SVMs. As can be observed from the figure, in these TRECVID 2009 concept detection experiments, the inter-concept methods did not bring any improvement over $N$-grams.

### 4.3.5 The best 2009 PicSOM system and its performance

In the above sections we have investigated many alternative techniques for implementing semantic concept detection from videos. In this section we collect the results together and describe the best-performing concept-detection module of a video retrieval system that we can assemble from the discussed components. We compare the performance of such a module with that of the state-of-the-art systems that participated in the TRECVID high-level feature extraction tasks in year 2009.

Our experiments have shown that with our fusion algorithms, the PicSOM system can benefit from all the shot-wise visual features we have extracted. For concept detection, we thus train one non-linear SVM detector based on each shot-wise feature. In SVM training, we have to make a compromise between accuracy and training time. The detectors from all the features are fused together with the multifold-SFBS post-classifier fusion algorithm. The concept detection is finalized with an $N$-gram temporal post-processing stage where we use the same $N$-gram order (eight) for all the concepts.

Figure 6 shows the MIAP concept detection performance of the PicSOM system in the TRECVID HLFE tasks of year 2009 in comparison with the best-performing systems of that year. The baseline system PicSOM B closely resembles our official submission in the evaluation. The PicSOM A result has been obtained after the official evaluation by using somewhat more comprehensive set of low-level visual features and more elaborate SVM training. It can be seen that the TRECVID 2009 performance has been further improved, and while not being absolutely the best, PicSOM's HLFE performance compared well with the state-of-the-art systems of that time.

[Figure 6 about here.]

## 4.4 Experiments 2014

In the experiments of TRECVID 2014, the used PicSOM system was again based on late fusion of a large variety of supervised detectors trained for each concept. We augmented the set of used features with CNN activation features (see Section 2.2.3) and dense SIFT descriptors encoded with Fisher vectors and VLAD encoding (Section 2.2.2). As classifiers for the CNN features, we utilized linear SVMs with homogeneous kernel maps [60] of order $d = 2$ to approximate the intersection kernel. For Fisher vectors and VLAD, the classifiers were

trained using linear SVMs due to the high dimensionality of the vectors and consequent computational complexity.

### 4.4.1 Data

In 2014, the TRECVID semantic indexing task used Internet Archive video data that consisted of 800 hours for development and 200 hours for evaluation. The development set contained 28123 videos with average 1 min 40 s length whereas there were 2373 videos with average 5 min length in the test set. In the training material keyframes were extracted and used from each shot, and in the test material the i-frames provided by NIST were utilized. Submissions were requested for 60 semantic concepts.

### 4.4.2 Hard negative mining in detector training

A concept-wise, two-class classifier generally produces false positives on negative examples that are similar to the positive examples according to the used feature space. Therefore, to acquire more relevant negative examples, we performed $n$ rounds of hard negative mining [36]. The final classifier for a given feature was obtained by fusing the classifier trained with the original, randomly sampled negatives and the $n$ classifiers using mined relevant negatives. We observed in preliminary experiments that a single round of mining hard negatives already brought the greatest improvement. We therefore used the value $n = 1$ in the following experiments.

[Table 5 about here.]

### 4.4.3 Submitted runs and results

Table 5 shows an overview of our submitted runs, where the four columns in the middle refer to the used features: global features, BoV features, Fisher vectors + VLAD, and CNN features. The next column indicates whether hard negative mining was used, and the rightmost column lists the corresponding mean extended inferred average precision (MXIAP) [68] values.

Run 1 is intended to match our best submission in TRECVID 2013, i.e. to use the same features, classifiers, and method of fusion [24]. In Run 2, the Fisher vector and VLAD features and the set of 24 CNN features were included and the global image features discarded. Run 3 uses only the CNN features, together with hard negative mining, and Run 4 combines the characteristics of Runs 2 and 3, that is, all SIFT-based and CNN features with hard negative mining.

