Fisher Kernel Temporal Variation-based Relevance Feedback for Video Retrieval

Ionuţ Mironică[1], Bogdan Ionescu[1], Jasper Uijlings[2], Nicu Sebe[3]

[1] *LAPI, University Politehnica of Bucharest, 061071 Romania*
{imironica,bionescu}@imag.pub.ro

[2] *University of Edinburgh, UK*
jrr.uijlings@ed.ac.uk

[3] *DISI, University of Trento, Italy*
sebe@disi.unitn.it

Abstract

This paper proposes a novel framework for Relevance Feedback based on the Fisher Kernel (FK). Specifically, we train a Gaussian Mixture Model (GMM) on the top retrieval results (without supervision) and use this to create a FK representation, which is therefore specialized in modelling the most relevant examples. We use the FK representation to explicitly capture temporal variation in video via frame-based features taken at different time intervals. While the GMM is being trained, a user selects from the top examples those which he is looking for. This feedback is used to train a Support Vector Machine on the FK representation, which is then applied to re-rank the top retrieved results. We show that our approach outperforms other state-of-the-art relevance feedback methods. Experiments were carried out on the Blip10000, UCF50, UCF101 and ADL standard datasets using a broad range of multi-modal content descriptors (visual, audio, and text).

Keywords: relevance feedback, Fisher Kernel representation, multimodal content description, video retrieval.

1. Introduction

Understanding video content is in general a subjective process for a user. Labeling video content with a predefined set of labels can greatly facilitate search, but is unlikely to capture all possible viewpoints of users. Hence

finding specific video content in the exponentially growing amount of digital video becomes increasingly difficult. One solution to this problem is to empower the user with personalized search by iteratively having the user refine its search queries. This is called Relevance Feedback (RF) and is the topic of this paper. A general RF scenario for video retrieval can be formulated as follows: First the user does an initial query using either a keyword or a specific video for which he wants related videos. After the system has returned the best matching results, the user indicates which videos are relevant and which are not. Results are updated using the input, and this refinement continues until the user is satisfied.

In this paper we propose a novel framework for Relevance Feedback based on the Fisher Kernel (FK) and Support Vector Machines (SVMs). The proposed approach operates on top of an existing retrieval system by refining the initial results. First, we alter the feature space: we train a Gaussian Mixture Model (GMM) on the top retrieved results, after which we obtain the FK representation with respect to the GMM. Hence the new feature space is specialized in representing the top results that are representative. Afterwards, we train an SVM using the user feedback, yielding a specialized classifier in the new feature space. Therefore, we have an unsupervised step which alters the feature space and a supervised step to incorporate user feedback. The entire process is illustrated in Figure 1.

Additionally, we propose to use the FK to capture temporal information as follows: we cut a video up in smaller temporal segments, extract a fixed-size feature representation for each segment, and represent the resulting feature set using the FK. Notice that since the FK captures variation in features in general, and we vary the features in time only, we effectively capture the temporal variation using this representation (but not the temporal order). This differs from other uses of the FK: The representation of images using SIFT [1] and FK leads to a representation of the local visual variation *in space only* while no temporal information is captured which discards meaningful video information; Representing videos using local Histograms of Oriented Gradients (HoG)/Histograms of Optical Flow (HoF) descriptors [2] and the FK leads to a representation of the local variation in *both time and space simultaneously*, where space and temporal information is mixed together thus reducing their individual representative power instead of exploiting it. In our approach, by having fixed-sized representations of single frames or small temporal segments with FK we manage to exploit the variation *in time only* thus capturing that unique video temporal characteristics. As experimental

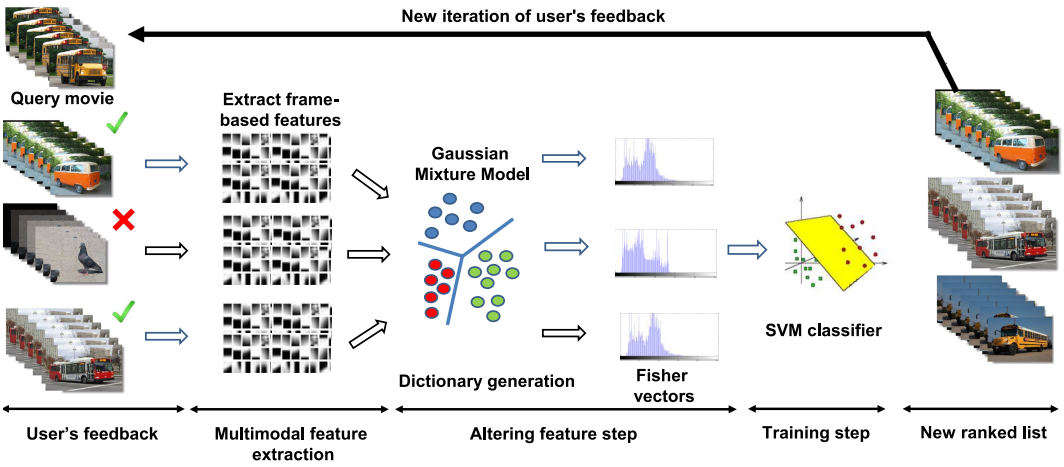

Figure 1: Diagram of the proposed Fisher Kernel Relevance Feedback (displayed images are from the *Blip10000* dataset [5]).

results show, this proves highly efficient in relevance feedback based retrieval scenarios.

This paper extends our previous work [3, 4] by including evaluation on a new video dataset, evaluating more feature extraction schemes, analyzing the influence of multiple relevance feedback iterations, and including a computational complexity analysis. To summarize, our main contributions are as follows:

1. We propose a novel method for Relevance Feedback based on a combination of the FK and SVMs. To the best of our knowledge, this is the first work that exploits the benefits of FK representations to video relevance feedback;
2. We explicitly model temporal variation by combining frame-based features with the FK;
3. We demonstrate the generality of our approach by evaluating it on a broad range of modalities: we use visual, audio, and text descriptors. We achieve better performance than other state-of-the-art relevance feedback algorithms on two standard datasets [5, 6], which makes the results both relevant and reproducible.

The remainder of the paper is organized as follows. In Section 2 we

present the current state-of-the-art on relevance feedback and FK and position our contribution. Section 3 presents the Fisher Kernel theory and Section 4 presents the algorithm of the proposed FK relevance feedback. Afterwards, in Section 5 we present an expansion of the method for temporal aggregation with FK. The experimental setup is presented in Section 6, while the experimental results are reported in Section 7. Finally, in Section 8 we conclude the paper.

2. Related Work

For decades now, content-based retrieval systems focused on bridging the semantic gap [7] by linking the low-level video features and their high-level semantic interpretation. Video content classification remains one of the most challenging video processing problems, mainly because it implies the classification of complex semantic categories from a huge volume of multimodal data. To effectively handle the quick growth of multimedia data with the objective of providing user access to the needed information, a large number of methods have been proposed for multimedia content analysis and retrieval [70, 72, 73, 74, 75]. In this context, a standard video classification system consists of detecting sparse spatio-temporal interest points which are then described using local spatio-temporal features, e.g., Histograms of Oriented Gradients (HoG), Histograms of optical Flow (HoF), Motion Boundary Histograms (MBH) or spatio-temporal Laplacian pyramids [74]. The features are then encoded into an aggregated representation, such as Bag of Words (BoW), Fisher Kernel [4], Vector of Locally Aggregated Descriptors (VLAD) [76] representations, or sparsely-constructed Gaussian processes [75] that are then combined with a classifier.

Existing state-of-the-art algorithms for video classification can achieve promising performance in benchmarking for many research challenges, starting from genre classification to event and human activity recognition. For instance, TRECVID [12] recently introduced a Multimedia Event Detection (MED) Track on detecting complex events in web videos when only few positive exemplars are available. Some interesting approaches have been successfully implemented, such as the one proposed by Yang et al. [70] where related exemplars which convey the precise semantic meaning of an event are used for complex event detection. The relatedness is automatically learned and soft labels are assigned to related exemplars adaptively.

Convolutional Neural Networks (CNNs) have been also proved to be an effective class of models for understanding video content, giving state-of-the-art results in many video recognition problems. For instance, Xu et. al. [71] introduces a new encoding technique that generates a video representation based on CNN descriptors. In this approach, a set of latent concept descriptors are used as frame descriptors, which further diversifies the output with aggregation on multiple spatial locations at deeper stage of the network.

2.1. Relevance feedback

In order to improve the retrieval performance, an efficient solution is to take advantage directly of the user's feedback through Relevance Feedback (RF) techniques. RF helps users improve the quality of their query statements and has been shown to be effective in many experimental environments, e.g., Ma et al. [8], Yang et al. [9], Jones et al. [10], Wang et al. [11]. The main idea behind RF is to take the results that are initially returned for a given query, and use the user's feedback to refine them. Recently, a relevance feedback track was organized by TREC to evaluate and compare different relevance feedback algorithms for text retrieval [12]. However, relevance feedback was successfully applied also for image [13, 14], multimodal retrieval [15], biomedical approaches [16], etc.

Many RF approaches have been proposed in the literature [17]. They can be grouped into *pseudo-relevance feedback*, *implicit relevance feedback* and *explicit relevance feedback*.

Pseudo-relevance feedback [18] automatically simulates the user feedback without any interaction. The assumption is that only a small number of the top-ranked documents in the initial retrieval results are relevant, and these are used for re-ranking the results. *Implicit relevance feedback* [19] uses the recorded actions of the users to simulate the feedback. The user behaviour, such as clicking on documents, clicking the browser "back" button, time spent per web-page or scrolling, are unobtrusively monitored and used to expand the futures queries. Finally, *explicit relevance feedback* is a framework in which the user is explicitly asked which results are desirable and which not in an interactive system. It requires more effort on the user side but is also much more reliable.