The most striking observation on the results is the notable increase of performance compared to our last year's submissions. This is mostly due to the extended set of features, in particular the CNN activation features. By comparing Runs 1 and 2, we observe a 40% increase on MXIAP induced by the different feature sets.

Second, the mining of hard negatives further improved the results, as can be observed by comparing Runs 2 and 4, the latter including the mining step and obtaining the highest MXIAP among our runs, 0.2880 (a 6% increase). The solid performance of the CNN features can furthermore be observed from Run 3, which contains only the CNN features but still almost reaches the MXIAP value of Run 4.

[Figure 7 about here.]

Figure 7 shows all runs submitted to the TRECVID 2014 semantic indexing task, our runs highlighted. In total, there were 75 submissions, and only the MediaMill group of the University of Amsterdam submitted runs that were superior to the two best PicSOM runs in their MXIAP results.

# 5 Conclusions

In this paper we have described the concept detection architecture of our PicSOM multimedia retrieval system and proposed and evaluated several alternative techniques for implementing its components. The presented experiments started with TRECVID 2005, where we used the Self-Organizing Maps as the detector algorithm for mostly global image and video features.

In TRECVID 2009, the non-linear Support Vector Machines had replaced SOMs as the detectors, and SIFT and Color SIFT based BoV features were shown to be superior in performance compared to the global descriptors. In that study, we also demonstrated the usefulness of feature selection and evolved late fusion, as well as that of temporal post-processing of shot-wise detection results. Using the proposed techniques, the performance of the PicSOM concept detection subsystem compares favorably with other state-of-the-art systems of that time.

With our recent experiments in TRECVID 2014, we have shown that the top performance obtained in many image classification tasks with deep convolutional neural networks can be carried over to semantic video indexing tasks. For the reasons of computational complexity, we used linear SVM detectors with homogeneous kernel maps to approximate the intersection kernel. Combined with the hard negative

mining technique in detector training, the PicSOM group ranked second among 75 submission to the semantic indexing task.

As a whole, this paper has shown an overview of the gradual evolution of the PicSOM multimedia retrieval system since our first participation in TRECVID's visual concept detection evaluations in 2005. This evolution has concerned the used features, which have developed from global to BoV-based and further to CNN-based, the applied detector algorithms, which have changed from the Self-Organizing Map to non-linear and linear Support Vector machines, and also various fusion and post-processing techniques. As a general trend, the PicSOM team's performance and ranking in the evaluation results has been steadily improving — being now the second in this highly competitive and appreciated evaluation.

# References


[1] Amsaleg, L. and Gros, P. 2001. Content-based retrieval using local descriptors: Problems and issues from a database perspective. *Pattern Analysis & Applications 4,* 2+3 (June), 108–124.

[2] Ayache, S. and Quénot, G. 2008. Video corpus annotation using active learning. In *Proceedings of 30th European Conference on Information Retrieval (ECIR'08)*. Glasgow, UK, 187–198.

[3] Baeza-Yates, R. and Ribeiro-Neto, B. 1999. *Modern Information Retrieval*. Addison-Wesley.

[4] Barnard, K. and Shirahatti, N. V. 2003. A method for comparing content based image retrieval methods. In *Proceedings of SPIE Internet Imaging IV*. Vol. 5018. Santa Clara, CA, USA, 1–8.

[5] Bay, H., Tuytelaars, T., and Gool, L. V. 2006. SURF: Speeded up robust features. In *Proc. ECCV 2006*.

[6] Bordes, A., Ertekin, S., Weston, J., and Bottou, L. 2005. Fast kernel classifiers with online and active learning. *Journal of Machine Learning Research 6*, 1579–1619.

[7] Brandt, S., Laaksonen, J., and Oja, E. 2002. Statistical shape features for content-based image retrieval. *Journal of Mathematical Imaging and Vision 17,* 2 (September), 187–198.