In this paper we address the *explicit relevance feedback* scenario. There are several ways to incorporate relevance feedback: by *changing the query points*, by *altering the feature representation*, and by *using classification*.

Changing query points was done by one of the earliest and most successful RF algorithms proposed by Rocchio et al. [20]. Using the set of R relevant and N non-relevant documents selected from the current user relevance feedback window, the Rocchio's algorithm modifies the feature of the initial query by adding the features of positive examples and subtracting the features of negative examples to the original feature. This class of RF algorithms are also known as Query Point Movement (QPM) approaches. A more recent extension of this work in the context of video data is proposed by Nguyen et al. [21].

Altering the feature representation was proposed by Rui et al. [22] in their Relevance Feature Estimation (RFE) algorithm, which assumes that for a given query, according to the user's subjective judgement, some specific features may be more important than other features. Recent extensions of this technique have been proposed by Yuanhua et al. [23]. It uses a probabilistic relevance model that exploits the term position and proximity evidence, and assigns more weights to closer semantic terms. The techniques from this category proved extremely fast and simple. However, one of the main drawbacks is that the re-weighted function cannot be fully generalized and adapted because of the diversification of multimedia concepts [24].

Finally, relevance feedback can be performed *using classification*. After an initial query the user indicates which results are positive and negative examples. These are used to train a classifier and update the results. Some of the most successful techniques use Support Vector Machines [25], classification trees, e.g., Random Forest [14], or boosting techniques, e.g., AdaBoost [26]. Another perspective of the machine learning RF are the approaches that exploit some *adaptive learning techniques*. Yuanhua et al. [24] propose a RF algorithm which adaptively predicts the balance coefficients between query and feedback information, using a regression approach, and then, it re-ranks the documents according to these coefficients. Su et al. [27] propose a Navigation-Pattern-based RF (NPRF) to achieve high performance for web image retrieval. The NPRF search makes use of the discovered navigation patterns and various query refinement strategies, e.g., QPM and RFE, to converge the search space toward the user's intention effectively. However, all these techniques tend to be less efficient when there is only a limited number or an asymmetric number of positive and negative feedback samples provided by the user. There have been several attempts to overcome this using Biased Discriminant Euclidean Embedding [28] and Active Re-ranking for Web Image Search [29]. In what concerns specifically the video relevance

feedback we can mention the approach proposed by Mei et al. [30] which uses a combination of multimodal descriptors with relevance feedback. It is based on a weighting strategy for each modality followed by re-ranking. Another relevant example is the approach in Shao et. al. [73] which introduces a new content-based video retrieval framework for searching video databases with human actions. It specifically incorporates spatio-temporal localization. They outline an efficient localization model that first performs temporal localization based on histograms of evenly spaced time-slices, followed by the spatial localization based on histograms of a 2-D spatial grid. As a final step, the framework uses a relevance feedback algorithm that enhance even more the performance of localization and ranking.

2.2. Fisher Kernel

The Fisher Kernel was introduced by Jaakkola et al. [31] to combine generative and discriminative methods. Specifically, a collection of features is represented by its gradient with respect to a generative distribution. The resulting vector is then used in discriminative classifiers. FK where introduced in computer vision by Perronnin et al. [32], which applied the FK framework to represent collections of local visual features such as SIFT [1] using Gaussian Mixture Models as generative distribution. FKs found their application in other fields as well, starting from topic-based text segmentation [35] to web audio genre classification [34]. Sun et al. [35] proposed a latent Dirichlet allocation (LDA)-based FK to exploit text semantic similarities, then employed dynamic programming to obtain global optimization. Aran and Akarun [33] introduced a multi-class classification strategy for a sign language data set. They applied a multi-class classification on each Fisher score space and combined the decisions of multi-class classifiers. They showed experimentally that the Fisher scores of one class provide discriminative information for the other classes as well. More recently, FK representation was used by Myers et al. [36] for detection of user-defined events. They propose a set of multimodal features (i.e., audio, motion, visual) together with a set of late fusion techniques.

In this paper we adopt the Fisher Kernel for Relevance Feedback in video for two purposes. (1) Instead of learning the generative probability distribution over all features of the data, we learn it only over the top retrieved results. Hence during relevance feedback we create a new FK representation based on the most relevant examples. (2) In addition, we use the FK to

capture temporal variation. We do this by cutting up a video in smaller segments, extract a feature vector from each segment, and represent the resulting feature set using the FK model. Since the variation in features is caused by varying time only, we effectively capture the temporal variation. This approach is in particular interesting for enriching the representative power of the video description scheme.

3. Fisher Kernel Theory

The main idea behind FK representation is to describe a signal as the gradient of the probability density function that is a learned generative model of that signal. Intuitively, such representation measures how to modify the parameters of the probability density function in order to best fit the signal, similar to the measurements in a gradient descent algorithm for fitting a generative model [31]. The FK representation obtained is then used in a discriminative classifier to solve the classification problem.

Given a collection of T labeled multimodal video descriptors, $X = \{x_1, x_2, ..., x_T\}$, X can be represented by its gradient vector with respect to a Gaussian Mixture Model (GMM), u_λ, with parameters λ:

$$\mathrm{G(X)}_\lambda = \frac{1}{\mathrm{T}} \bigtriangledown_\lambda \log\left(\mathrm{u}_\lambda(\mathrm{X})\right) \tag{1}$$

where $\bigtriangledown\{.\}$ is the gradient operator. Also, we assume that the covariance matrices are diagonal.

Intuitively, the gradient of the log-likelihood describes the direction in which the model parameters should be modified to best fit the data. In our approach, we will exploit this property to learn the user's feedback when only a small amount of samples are available. The dimensionality of this vector depends only on the number of parameters in λ, and not on the number of descriptors T [31].

The gradient vector is, by definition, the concatenation of the partial derivatives with respect to the model parameters. Let μ_i and σ_i be the mean and the standard deviation of i's Gaussian centroid, $\gamma(i)$ be the soft assignment of descriptor x_t to Gaussian i (with $t = 1, ..., T$), and let D denote the dimensionality of the descriptors x_t. $G^x_{\mu,\sigma,i}$ is the D-dimensional gradient with respect to the mean μ_i and standard deviation σ_i of Gaussian

i. Mathematical derivation leads to [32]:

$$\mathrm{G}^{\mathrm{x}}_{\mu,\mathrm{i}} = \frac{1}{\mathrm{T}\sqrt{\omega_{\mathrm{i}}}} \sum_{\mathrm{t}=1}^{\mathrm{T}} \gamma(\mathrm{i}) \frac{\mathrm{x_t} - \mu_{\mathrm{i}}}{\sigma_{\mathrm{i}}} \tag{2}$$

$$\mathrm{G}^{\mathrm{x}}_{\sigma,\mathrm{i}} = \frac{1}{\mathrm{T}\sqrt{2\omega_{\mathrm{i}}}} \sum_{\mathrm{t}=1}^{\mathrm{T}} \gamma(\mathrm{i}) \left[\frac{(\mathrm{x_t} - \mu_{\mathrm{i}})^2}{{\sigma_{\mathrm{i}}}^2} - 1 \right] \tag{3}$$

where the division between vectors is a term-by-term operation.

Using this representation, the final gradient vector G^x, i.e., our new descriptor, is the concatenation of the $G^x_{\mu,i}$ and $G^x_{\sigma,i}$ vectors, for $i = 1, ..., T$. This leads to a $2 \cdot T \cdot D$ dimensional vector compared to the initial feature vector of size D.

In this paper, we exploit further the formalization introduced by [32]. The novelty of this paper is represented by the adaption of the algorithm to the relevance feedback problem. Also, instead of using the keypoint descriptors, the proposed FK representation is applied on a frame-based representation that allows capturing the variation in time (see Section 5). Furthermore, we demonstrate the generality of the method by using several multimodal features, stating from audio, text, motion and global visual features. The proposed approach is introduced in the following section.

4. The Fisher Kernel for Relevance Feedback

Our method is visualized in Figure 1 and presented with Algorithm 1. First, an initial ranking is obtained using a Nearest Neighbour query with the objective of simulating an initial retrieval system. Then, our Relevance Feedback mechanism works in two steps: (1) *Altering Features*. Based on the top n results we train a Gaussian Mixture Model on the top n videos. We represent the top k videos using the FK with respect to this GMM, where $n << k$. (2) *Training*. After the user has labelled the top n videos (n is in general small), we train a SVM classifier on the FK representation. We apply this classifier to the top k videos. We now describe these two steps in detail.

4.1. Altering Features

Initially, given a user query, we use a nearest-neighbour search to return a ranking of the most likely videos. We take the top n videos and train

Algorithm 1: FK relevance feedback approach

Initial parameters:

Labeled video sample set: V_{t_i} *and labels* Y_i;
Labeled video features set: $X_{t_{ik}}$;
Unlabeled video sample set: V_{nr};
Unlabeled video features set: $X_{n_{rk}}$;
SVM Classifier parameters (C, γ*)*;
n*: the window size*;

Start:

do PCA reduction for all multimodal features;

Altering features step:

Compute GMM centroids for Xt_{ik};
for $x \subset X_t$ **do**
    compute $FK(X_{t_{ik}}) = FK(X_{t_{ik}}, GMM)$;
    normalize $FK(X_{t_{ik}})$;

Training - reranking step:

train SVM(C, γ*) using FK features*;

for $x \subset X_n$ **do**
    compute $FK(X_{n_{rk}}) = FK(X_n, GMM)$;
    normalize $FK(X_{n_{rk}})$;
    compute $h(X_{n_{rk}})$ = *SvmConfidenceLevel(*$FK(X_{n_{rk}})$*)*;

sort $h(X_n)$ *values*;
show new ranked list according to $h(X_n)$ *values*;

a GMM model using a diagonal covariance matrix on the features of these videos. Hence we change the representation of our feature space based on the highest ranked examples. In our experiments, we use $n = 20$. Since our method of altering the feature space is unsupervised, the GMM can be trained in the background during the time that the user is providing feedback. Initially, in order to speed up the GMM algorithm, we initialize the centroids using k-means, that allows a fast convergence. The GMM contains several parameters which impact the performance of the algorithm: the number of clusters c, the size of video features and the normalization techniques.

The number of clusters c is proportional with the size of FK representations, thus, for a practical system the number of clusters has to be low. Also, the feature's size represents another parameter which is proportional with the final size of FK representation. Therefore, to make the FK computationally feasible, we first apply Principal Component Analysis (PCA) on the original feature vectors of the videos. After obtaining the mixture model, we convert the original features of the top k videos into the FK representation as presented in equations 2 and 3. Note that in some experiments we only use a single cluster ($c = 1$). In this case the FK representation consists of both the absolute and quadratic Mahalanobis distance to this single Gaussian cluster, which we show to be a good alteration of the feature space.

The final step is the normalization of the obtained FK representation. It has been shown in [31] that normalization significantly increases the performance of the FK representations. Details are presented in Section 7.1.

4.2. Training - re-ranking step

The training step is represented by a SVM classifier. The classic binary SVM training algorithm builds a linear margin that maximizes the distance between two classes, but also it can efficiently perform a non-linear classification. More recently, SVMs found their application with relevance feedback approaches. In these approaches, the relevance feedback problem can be formulated either as a two class classification of the negative and positive samples or as an one class classification problem [37], i.e., separate positive samples by negative samples.

In our algorithm, the SVM model is trained on the top n documents, according to the user's feedback (e.g., $n = 20$). After the training step, the top k documents are ranked according to the SVMs confidence level, where k is typically 1000 in our experiments ($n << k$). The reason to only re-rank the top k is two-fold. First of all, it is unlikely that relevant videos are ranked lower than the top k of the initial query. Using only the top k is faster. Second, the GMM is trained on the top n examples only. This means that the feature space is highly suited for representing examples close to the top n, but less suited for examples further away.

We use cross-validation to optimize the slack-parameters. We can do this fast because due to the Relevance Feedback we have only few training examples.

5. Frame Aggregation with Fisher Kernel

Global features are often used for reasons of computational efficiency. Also, most of the video content description approaches compute a global representation obtained by aggregating those frame-based features. However, frames in videos change over time, so an important question is: how can we meaningfully aggregate frame-based features in order to preserve that variation in time? One method is to aggregate all the features in one descriptor by computing the mean over all the frames, but the variation in time is ignored [42, 43]. Those approaches mix information, disregarding appearance variation over time. Alternatively, a video document can be represented as a set of multiple vectors and the similarity between two videos may be computed as the distance between two sets of points using, for example, the Earth Mover distance [45]. However, these metrics involve a huge computational cost for large databases.

The FK representation was created to model the variation of a set of vectors into a single fixed-sized representation. Hence by using features from different time-steps, we effectively model the variation in time. In the context of Relevance Feedback, we train a GMM only on the top n videos. This creates a feature space specialized to represent differences between relevant examples.

For cutting up the video into temporal chunks, we select T frames from each video and we compute a visual feature for each video frame: $X_k = \{x_1, x_2, ..., x_T\}$. Then, we train a GMM on the features of the top n videos X_{ik}, where $i \leq n$ and $k \leq T$. Once the generative model is trained, every training sequence of feature vectors, $X_i = \{x_1, x_2, ..., x_T\}$, is transformed into a vector of fixed length. The main difference between the previous approach proposed in Section 4.1 and this one concerns the data the GMM was learned on. Instead of using one global aggregated video feature, we will use more features per document. By using this approach, the GMM will learn from more data and the final FK representation contains more information. It is worth mentioning that the resulting FK representation will still have the same dimension.

Experiments in Section 7.4 show the performance of the frame aggregation on the relevance feedback (denoted as $T \geq 1$), while experiments in Section 7.2 to 7.3 discuss the performance of the algorithm when we use only one video global descriptor (denoted as $T = 1$).

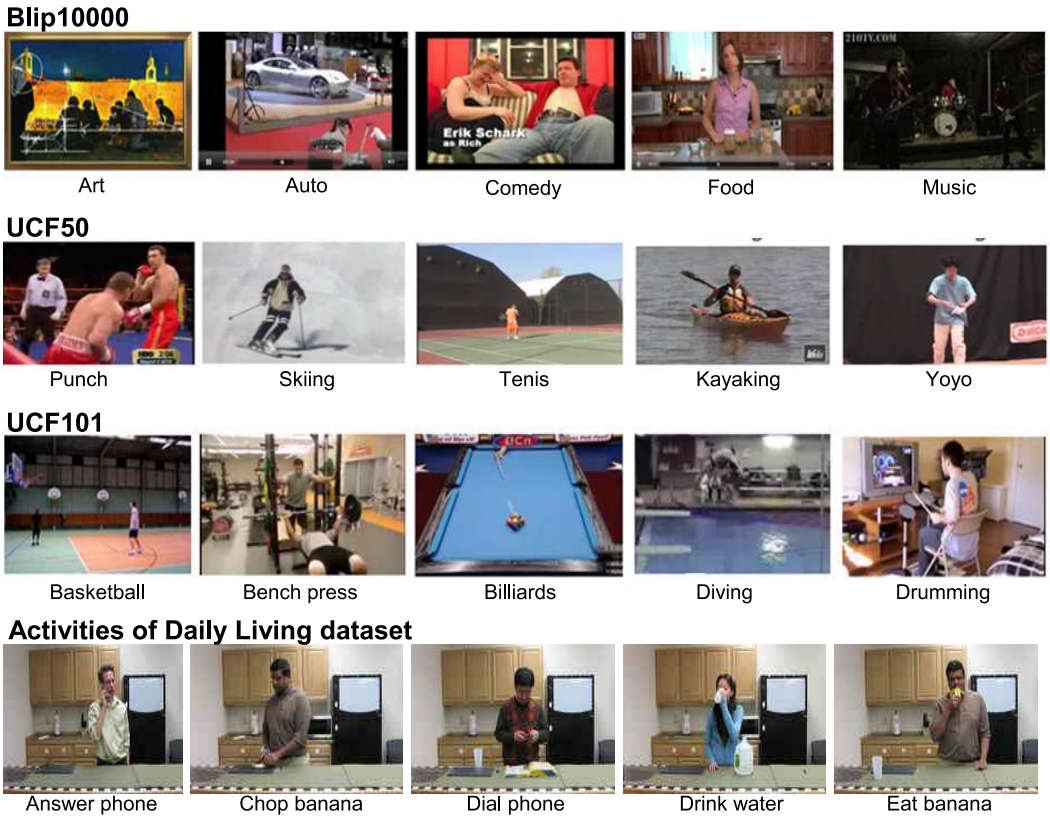

Figure 2: Sample images from the *Blip10000* [5], *UCF50* [6], *UCF101* [61] and *ADL* [62] datasets.

6. Experimental Setup

In this section we discuss the evaluation framework (dataset and metrics) and the choice of content descriptors.

6.1. Datasets

The validation of the proposed relevance feedback approach was carried out on four standard video datasets, namely: *Blip10000* - Video Genre Tagging dataset [5], *UCF50* - Sport Action Recognition dataset [6], *UCF101* - Action Recognition dataset [61] and *ADL* - Daily Activities dataset [62].

Blip10000: consists of 15,000 video sequences (with more than 3,250 hours of footage) retrieved from blip.tv[1] web platform. Each video is labeled according to 26 web specific video genre categories, namely: art, autos and vehicles, business, citizen journalism, comedy, conferences and other events, documentary, educational, food and drink, gaming, health, literature, movies and television, music and entertainment, personal or auto-biographical, politics, religion, school and education, sports, technology, the environment, the mainstream media, travel, videoblogging and web development and sites. A "default category" is provided for movies which cannot be assigned to neither one of the previous categories. Apart from the video data, the dataset provides associated social metadata, automatic speech recognition transcripts (ASR transcripts) and shot information including key frames. The dataset was successfully validated during 2010-2012 MediaEval benchmarking campaigns [38].

UCF50: consists of 6,600 realistic videos from YouTube[2] with large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. Videos are labeled according to 50 action categories, namely: baseball pitch, basketball shooting, bench press, biking, billiards shot, breaststroke, clean and jerk, diving, drumming, fencing, golf swing, playing guitar, high jump, horse race, horse riding, hula hoop, javelin throw, juggling balls, jump rope, jumping jack, kayaking, lunges, military parade, mixing batter, nun chucks, playing piano, pizza tossing, pole vault, pommel horse, pull ups, punch, push ups, rock climbing indoor, rope climbing, rowing, salsa spins, skate boarding, skiing, soccer juggling, swing, playing tabla, TaiChi, tennis swing, trampoline jumping, playing violin, volleyball spiking, walking with a dog, and YoYo.

UCF101: consists of 13,320 realistic videos from YouTube with large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. The videos are labelled according to 101 action categories, with each containing 25 groups (each group consisting of 4 to 7 videos of an action). The videos from the same group may share some common features, such as similar background, similar viewpoint, etc. The action categories can be divided into five types: human-

[1] `http://blip.tv/`
[2] `http://www.youtube.com/`

object interaction, body-motion only, human-human interaction, playing musical instruments and sports.

ADL: contains 10 different activities, i.e., answering a phone, dialing a phone, looking up numbers in a phone book, writing on a white board, drinking water, eating a snack, peeling a banana, eating a banana, chopping a banana and eating food with silverware. Each of these activities is performed 3 times by 5 different people. These people have different genders, ethnicity, and appearance so sufficient appearance variation is available in the dataset. Each clip is in the range of 3-50s. In total the dataset contains 150 videos.