[8] Burges, C. J. C. and Schölkopf, B. 1997. Improving the accuracy and speed of support vector learning machines. In *Proc. NIPS*.

[9] Chang, C.-C. and Lin, C.-J. 2001. *LIBSVM: a library for support vector machines.* Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[10] Cortes, C. and Vapnik, V. 1995. Support-vector networks. *Machine Learning 20,* 3, 273–297.

[11] Davis, S. B. and Mermelstein, P. 1990. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. In *Readings in speech recognition*, A. Waibel and K. Lee, Eds. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 65–74.

[12] de Rooij, O., Snoek, C. G. M., and Worring, M. 2008. Balancing thread based navigation for targeted video search. In *Proceedings of the International Conference on Image and Video Retrieval (CIVR 2008)*. Niagara Falls, Canada, 485–494.

[13] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*.

[14] Dimitrova, N., Zhang, H.-J., Shahraray, B., Sezan, I., Huang, T., and Zakhor, A. 2002. Applications of video-content analysis and retrieval. *IEEE MultiMedia 9,* 3 (July-September), 42–55.

[15] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. 2014. DeCAF: A deep convolutional activation feature for generic visual recognition. In *ICML 2014*.

[16] Downs, T., Gates, K. E., and Masters, A. 2002. Exact simplification of support vector solutions. *Journal of Machine Learning Research 2*, 293–297.

[17] Fellbaum, C., Ed. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press.

[18] Genkin, A., Lewis, D. D., and Madigan, D. 2005. BBR: Bayesian logistic regression software. Software available at http://www.stat.rutgers.edu/ madigan/BBR/.

[19] Gong, Y., Wang, L., Guo, R., and Lazebnik, S. 2014. Multi-scale orderless pooling of deep convolutional activation features. arXiv.org:1403.1840.

[20] Hauptmann, A. G., Christel, M. G., and Yan, R. 2008. Video retrieval based on semantic concepts. *Proceedings of the IEEE 96,* 4 (April), 602–622.

[21] Hsieh, C.-J., Chang, K.-W., Lin, C.-J., Keerthi, S. S., and Sundararajan, S. 2008. A dual coordinate descent method for large-scale linear SVM. In *Proceedings of 25th International Conference on Machine Learning (ICML 2008)*. Helsinki, Finland.

[22] Inoue, N., Hao, S., Saito, T., Shinoda, K., Kim, I., and Lee, C. 2009. TITGT at TRECVID 2009 workshop. In *Proceedings of the TRECVID 2009 Workshop*. Gaithersburg, MD, USA.

[23] Ishikawa, S., Koskela, M., Sjöberg, M., Anwer, R., Laaksonen, J., and Oja, E. 2014. PicSOM experiments in TRECVID 2014. In *Proceedings of the TRECVID 2014 Workshop*. Gaithersburg, MD, USA.

[24] Ishikawa, S., Koskela, M., Sjöberg, M., Laaksonen, J., Oja, E., Amid, E., Palomäki, K., Mesaros, A., and Kurimo, M. 2013. PicSOM experiments in TRECVID 2013. In *Proceedings of the TRECVID 2013 Workshop*. Gaithersburg, MD, USA.

[25] ISO/IEC. 2002. Information technology - Multimedia content description interface - Part 3: Visual. 15938-3:2002(E).

[26] Jegou, H., Douze, M., Schmid, C., and Perez, P. 2010. Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*.

[27] Kohonen, T. 2001. *Self-Organizing Maps*, Third ed. Springer Series in Information Sciences, vol. 30. Springer-Verlag, Berlin.

[28] Koikkalainen, P. and Oja, E. 1990. Self-organizing hierarchical feature maps. In *Proceedings of International Joint Conference on Neural Networks*. Vol. II. San Diego, CA, USA, 279–284.

[29] Koskela, M. 2003. Interactive image retrieval using self-organizing maps. Ph.D. thesis, Laboratory of Computer and Information Science, Helsinki University of Technology. Available online at: http://lib.hut.fi/Diss/2003/isbn9512267659/.