These datasets are particularly challenging due to the diversity of video footage, and specifically the variability of videos within the same categories, as well as due to the number of high level concepts proposed. Figure 2 illustrates some image examples in this respect.

6.2. Evaluation

In all the experiments we consider the scenario where user feedback is automatically simulated with the known class membership of each video document, retrieved from the ground truth. This approach allows a fast and extensive simulation which is necessary to evaluate different methods and parameter settings, otherwise impossible with realtime user studies. Such simulations represent a common practice in evaluating relevance feedback scenarios [25, 22, 26, 27, 30].

To assess retrieval performance, we use several metrics. Firstly, we compute precision and recall. Precision represents the fraction of retrieved videos relevant to the find (measure of false positives) and recall is the fraction of the videos relevant to the query that are successfully retrieved (measure of false negatives). The retrieval response of the system is assessed with the precision-recall curves, which plot the precision for all the recall rates that can be obtained according to the current video class population.

Secondly, to provide a global measure of performance, we estimate the overall Mean Average Precision (MAP), which is computed as the mean of the average precision scores for each query:

$$\mathrm{MAP} = \sum_{\mathrm{q}=1}^{\mathrm{Q}} \frac{\mathrm{AP(q)}}{\mathrm{Q}} \tag{4}$$

where Q represents the number of queries, and $AP()$ is given by

$$\mathrm{AP} = \frac{1}{\mathrm{m}} \sum_{\mathrm{k}=1}^{\mathrm{n}} \frac{\mathrm{f_c}(\mathrm{v_k})}{\mathrm{k}} \tag{5}$$

where n is the number of videos, m is the number of videos of category c, and v_k is the k-th video in the ranked list $\{v_1, ..., v_n\}$. Finally, $f_c()$ is a function which returns the number of videos of genre c in the first k videos if v_k is of genre c and 0 otherwise (we used the trec_eval scoring tool[3]).

In our evaluation, we systematically consider each video from the database as query and retrieve the remainder of the database accordingly. Precision, recall and MAP are averaged over all retrieval experiments. Relevance feedback was collected from various browsing top n result windows, with n ranging from 10 to 30. For brevity reasons, in the following we shall present only the results obtained for $n = 20$, which we think it represents a good trade-off between user-input and the accuracy of the system. Moreover, experimenting with other values of n proved to have little influence on the overall conclusions.

6.3. Content descriptors

Video information is represented with content descriptors. At the moment, there is a huge amount of literature in this area, and covering all the existing techniques is impossible. For evaluation, we selected some of the representative approaches that are known to perform well in many benchmarking scenarios [12, 39, 40, 41] as well as which are adapted to our experimentation tasks (genre and action -based retrieval). The scope of this paper is to demonstrate the efficiency of the general Fisher Kernel representation relevance feedback framework which is not dependent on a particular type of descriptor but can be adapted to respond to various retrieval scenarios.

It is well know that different modalities tend to account for different information providing complementary discriminative power. For video content description we experiment with all the available sources of information, from the audio, and visual, to highly semantic textual information obtained with Automatic Speech Recognition (ASR) as well as user generated data (e.g., metadata that typically accompany video content on the Internet).

[3] `http://trec.nist.gov/trec_eval/`

Visual descriptors:

- *MPEG-7 related descriptors (1,009 values)* [50] - we adopted standard color and texture-based descriptors such as: Local Binary Pattern, autocorrelogram, Color Coherence Vector, Color Layout Pattern, Edge Histogram, Scalable Color Descriptor, classic color histogram and color moments. For each sequence, we aggregate the features by taking the mean, dispersion, skewness, kurtosis, median and root mean square statistics over all frames (exploiting all the statistical moments performs better than using only a few [59]);

- *HoG features (81 values)* [51] - exploits local object appearance and shape within an image via the distribution of edge orientations. The image is divided into small connected regions (3x3) and for each of them building a pixel-wise histogram of edge orientations is computed. In the end, the combination of these histograms represent the final descriptor;

- *structural features (1,430 values)* [52] - characterize the geometric properties of contours via a local/global space transformation. On this transformed space, parameters are derived to classify contour global geometry (e.g., arc, inflexion or alternating) and describe local aspects (e.g., degree of curvature, edginess, symmetry). These descriptors were reported to be successfully employed in tasks such as the annotation of photos and object categorization [53];

- *Bag-of-Visual-Words of SIFT features (20,480 values)* [54] - we extract a bag of words model (BoVW) over a selection of key frames (uniformly sampled). We use a visual vocabulary of 4,096 words (which represent a common value for video related tasks and gives good results on both the TRECVID and Pascal VOC datasets [60, 41]) and the keypoints are extracted with a dense sampling strategy. We use rgbSIFT features [54] and final descriptors are represented at two different spatial scales of a spatial pyramidal image representation [55];

- *color naming histograms (11 dimensions)* [56] - describes the global color contents and it maps colors to 11 universal color names. We select this feature, in addition to the classic color histogram, because the color naming histogram is designed as a perceptually based color naming metric that is more discriminative and compact;

- *Convolutional Neural Network descriptors (4,096 dimensions)* [63, 64, 68] - we use a set of Convolutional Neural Networks (CNN) features, using the protocol laid out from [63]. The employed CNNs were trained on either ImageNet 2010 or 2012 datasets, following as closely as possible the network structure parameters of Krizhevsky et al. [64]. We use the activations of the first fully-connected layer of each network as our features, which results in 4096-dimensional feature vectors.

Motion descriptors:

- *Histograms of optical Flow (72 dimensions)* [57] - computes a rough estimate of velocity at each pixel given two consecutive frames. We use optical flow at each pixel obtained using Lucas-Kanade method [57] and apply a threshold on the magnitude of the optical flow, to decide if the pixel is moving or stationary. For all the features, we divide the frames in 2x2 and 3x3 regions and then we compute the feature for each region [55];

- *3DHoG cuboids (72 dimensions)* [65] - we compute the Histogram of Oriented Gradients cuboids motion features. First of all, we compute each feature in 3D blocks with a dense sampling strategy, i.e., the gradient magnitude responses in horizontal and vertical directions are computed. Then, for each response the magnitude is quantized in k orientations, where k = 8. Finally, these responses are aggregated over blocks of pixels in both spatial and temporal directions and concatenated;

- *Body-part features (144 values)* [67]] - approximate the optical flow that is computed on the body-part components. Human pose and body-part motion obtained good results in many event detection categories [67, 66]. We extract the body-part components using the state-of-the-art body-part detector in [66] and compute at every frame for all 18 body parts a Histogram of Optical Flow in 8 orientations [67].

Audio descriptors:

- *block-based audio features (11,242 values)* [48] - capture the temporal properties of the audio signal. We choose a set of audio descriptors that are computed from overlapping audio blocks (groups of audio frames). On each block, we compute the Spectral Pattern which characterizes

the soundtrack's timbre, delta Spectral Pattern which captures the strength of onsets, variance delta Spectral Pattern which represents the variation of the onset strength over time, Logarithmic Fluctuation Pattern which captures the rhythmic aspects, Spectral Contrast Pattern, Correlation Pattern which compute the temporal relation of loudness changes, Local Single Gaussian Model and Mel-Frequency Cepstral Coefficients (MFCC). Sequence aggregation is achieved by taking the mean, variance and median over all audio blocks;

- *standard audio features (196 values)* [49] - we use a set of general-purpose audio descriptors, namely: Linear Predictive Coefficients, Line Spectral Pairs, MFCCs, Zero-Crossing Rate, spectral centroid, flux, rolloff and kurtosis, augmented with the variance of each feature over a certain window (we use the common setup for capturing enough local context that is equal to 1.28s). For a sequence, we take the mean and standard deviation over all frames.

Text descriptors:

- *TF-IDF of ASR data (3,466 values)* [38] - describes textual data obtained from Automatic Speech Recognition of the audio signal. Tf-Idf is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus, and it represents the product of two statistics, the term frequency (measures how frequently a term occurs in a document) and inverse document frequency (computes how much information the term provides, that is, whether the term is common or rare across all documents). For ASR we use the transcripts provided by [58] that proved highly efficient to genre classification [38].

Depending on the dataset (and available information) different descriptor combinations were employed. For *Blip10000* we use all the above mentioned descriptors except for the HoF motion information; we also use the combination of all visual descriptors and all the visual, audio and text features. All the visual and audio descriptors are normalized by using the L_∞ norm, and text descriptors with the cosine normalization. Descriptor aggregation is accomplished with an early fusion approach. For this dataset, we decided not to use motion features because of their high computational complexity which makes them inefficient. For *UCF50* and *UCF101* datasets we only use several of the visual descriptors, HoG (to account for feature points information), and color naming histogram (to account for color information); and

motion features that are more representative for this dataset (3DHoG and HoF). Also, we added the CNN features which obtained good results in many multimedia classification tasks. We did not use audio and text information because the videos from *UCF50 / UCF101 / ADL* datasets do not contain sound and metadata are not available. For the ADL dataset we use only the body-part and 3DHoG features which already provided state-of-the-art results in many approaches [67, 66].

7. Experimental results

To validate our approach we conducted several experiments which are presented in the following. The first experiment (Section 7.1) motivates the choice of the best feature - metric combination for the retrieval and we study the influence of Fisher Kernel parameters on system's accuracy. The second experiment (Section 7.2) deals with comparing our relevance feedback with other relevant work from the literature. A third experiment (Section 7.3) studies the relevance feedback performance with a global Fisher Kernel representation by learning the GMM model on the entire sequence. The fourth experiment (Section 7.4) investigates the benefits of using the Fisher Kernels representation locally to capture the temporal variation at frame level in the sequence. The final experiment (Section 7.5) consists in assessing the computational complexity of the proposed framework.