[30] Koskela, M. and Laaksonen, J. 2006. Semantic concept detection from news videos with self-organizing maps. In *Proceedings of 3rd IFIP Conference on Artificial Intelligence Applications and*

*Innovations*, I. Maglogiannis, K. Karpouzis, and M. Bramer, Eds. IFIP, Springer, Athens, Greece, 591–599.

[31] Koskela, M. and Laaksonen, J. 2014. Convolutional network features for scene recognition. In *Proceedings of the 22nd International Conference on Multimedia*. Orlando, Florida.

[32] Koskela, M., Laaksonen, J., Sjöberg, M., and Muurinen, H. 2005. PicSOM experiments in TRECVID 2005. In *Proceedings of the TRECVID 2005 Workshop*. Gaithersburg, MD, USA, 262–270. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[33] Koskela, M., Sjöberg, M., Viitaniemi, V., and Laaksonen, J. 2008. PicSOM experiments in TRECVID 2008. In *Proceedings of the TRECVID 2008 Workshop*. Gaithersburg, MD, USA. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[34] Krizhevsky, A., Sutskever, I., and Hinton, G. 2012. ImageNet classification with deep convolutional neural networks. In *NIPS*.

[35] Lazebnik, S., Schmid, C., and Ponce, J. 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 2. 2169–2178.

[36] Li, X., Snoek, C. G. M., Worring, M., Koelma, D. C., and Smeulders, A. W. M. 2013. Bootstrapping visual categorization with relevant negatives. *IEEE Transactions on Multimedia 15,* 4 (June), 933–945.

[37] Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision 60,* 2 (November), 91–110.

[38] Maji, S., Berg, A., and Malik, J. 2008. Classification using intersection kernel support vector machines is efficient. In *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2008)*. 1–8.

[39] Manjunath, B. S., Salembier, P., and Sikora, T., Eds. 2002. *Introduction to MPEG-7: Multimedia Content Description Interface*. John Wiley & Sons Ltd.

[40] Muurinen, H. and Laaksonen, J. 2007. Video segmentation and shot boundary detection using self-organizing maps. In *Proceedings of 15th Scandinavian Conference on Image Analysis (SCIA 2007)*. Aalborg, Denmark, 770–779.

[41] Naphade, M., Smith, J. R., Tešić, J., Chang, S.-F., Hsu, W., Kennedy, L., Hauptmann, A., and Curtis, J. 2006. Large-scale concept ontology for multimedia. *IEEE MultiMedia 13,* 3, 86–91.

[42] Naphade, M. R. and Huang, T. S. 2002. Extracting semantics from audiovisual content: The final frontier in multimedia retrieval. *IEEE Transactions on Neural Networks 13,* 4 (July), 793–810.

[43] Natsev, A. P., Haubold, A., Tešić, J., Xie, L., and Yan, R. 2007. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *Proceedings of ACM Multimedia (ACM MM'07)*. Augsburg, Germany, 991–1000.

[44] Ngo, C.-W., Jiang, Y.-G., Wei, X.-Y., Zhao, W., Liu, Y., Wang, J., Zhu, S., and Chang, S.-F. 2009. VIREO/DVMM at TRECVID 2009: High-level feature extraction, automatic video search, and content-based copy detection. In *Proceedings of the TRECVID Workshop*. 415–432.

[45] Perronnin, F. and Dance, C. 2007. Fisher kernels on visual vocabularies for image categorization. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*. 1 –8.

[46] Petersohn, C. 2004. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *TREC Video Retrieval Evaluation Online Proceedings*. TRECVID.

[47] Salton, G. and McGill, M. J. 1983. *Introduction to Modern Information Retrieval*. Computer Science Series. McGraw-Hill, New York.

[48] Shalev-Shwartz, S., Singer, Y., and Srebro, N. 2007. Pegasos: Primal estimated sub-gradient solver for SVM. In *Proceedings of the 24th international conference on Machine learning*. ICML '07. ACM, New York, NY, USA, 807–814.

[49] Sjöberg, M., Koskela, M., Chechev, M., and Laaksonen, J. 2010. PicSOM experiments in TRECVID 2010. In *Proceedings of the TRECVID 2010 Workshop*. Gaithersburg, MD, USA. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[50] Sjöberg, M., Muurinen, H., Laaksonen, J., and Koskela, M. 2006. PicSOM experiments in TRECVID 2006. In *Proc. of the TRECVID 2006 Workshop*. Gaithersburg, MD, USA.

[51] Sjöberg, M., Viitaniemi, V., Koskela, M., and Laaksonen, J. 2009. PicSOM experiments in TRECVID 2009. In *Proceedings of the TRECVID 2009 Workshop*. Gaithersburg, MD, USA. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[52] Smeaton, A. F., Over, P., and Kraaij, W. 2006. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*. ACM Press, New York, NY, USA, 321–330.

[53] Smeaton, A. F., Wilkins, P., Worring, M., de Rooij, O., Chua, T.-S., and Lua, H. 2008. Content-based video retrieval: Three example systems from TRECVid. *International Journal of Imaging Systems and Technology 18,* 2-3, 195–201.

[54] Smoliar, S. W. and Zhang, H. 1994. Content-based video indexing and retrieval. *IEEE MultiMedia 1,* 2, 62–72.

[55] Snoek, C. G. M., van de Sande, K. E. A., de Rooij, O., Huurnink, B., van Gemert, J. C., Uijlings, J. R. R., and et al. 2008. The MediaMill TRECVID 2008 semantic video search engine. In *Proceedings of the TRECVID Workshop*.

[56] Snoek, C. G. M. and Worring, M. 2009. Concept-based video retrieval. *Foundations and Trends in Information Retrieval 4,* 2, 215–322.

[57] van de Sande, K., Gevers, T., and Snoek, C. 2011. Empowering visual categorization with the GPU. *Multimedia, IEEE Transactions on 13,* 1 (February), 60–70.

[58] van de Sande, K. E. A., Gevers, T., and Snoek, C. G. M. 2010. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 32,* 9, 1582–1596.

[59] van Gemert, J. C., Veenman, C. J., Smeulders, A. W. M., and Geusebroek, J. M. 2010. Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence 32,* 7, 1271–1283.

[60] Vedaldi, A. and Zisserman, A. 2010. Efficient additive kernels via explicit feature maps. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2010)*.

[61] Vedaldi, A. and Zisserman, A. 2012. Efficient additive kernels via explicit feature maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 34,* 3 (march), 480 –492.

[62] Viitaniemi, V. and Laaksonen, J. 2006. Use of image regions in context-adaptive image classification. In *Proceedings of the 1st International Conference on Semantic and Digital Media Technologies (SAMT 2006)*, Y. Avrithis, S. Staab, and N. O'Connor, Eds. Lecture Notes in Computer Science. Springer, Athens, Greece, 169–183.

[63] Viitaniemi, V. and Laaksonen, J. 2007. Improving the accuracy of global feature fusion based image categorisation. In *Proceedings of the 2nd International Conference on Semantic and Digital Media Technologies (SAMT 2007)*, B. Falcidieno, M. Spagnuolo, Y. S. Avrithis, I. Kompatsiaris, and P. Buitelaar, Eds. Lecture Notes in Computer Science, vol. 4669. Springer, Genova, Italy, 1–14.

[64] Viitaniemi, V., Sjöberg, M., Koskela, M., and Laaksonen, J. 2008. Exploiting temporal and inter-concept co-occurrence structure to detect high-level features in broadcast videos. In *Proceedings of 9th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2008)*. Klagenfurt, Austria, 12–15.