To introduce our approach and focus solely on its performance, we simulated an environment for the retrieval system on top of which the relevance feedback will operate. We select a classic Nearest-Neighbour strategy for retrieving the initial results. Using a video as query, we achieve a rank list of videos from the dataset. The user will label only a reduced number of documents, which constitutes the initial RF window.

7.1. Feature metrics and normalization

Distance metrics are often used to compare the similarity of two multimedia objects, each represented by a set of features in high-dimensional spaces. Motivated by the assumption that a better initial performance of the retrieval system will trigger a better relevance feedback performance, we have tested the performance of several metrics [44]: Euclidean, Manhattan, probabilistic divergence measures such as Canberra [47], intersection family: Cosine

Table 1: Best descriptor - metric combination for the initial retrieval without relevance feedback.

descriptor	**best metric**	**MAP**
Blip10000 dataset		
HoG	Euclidean	17.18%
structural features	Chi Square	11.58%
BoVW of SIFTs	Euclidean	19.85%
MPEG-7 and related	Mahalanobis	21.14%
standard audio features	Mahalanobis	29.26%
block-based audio features	Canberra	17.18%
TF-IDF on ASR	Bray Curtis	20.41%
UCF50 dataset		
color naming histogram	Chi Square	24.22%
HoG	Chi Square	26.34%
HoF	Chi Square	25.99%
CNN	Euclidean	27.38%
BoW-3DHoG	Euclidean	28.79%
UCF101 dataset		
color naming histogram	Chi Square	21.19%
HoG	Chi Square	24.22%
HoF	Chi Square	23.58%
CNN	Euclidean	24.79%
BoW-3DHoG	Euclidean	25.37%
ADL dataset		
Body-part features	Euclidean	57.31%
3DHoG features	Euclidean	51.22%

Distance, Chi-Square distance used in machine learning and data clustering, Bray Curtis [46], Mahalanobis [46], Kullback-Leibler divergence [46] and Earth's Mover distance [45].

Based on this experiment, each descriptor will be associated to a specific metric which provided the best retrieval accuracy. The results point out that most of the features have their own suitable metric. For instance, on the Blip10000 dataset, the best results are obtained with five different metrics: Euclidean, Chi Square, Mahalanobis, Canberra and Bray Curtis [46]. On the other hand, most of the best results on the UCF50 dataset are obtained with the Chi Square distance, with the exception of CNN features

and BoW-3DHoG, where the Euclidean distance gets the best results. Also, similar results are obtained on UCF101 dataset. Finally, on ADL dataset we obtained the best results with the Euclidean distance.

Although the descriptors provide in general for some of the metrics and same dataset more or less comparable performance, the distance measure still plays a critical role and may lead to performance variations which can be of more than 25%. For brevity reasons, we show only the best results. Table 1 summarizes the best performance descriptor - metric combination and their associated MAP values. These combinations are adopted in the following experiments.

FK parameters. Further, we study the influence of FK parameters on the system's performance. All experiments are done in the scenario of global FK representation (when T=1).

In the first test we study the influence of *the number of Gaussian centroids*. For both datasets the best results are obtained using only one GMM centroid. In this case the size of FK descriptors will be only 2 times bigger than of the video descriptor.

A second test consists in analyzing the influence of the *FK normalization strategies* on system's performance. By increasing the number of GMM centroids, the FK representation become sparser. In order to counteract this effect, we use some normalization strategies [32]. We tested four normalizations and some combination of them, namely: L_1 normalization, L_2 normalization, power normalization ($f(x) = sign(x)\sqrt{\alpha \cdot |x|}$, where $sign(x)$ is the signum operator that returns 1 if $x > 0$, 0 if $x = 0$ and -1 otherwise; and α represents a parameter of the normalization $0 \leq \alpha \leq 1$), and logarithmic normalization ($f(x) = sign(x)log(\alpha \cdot |x|)$). We obtained the best results when we use L_1 normalization, except for the text descriptors which lead to better results using the logarithmic normalization.

To compare our approach against other relevance feedback approaches from the literature we have selected the settings that provide the largest improvement in performance, i.e., one GMM centroid, L_1 normalization with power normalization for all UCF50 / UCF101 / ADL descriptors and Blip10000 audio and visual descriptors, while the logarithmic normalization is used for text descriptors.

The last parameter that has to be taken into consideration is the *SVM kernel*. Initial experiments showed that we obtained good results with linear and RBF kernels. In the next experiments, we will use both SVM kernels:

the linear SVM classifier and the nonlinear RBF kernel to study the impact of both linear and non-linear approaches.

7.2. Comparison with state-of-the-art

In this section we compare the performance of the proposed FK Relevance Feedback against other validated techniques from the literature, namely: the Rocchio's algorithm [20] (ROCCHIO), Relevance Feature Estimation (RFE) [22], and some classification-based approaches: Support Vector Machines (SVM RF) [25], AdaBoost (BOOST RF) [26] and Random Forests (RF - RF) [14].

Figure 3 presents the precision-recall curves after one iteration of relevance feedback for different descriptors. Generally, all relevance feedback strategies provide significant improvement in retrieval performance compared to the retrieval without relevance feedback (see the dashed green line in Figure 3). However, the proposed FK Relevance Feedback algorithm (with linear - FK linear, and RBF - FK RBF, kernels) tends to provide better retrieval performance in all cases (see the solid black and solid blue lines in Figure 3). For brevity reasons, we present only the charts for the Blip10000 and UCF50 datasets.

The results from Figure 3 are synthesized in terms of MAP in Table 2. On Blip10000 dataset the highest performance is obtained using the proposed FK RBF run on standard audio descriptors, with an increase of MAP from 29.3% (without RF) to 46.3%; as well as run on all the combined descriptors which yields an increase from 30.2% (without RF) to 46.8%. On the other hand, the smallest increase in performance is obtained with BoVW descriptors, which also achieve low results during MediaEval 2012 Tagging Task benchmarking [59];

On UCF50 dataset, the proposed approach obtains the best results on all the cases. The highest increase in performance is obtained using the FK RBF run on 3DHoG descriptors, an increase of MAP from 28.79% to 49.4%. Also, competitive results were obtained with the CNN features, namely 48.9% for the linear kernel and 47.5% for the RBF kernel. The smallest increase in performance is obtained with the FK RBF run on color naming histogram, namely increase of MAP from 24.22% to 31.7%. A reason for this could be the high diversity of classes for which the color naming histograms provide little representative power compared to the more discriminant 3DHoG descriptors. In consequence, in many cases there are not sufficient positive

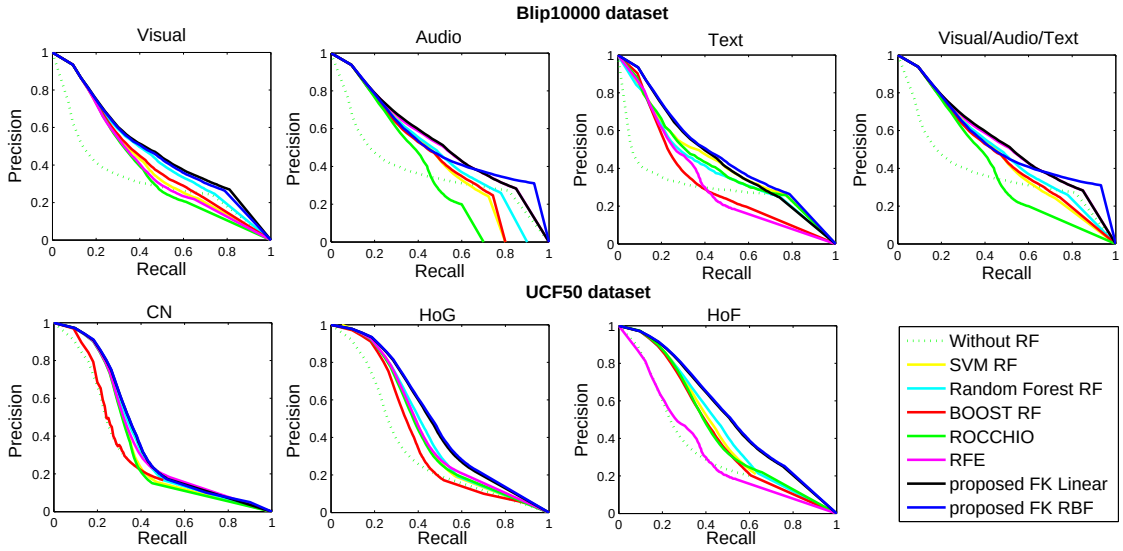

Figure 3: Precision-recall curves for different relevance feedback strategies and content descriptors (results after one relevance feedback iteration).

feedback examples for the relevance feedback algorithms to work with. Similar results are also obtained on the UCF101 dataset. The best results were obtained with the 3DHoG features (MAP 45.2%), while the lowest results were obtained with the color naming histograms (MAP 30.1%). On the ADL dataset, the body-part features are more effective, we obtain MAP 82.7%. This represents an improvement of more than 25%. The lowest performance is obtained when we use the 3DHoG features. However, improvement is still good, from 51.22% to 75.5%.

From the information source point of view, on Blip10000 dataset, audio information proves to be highly discriminative compared to visual or text information, and leads to very good retrieval ratios (see Table 2). At genre level, audio features are more accurate at retrieving music, sports, news, and commercials, as these genres have specific audio patterns. Compared to audio descriptors, visual and text descriptors used in combination are more discriminative for categories such as educational, art or web design tutorials. Finally, the best performance is achieved by using all audio-visual-textual features combined. On the UCF50 and UCF101 datasets, the highest performance is obtained using 3DHoG motion descriptors (MAP 49.4% and

Table 2: Mean Average Precision for various relevance feedback techniques and descriptors (the highest values are depicted in bold; results after one relevance feedback iteration).