[65] Wu, J. 2012. Power mean SVM for large scale visual classification. In *Proceedings of The IEEE Int'l Conference on Computer Vision and Pattern Recognition (CVPR 2012)*. Providence, USA.

[66] Yang, J., Yu, K., Gong, Y., and Huang, T. 2009. Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 1794 –1801.

[67] Yilmaz, E. and Aslam, J. A. 2006. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of 15th International Conference on Information and Knowledge Management (CIKM 2006)*. Arlington, VA, USA.

[68] Yilmaz, E., Kanoulas, E., and Aslam, J. A. 2008. A simple and efficient sampling method for estimating AP and NDCG. In

*Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '08)*. 603–610.

[69] Yuan, G.-X., Ho, C.-H., and Lin, C.-J. 2012. Recent advances of large-scale linear classification. *Proceedings of the IEEE 100,* 9, 2584 –2603.

[70] Zeiler, M. and Fergus, R. 2013. Visualizing and understanding convolutional networks. arXiv:1311.2901.

[71] Zhou, X., Yu, K., Zhang, T., and Huang, T. 2010. Image classification using super-vector coding of local image descriptors. In *Proceedings of European Conference on Computer Vision (ECCV 2010)*.

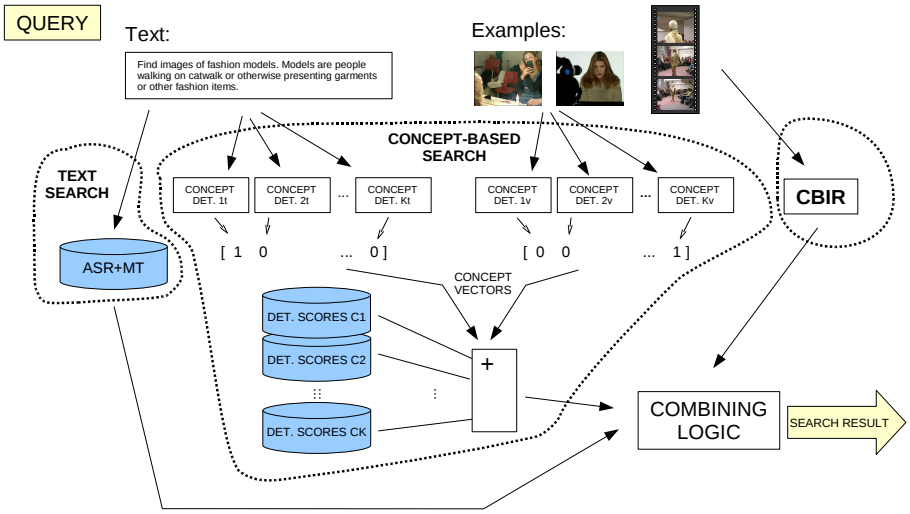



Figure 1: General architecture of the PicSOM multimedia retrieval system when applied to video search. Concept-based search is supplemented with textual and content-based (CBIR) search. The text is extracted from the video soundtracks using a combination of automatic speech recognition (ASR) and machine translation (MT).

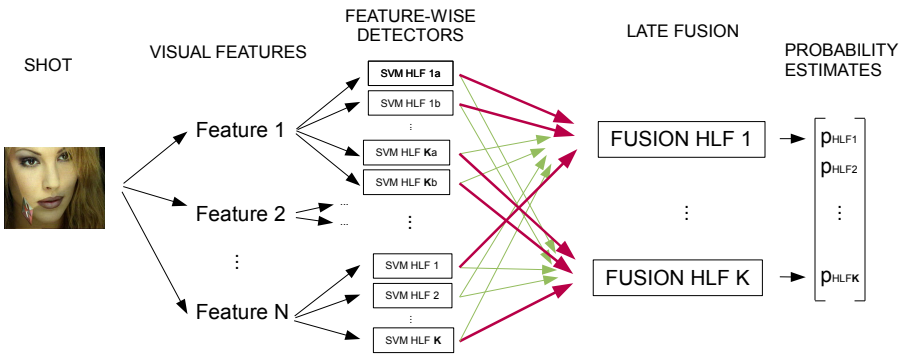



Figure 2: Fusion-based shot-wise concept detection module in PicSOM system. $K$ denotes the number of concepts that are to be detected. The solid red lines between the feature-wise detector and fusion stages are intra-concept connections, the dashed green lines represent cross-concept links.

Figure 3: A TS-SOM partitioning of the feature space defined by color and texture distribution of image segments. From [62].

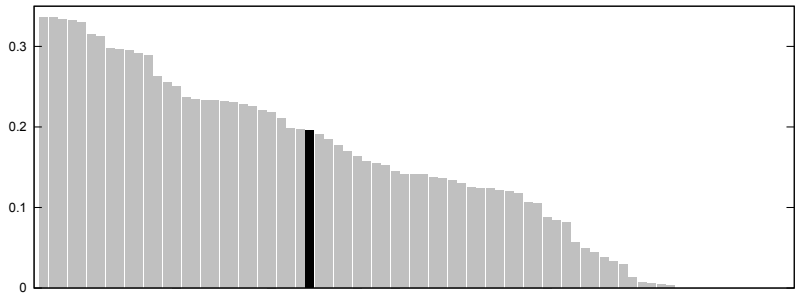
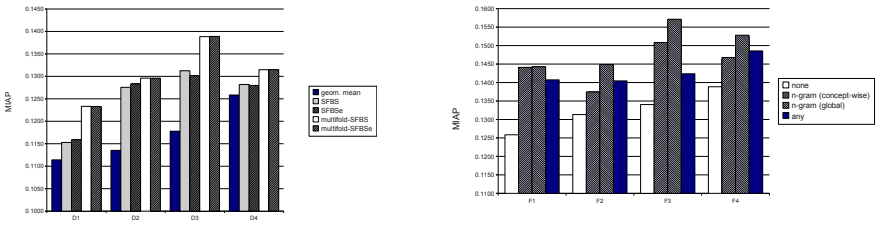



Figure 4: Mean average precision values for all runs submitted to the TRECVID 2005 high-level feature extraction task, our run highlighted




(a) (b)

Figure 5: (a) Comparison of algorithms for selecting detectors for geometric mean fusion for four different sets of detectors D1–D4. The SFBSe and multifold-SFBSe bars with diagonal hatching correspond to algorithms with early stopping (b) The effect of applying temporal post-processing on four different shot-wise fusion based detectors. The bars with diagonal hatching correspond to the $N$-gram technique with two different strategies for order selection.

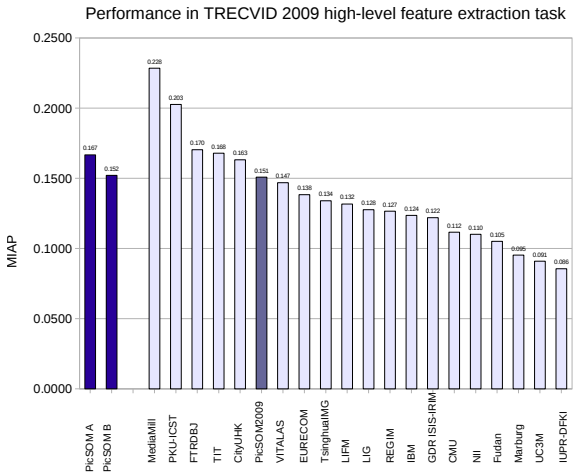



Figure 6: The MIAP performance in TRECVID 2009 high-level feature extraction task compared with the systems submitted by the best groups to the evaluation. The dark bars correspond to the PicSOM system discussed here, not any submitted system. Note that the figures show only the best-performing end of the distribution, all the systems are significantly more accurate than median MIAP 0.049 of the submissions.