Feature	Without RF	Rocchio	BOOST RF	SVM RF	RF-RF	RFE	**FK Lin.**	**FK RBF**
Blip10000 dataset								
HoG	17.1%	25.5%	26.7%	26.4%	26.8%	27.5%	29.4%	**29.5**%
structural	14.8%	21.9%	23.6%	24.6%	24.6%	23.9%	**26.2**%	23.9%
MPEG 7	25.9%	30.8%	32.5%	32.9%	36.8%	31.9%	40.5%	**40.8**%
BOVW	21.5%	25.2%	25.2%	27.6%	28.4%	28.0%	29.0%	**29.3**%
all visual	26.1%	32.9%	35.9%	36.1%	**42.2**%	32.4%	41.3%	42.2%
standard audio	29.2%	32.7%	32.8%	38.5%	40.4%	44.3%	44.8%	**46.3**%
block-based audio	21.2%	35.3%	39.8%	31.4%	33.4%	31.9%	**43.9**%	43.6%
TF-IDF on ASR	20.4%	32.5%	26.9%	34.7%	34.7%	25.8%	34.8%	**35.1**%
All Features	30.2%	37.9%	38.8%	40.9%	45.3%	44.9%	46.4%	**46.8**%
UCF50 dataset								
color naming hist.	24.22%	28.7%	30.6%	29.2%	30.8%	30.8%	31.6%	**31.7**%
HoG	26.34%	36.7%	35.8%	38.7%	39.5%	39.4%	40.4%	**41.1**%
HoF	25.24%	35.1%	36.5%	35.2%	36.3%	36.1%	44.6%	**44.8**%
BoW-3DHoG	28.79%	37.2%	36.2%	36.8%	38.7%	39.9%	48.2%	**49.4**%
CNN	27.38%	37.1%	36.3%	36.9%	38.1%	39.1%	**48.9**%	47.5%
UCF101 dataset								
color naming hist.	21.19%	26.7%	27.1%	26.2%	28.1%	26.5%	29.6%	**30.1**%
HoG	24.22%	33.7%	32.9%	32.5%	37.5%	37.4%	39.4%	**39.9**%
HoF	23.58%	30.1%	32.5%	33.7%	33.2%	34.1%	40.9%	**41.9**%
BoW-3DHoG	25.37%	35.2%	34.5%	34.7%	35.5%	36.2%	44.5%	**45.2**%
CNN	24.79%	35.1%	34.3%	34.2%	35.3%	35.8%	**43.7**%	42.5%
ADL dataset								
body-part features	57.31%	71.1%	74.1%	76.2%	78.1%	75.5%	81.1%	**82.7**%
BoW-3DHoG	51.22%	63.1%	67.3%	69.1%	70.3%	70.8%	**76.7**%	75.5%

45.2%, respectively) while the smallest increase in performance is obtained again with the color naming histograms. This is mainly due to the fact that color may not be very important for action recognition. The color features were employed with the idea to capture complementary information about the scene context, because the sports usually tend to have predominant hues. On the ADL dataset, we obtained the best results with the body-part features which are very efficient on modeling daily activities classification problems (MAP 82.7%).

Overall, for this one relevance feedback iteration, we conclude that in most of the cases, RFE and Random Forests RF provide good results, but the proposed approach is better. At the other end, the smallest increase in performance is obtained using BOOST. Also, it can be observed that the proposed FK RBF obtains slightly better performance than the approach using the linear kernel, FK linear. Therefore, FK RBF will be used in the further experiments.

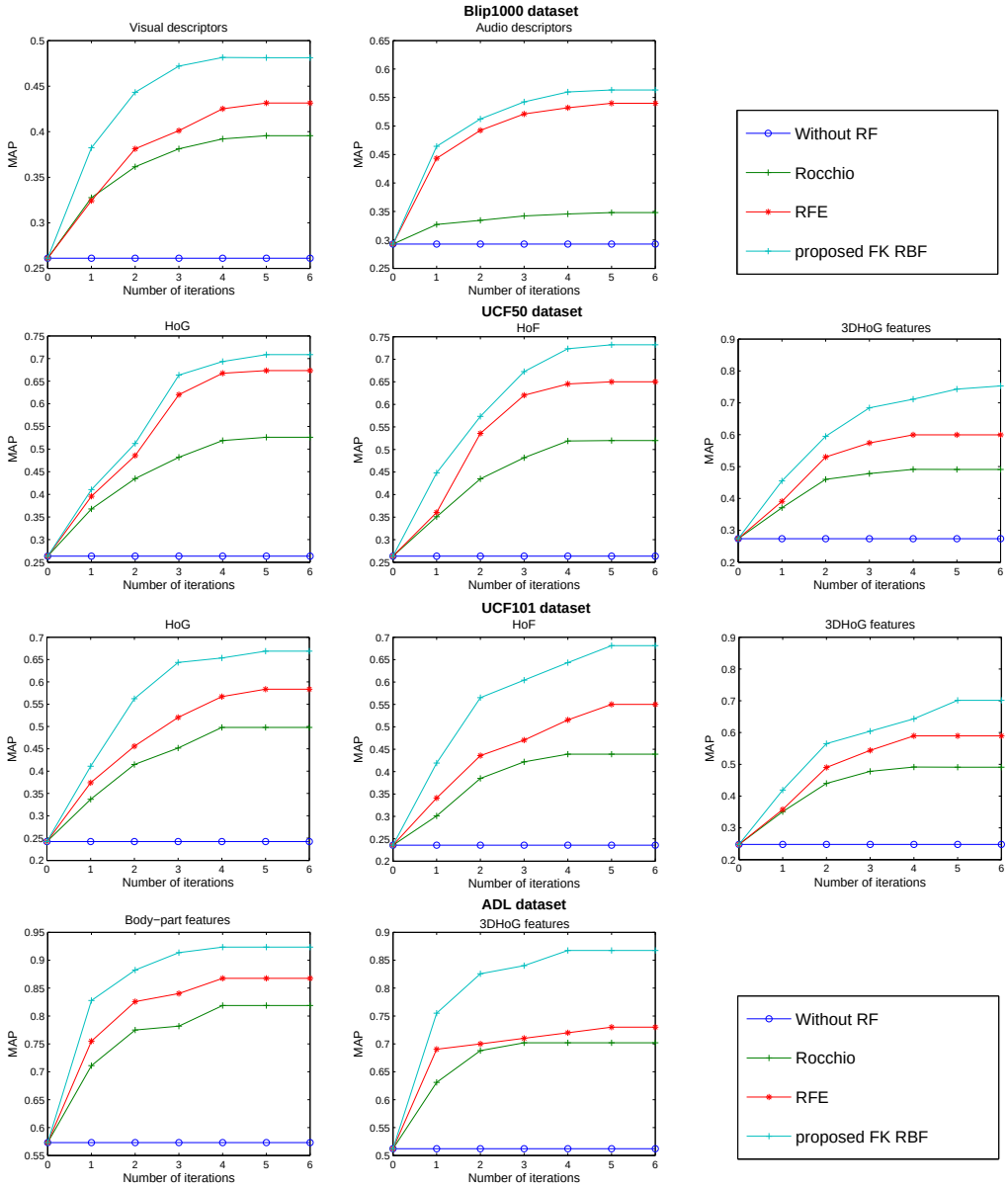

Figure 4: Mean Average Precision (MAP) for different relevance feedback iterations.

Another experiment is to assess the performance of the relevance feedback when running several feedback sessions. Figure 4 plots MAP against the number of relevance feedback iterations. For brevity reasons, we present only the best performing approaches from the previous experiment. One

may observe that the retrieval performance increases with each new feedback session. The best performance is still obtained with the proposed approach, followed by the RFE algorithm. For instance, at 5 feedback sessions, the largest increase in performance on Blip10000 database is from MAP 26.18% (without relevance feedback) to 48.12% (using visual features); on UCF50 is from 27.38% (without relevance feedback) to 75.30% (using the 3DHoG descriptor); on UCF101 is from 24.79% to 70.12% (also when we use the 3DHoG descriptor); while on ADL dataset from 57.31% to 92.34% (with body-part features). Compared to the RFE, the proposed FK RBF provides an increase of MAP up to 5% on Blip10000 database, of 7% on UCF50 and more than 8 percents on UCF101 and ADL (see the cyan lines in Figure 4).

7.3. A global GMM for the Fisher Kernel representation

In this experiment, we demonstrate that the FK representation is particularly suited for use in a relevance feedback scenario, when we use the global FK approach (when T=1). Another alternative to the proposed method is to generate a FK representation by learning a GMM on *all* the data. By testing this scenario, we want to answer to the following hypothesis: do we obtain good results because the FK representation is in general more powerful than our initial features, or are our performance improvements caused by altering the features with respect to the top n results? In the first scenario, we can just alter the features once offline. In this case the computational speed will be increased and the proposed method will be similar to the SVM relevance feedback. Furthermore, we would just prove that the FK representation is more powerful than our initial features, independent of our relevance feedback settings.

To test this, we train a GMM on all the feature vectors of the whole dataset, and represent all videos as FK representations with respect to this global mixture model. We use these features in the SVM relevance feedback and compare this with our proposed FK framework. Notice that the only differences between these two systems are on what data the GMM is learned.

The results are presented in Table 3. The first column presents the performance of FK representation by learning a GMM on *all* the data (FK RF on all data) and the second column provides the results for the proposed approach. It can be observed that by using a GMM trained on the top n results instead of using a global GMM, performance increases for the Blip10000 dataset starting from 4% for visual features and with more than 8% for audio features. Furthermore, the increase in performance is higher on the UCF50

Table 3: Comparison between FK RBF on all data and the proposed FK RF algorithm (MAP values, highest values are depicted in bold).

Feature	**FK RF for all data**	**proposed FKRF**
Blip10000 dataset		
visual descriptors	34.02%	**38.23**%
standard audio desc.	38.25%	**46.34**%
TF-IDF on ASR	32.37%	**35.14**%
UCF50 dataset		
color naming histogram	28.02%	**31.70**%
HoG	34.21%	**41.10**%
HoF	35.27%	**44.8**%

dataset. The performance increases with 3% for the color naming histogram and with 9% for HoF features. This experiment demonstrates that altering the data based on the top n videos is crucial for obtaining good performance.

7.4. Frame Aggregation with Fisher Kernel

In the following, we will show the improvements using FK on relevance feedback when we use more than one feature per video. This allows us to use multiple clusters for the GMM.

For computational reasons, we selected for the Blip10000 dataset only three types of descriptors: visual features, namely HoG and MPEG-7 related descriptors, that are more representative for the visual information, and standard audio features. On the other hand, on UCF50 / UCF101 datasets, because there are smaller sized, we used HoG, HoF, color naming histogram descriptors and 3DHoG cuboids. We also use both body-part features and 3DHoG cuboids for the ADL dataset.

We first test what is the optimal number of centroids for the GMM used by the FK. The experiments are carried out on Blip1000 and UCF50 dataset and the results are presented in Figure 5. This shows that the best results are obtained using 6 to 10 centroids. Notice the big increase when using multiple centroids: the performance increases with 2 percents in all the cases. Also, it can be viewed that the performance decreases with more than one percent when the number of centroids increases.

In Table 4 we present a comparison between the MAP values of the previous global FK approach (with RBF kernel) and the frame aggregation FK approach. SVM is presented as a baseline for comparison. The frame ag-

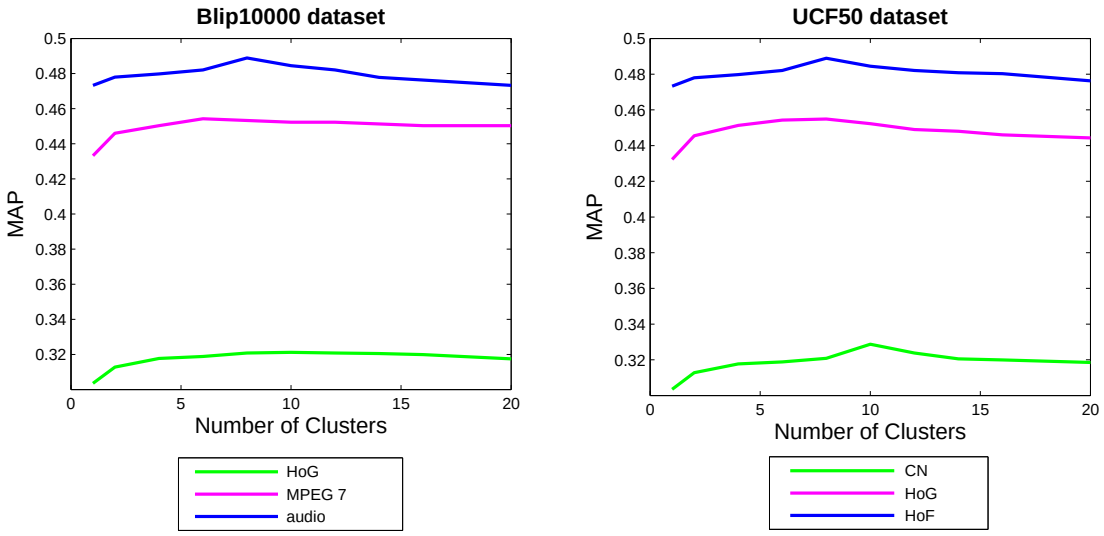

Figure 5: The influence of the number of GMM centroids on relevance feedback using frame aggregation FK RBF.

gregation FK representation for relevance feedback tends to provide better retrieval performance in all cases with more than 3% improvement.

For instance, on Blip10000 dataset, the MAP increases from 29.59% to 32.87% for HoG features and from 40.80% to 45.43% from MPEG-7 related descriptors. Also, the performance of audio features is improved from 46.30% to 49.23%. Similar improvements are obtained on the UCF50 dataset: color naming histogram yields an improvement from 31.70% to 34.35%, HoG from 41.10% to 45.49%, HoF from 44.80% to 49.82%, CN from 47.50% to 53.82% and 3DHoG from 49.40% to 54.37%. We also obtained good results on UCF101 dataset: color naming histograms are improved from 31.70% to 34.81, HoG from 41.10% to 45.49%, HoF from 44.80% to 49.82%, CNN from 47.50% to 53.82% and 3DHoG from 49.40% to 54.37%. On the ADL dataset the improvement is also higher with 7%: the body-part features are improved from 82.70% to 89.40% and the 3DHoG cuboids from 75.50% to 82.80%.

We conclude that modeling the variation in time using GMMs of 6-10 clusters yields significant improvements of performance.

Table 4: Mean Average Precision comparison between SVM relevance feedback, proposed global FK and the frame aggregation FK (highest values are depicted in bold).

Feature	SVM RF	FK RBF global	frame aggr. FK RBF
Blip10000 dataset			
HoG	26.40%	29.59%	**32.87**%
MPEG-7	32.90%	40.80%	**45.43**%
standard audio	38.5%	46.30%	**49.23**%
UCF50 dataset			
color naming hist.	29.2%	31.70%	**34.81**%
HoG	38.7%	41.10%	**45.49**%
HoF	35.2%	44.80%	**49.82**%
CNN	36.9%	47.50%	**53.82**%
3DHoG	36.8%	49.40%	**54.37**%
UCF101 dataset			
color naming hist.	26.2%	30.10%	**32.81**%
HoG	32.5%	39.90%	**44.20**%
HoF	33.7%	41.90%	**47.80**%
3DHoG	34.7%	45.20%	**52.40**%
CNN	34.2%	42.50%	**51.10**%
ADL dataset			
Body-part features	76.2%	82.70%	**89.40**%
3DHoG	69.1%	75.50%	**82.80**%

7.5. Evaluation of Computational Efficiency

The final experiment consists in assessing the computational efficiency of the proposed FK relevance feedback framework. We run a computation experiment on a regular PC and using a single core at 2.9 GHz (CPU Intel Xeon).

The computational speed is first of all dependent on the descriptor type and therefore size. First, we experimented by simulating various length descriptors, with size ranging from 10 to 1,000 dimensions. Estimated running times are presented in Table 5. Methods were implemented using C++ and Matlab environment without any hardware acceleration. We use as baseline for comparison the SVM relevance feedback and all the experiments are provided on Blip10000 dataset.

Using FK in combination with RBF SVM and global video features, we

Table 5: Computational speed of the proposed framework.

feature vector length	10	20	50	100	500	1,000
SVM	0.02s	0.021s	0.023s	0.024s	0.027s	0.033s
global FK RBF	0.31s	0.311s	0.315s	0.32s	0.35s	0.41s
frame aggr. FK RBF	4.12s	4.6s	4.9s	5.2s	5.6s	6.3s

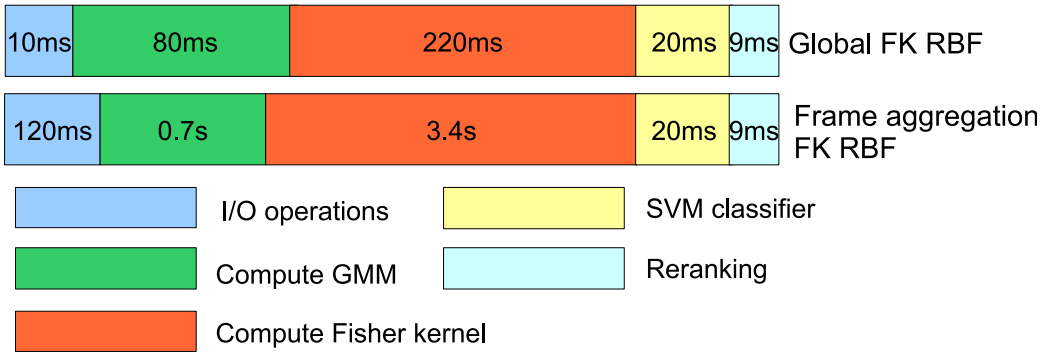

Figure 6: Total computational time, in ms, per retrieval, estimated for the proposed Fisher Kernel framework: Global FK RBF and Frame aggregation FK RBF (evaluation on Blip10000 dataset [5]).

generate a RF iteration in less than half of second. By aggregating all the frames with FK, the execution time of a relevance feedback iteration is near to 4 to 6 seconds (depending on feature size). However, even if the speed is lower, the performance of FK RF is superior. The increase of performance between SVM RF and FK RF is presented in Table 4. This performance gain presents an additional computational cost. Using the global FK approach, we obtain a better performance than classical relevance feedback methods and with a good computational trade-off.

A detailed overview of the computational time for each of the processing steps per retrieval is provided in Figure 6. We present them for both the global FK RBF and the Frame Aggregation FK RBF. For the first approach, the input/output (I/O) operations take 10 ms per retrieval. The computation of the GMM takes more than 20% of the global computation time. The Fisher Kernel calculation is very fast, namely 220 ms. Finally, the classification step takes 20 ms whereas the final re-ranking takes 9 ms. On the other hand, for the Frame aggregation FK RBF the computational time is much higher. It

takes more than 120 ms for I/O operations and less than a second for creating the GMM dictionary. The main time consuming step is the Fisher Kernel representation. The final classification and re-ranking steps are similar for both experiments.

We conclude that frame aggregation with relevance feedback represents a reasonable cost being very close to a real system scenario.