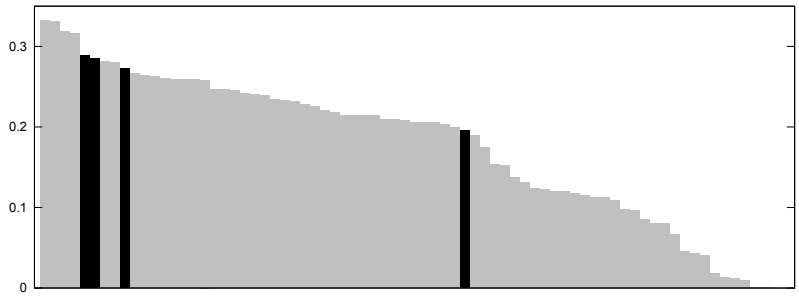



Figure 7: MXIAP values for all submissions to the TRECVID 2014 semantic indexing task, our runs highlighted.

Table 1: Features used in TRECVID 2005 in the high-level feature extraction task for each concept.

| high-level feature | video | | | | image | | | | | | audio | text | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MA | AC | CM | TN | CL | CS | DC | SC | EH | HT | CE | 10 | 100 |
| Walking/Running | × | × | | | | | | | × | | | | |
| Explosion/Fire | | | × | | × | | | | | × | | | × |
| Maps | | × | × | | | × | | × | × | × | | | × |
| Flag-US | × | | × | × | | | | | × | | | × | |
| Building | | × | × | | | | × | | × | × | | | × |
| Waterscape/Waterfront | × | × | × | × | × | | | | × | | | × | |
| Mountain | × | × | | × | | | | | × | × | × | | |
| Prisoner | | | | | | | | | | × | | | |
| Sports | × | × | × | | × | | | | × | | | × | |
| Car | × | × | × | × | × | | | | × | × | × | | × |

Table 2: The 20 concepts detected in TRECVID 2009 high-level feature extraction task.

| | | |
|---|---|---|
| Classroom | Person playing a musical instrument | Hand |
| Chair | Person playing soccer | People dancing |
| Infant | Cityscape | Nighttime |
| Traffic | Person riding a bicycle | Boat or ship |
| Doorway | Telephone | Female human face closeup |
| Airplane flying | Person eating | Singing |
| Bus | Demonstration or protest | |

Table 3: Concept detection accuracy based on various BoV image features in TRECVID 2009.

| Feature | sampling | histograms | spatial partitioning | MIAP |
|---|---|---|---|---|
| Color SIFT | dense | soft histograms | spatial pyramid | 0.1166 |
| Color SIFT | dense | soft histograms | global | 0.1031 |
| Color SIFT | interest points | soft histograms | spatial pyramid | 0.1014 |
| Color SIFT | interest points | soft histograms | global | 0.0961 |
| Color SIFT | dense | hard histograms | global | 0.0988 |
| SIFT | interest points | hard histograms | global | 0.0832 |

Table 4: A selection of feature-wise concept detection accuracies in TRECVID 2009.

| Feature | type | MIAP |
|---|---|---|
| Edge Histogram | video | 0.0625 |
| Color Moments | image | 0.0438 |
| MPEG-7 Edge Histogram | image | 0.0417 |
| Edge Histogram | image | 0.0403 |
| Color Layout | video | 0.0340 |
| Color Layout | image | 0.0309 |
| Scalable Color | image | 0.0330 |
| Edge Fourier | image | 0.0290 |
| MPEG-7 Color Structure | image | 0.0263 |

Table 5: An overview of our runs submitted for the TRECVID 2014 evaluation.

| | features | | | | hard neg. | |
|---|---|---|---|---|---|---|
| id | glob. | BoV | FV | CNN | mining | MXIAP |
| 1 | • | • | | | | 0.1951 |
| 2 | | • | • | • | | 0.2722 |
| 3 | | | | • | • | 0.2843 |
| 4 | | • | • | • | • | 0.2880 |