8. Conclusions

In this paper we formulated and analyzed a new approach for relevance feedback using Fisher Kernels in the context of video retrieval. Our relevance feedback consists of two steps: (1) altering the feature space by training a Gaussian Mixture Model on the top retrieved results and re-representing those features using Fisher Kernels; (2) using the user feedback to train a personalized Support Vector Machine. Additionally, the Fisher Kernel representation made it possible to capture temporal variation (but not temporal order) by using frame-based features.

Our Relevance Feedback experiments showed that our method always performs equal or better to other methods even without using temporal information: Compared to the next best method, RFE [22], we get improvements on Blip10000 between 0% and 11% MAP, averaging 5.2% MAP. For UCF50 the next best method is Random Forest RF [14] for which we get improvements of 0.9%, 1.6%, and 8.5% MAP respectively for color naming histograms, HoG, and HOF.

If we capture temporal information we get even better improvements at an acceptable computational cost. By using a GMM with only 5-10 clusters a Relevance Feedback iteration becomes 4-6 seconds, in which the user can give its feedback. Improvements are significant: On Blip10000, we get absolute MAP improvements of 3.3%, 4.6%, and 4.9% respectively for HoG, MPEG-7, and standard audio features. On UCF50 we get absolute MAP improvements of 3.1%, 4.4%, and 5.0% for respectively color naming histograms, HoG, and HoF.

Acknowledgments

Part of this work was supported under InnoRESEARCH POSDRU/ 159/1.5/S/132395 (2014-2015).

References

[1] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, 60(2), pp. 91-110, 2004.

[2] I. Laptev, M. Marszalek, C. Schmid, B. Rozenfeld, "Learning Realistic Human Actions from Movies", IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 23-28 June, Anchorage, AK, 2008.

[3] I. Mironică, B. Ionescu, J. Uijlings, N. Sebe, "Fisher Kernel based Relevance Feedback for Multimodal Video Retrieval", ACM International Conference on Multimedia Retrieval, pp. 65-72, 16-19 April, Dallas, USA, 2013.

[4] I. Mironică, J. Uijlings, N. Rostamzadeh, B. Ionescu, N. Sebe, "Time matters!: Capturing Variation in Time in Video using Fisher Kernels", ACM International Conference on Multimedia, pp. 701-704, 21-25 October, Barcelona, Catalunya, Spain, 2013.

[5] S. Schmiedeke, P. Xu, I. Ferrané, M. Eskevich, C. Kofler, M. Larson, Y. Estève, L. Lamel, G. Jones, T. Sikora, "Blip10000: A Social Video Dataset Containing SPUG Content for Tagging and Retrieval", ACM Multimedia Systems Conference, 27 February - 1 March, Oslo, Norway, 2013.

[6] K.K. Reddy, M. Shah, "Recognizing 50 Human Action Categories of Web Videos", Machine Vision and Applications, 5, pp. 971-981, 2013.

[7] F. Schroff, A. Criminisi, A. Zisserman, "Harvesting Image Databases from the Web", IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(4), pp. 754-766, 2011.

[8] H. Ma, J. Zhu, M. T. Lyu, I. King, "Bridging the Semantic Gap Between Image Contents and Tags", IEEE Transactions on Multimedia, 12(5), pp. 462-473, 2010.

[9] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, Y. Pan, "A Multimedia Retrieval Framework based on Semi-Supervised Ranking and Relevance Feedback", IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(4), pp. 723-742, 2012.

[10] S. Jones, L. Shao, "Content-based Retrieval of Human Actions from Realistic Video Databases", Information Sciences, 236, pp. 56-65, 2013.

[11] X. Y. Wang, B. B. Zhang, H. Y. Yang, "Active SVM-based Relevance Feedback using Multiple Classifiers Ensemble and Features Reweighting", Engineering Applications of Artificial Intelligence, 26(1), pp. 368-381, 2013.

[12] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A.F. Smeaton, G. Quéenot, "TRECVID 2013 — An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics", TRECVID 2013, `http://www-nlpir.nist.gov/projects/tvpubs/tv13.papers/tv13overview.pdf`, NIST, USA, 2013.

[13] C. Y. Li, C. T. Hsu, "Image Retrieval with Relevance Feedback based on Graph-Theoretic Region Correspondence Estimation", IEEE Transactions on Multimedia, 10(3), pp. 447-456, 2008.

[14] Y. Wu, A. Zhang, "Interactive Pattern Analysis for Relevance Feedback in Multimedia Information Retrieval", Multimedia Systems, 10(1), pp. 41-55, 2004.

[15] B. Ionescu, K. Seyerlehner, I. Mironică, C. Vertan, P. Lambert, "An Audio-Visual Approach to Web Video Categorization", Multimedia Tools and Applications, 70(2), pp. 1007-1032, 2014.

[16] M.M. Rahman, S.K. Antani, G.R. Thoma, "A Learning-based Similarity Fusion and Filtering Approach for Biomedical Image Retrieval using SVM Classification and Relevance Feedback", IEEE Transactions on Information Technology in Biomedicine, 15(4), pp. 640-646, 2011.

[17] J. Li, N.M. Allinson, "Relevance Feedback in Content-Based Image Retrieval: A Survey", Handbook on Neural Information Processing, Intelligent Systems Reference Library, 49, pp. 433-469, 2013.

[18] G. Cao, J. Y. Nie, J. Gao, S. Robertson, "Selecting Good Expansion Terms for Pseudo-Relevance Feedback", ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 243-250, 20-24 July, Singapore, 2008.

[19] J. Y. Kim, M. Cramer, J. Teevan, D. Lagun, "Understanding How People Interact with Web Search Results that Change in Real-Time using Implicit Feedback", ACM International Conference on Information and Knowledge Management, pp. 2321-2326, 27 October - 1 November, San Francisco, CA, USA, 2013.

[20] J. Rocchio, "Relevance Feedback in Information Retrieval", The Smart Retrieval System Experiments in Automatic Document Processing, Ed. G. Salton, Prentice Hall, Englewood Cliffs NJ, pp. 313-323, 1971.

[21] N.V. Nguyen, J.-M. Ogier, S. Tabbone, A. Boucher, "Text Retrieval Relevance Feedback Techniques for Bag-of-Words Model in CBIR", International Conference on Machine Learning and Pattern Recognition, 24 Jun - 26 Jun, Paris, France, 2009.

[22] Y. Rui, T. S. Huang, M. Ortega, M. Mehrotra, S. Beckman, "Relevance Feedback: a Power Tool for Interactive Content-based Image Retrieval", IEEE Transactions on Circuits and Video Technology, pp. 644-655, 1998.

[23] Y. Lv, C. Zhai, "Positional Relevance Model for Pseudo-Relevance Feedback", ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 579-586, 19-23 July, Geneva, Switzerland, 2010.

[24] L. Yuanhua Lv, C. Zhai, "Adaptive Relevance Feedback in Information Retrieval", ACM Conference on Information and Knowledge Management, pp. 255-264, 2-6 November, Hong Kong, China, 2009.

[25] S. Liang, Z. Sun: "Sketch Retrieval and Relevance Feedback with Biased SVM Classification", Pattern Recognition Letters, 29, pp. 1733-1741, 2008.

[26] J. Yu, Y. Lu, Y. Xu, N. Sebe, Q. Tian, "Integrating Relevance Feedback in Boosting for Content-Based Image Retrieval", IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 965-968, 15-20 April, Honolulu, Hawaii, USA, 2007.

[27] J. H. Su, W. J. Huang, P. S. Yu, V. S. Tseng, "Efficient Relevance Feedback for Content-based Image Retrieval by Mining User Navigation Patterns", IEEE Transactions on Knowledge and Data Engineering, 23(3), pp. 360-372, 2011.

[28] W. Bian, D. Tao, "Biased Discriminant Euclidean Embedding for Content-based Image Retrieval", IEEE Transactions on Image Processing, 19(2), pp. 545-554, 2010.

[29] D. Tao, X. Li, S. Maybank, "Negative Samples Analysis in Relevance Feedback", IEEE Transactions on Knowledge Data Engineering, 19(4), pp. 568-580, 2010.

[30] T. Mei, B. Yang, X. Hua, S. Li, "Contextual Video Recommendation by Multimodal Relevance and User Feedback", ACM Transactions on Information Systems, 29(2), 2011.

[31] T. Jaakkola, D. Haussler, "Exploiting Generative Models in Discriminative Classifiers", Conference on Advances in Neural Information Processing Systems II, pp. 487-493, 1998.

[32] F. Perronnin, J. Sanchez, T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification", European Conference on Computer Vision, LNCS 6314, pp. 143-156, 5-11 September, Crete, Greece, 2010.

[33] O. Aran, L. Akarun, "A Multi-Class Classification Strategy for Fisher Scores: Application to Signer Independent Sign Language Recognition", Pattern Recognition, 43(5), pp. 1776-1788, 2010.

[34] P. J. Moreno, R. Rifkin, "Using the Fisher Kernel Method for Web Audio Classification", IEEE International Conference on Acoustics, Speech and Signal Processing, 6, pp. 2417-2420, 5-9 June, Istambul, Turkey, 2000.

[35] Q. Sun, R. Li, D. Luo, W. Xihong, "Text Segmentation with LDA-based Fisher Kernel", 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers, pp.269-272, 2008.

[36] G. K. Myers, C. G. Snoek, R. Nallapati, J. van Hout, S. Pancoast, R. Nevatia, C. Sun, "Evaluating Multimedia Features and Fusion for Example-based Event Detection", Machine Vision and Applications, 25(1), pp. 17-32, 2014.

[37] Y. Chen, X.S. Zhou, T. S. Huang, "One-Class SVM for Learning in Image Retrieval", IEEE International Conference of Image Processing, pp. 34-37, 7-10 October, Thessaloniki, Greece, 2001.

[38] S. Schmiedeke, C. Kofler, I. Ferrané, "Overview of the MediaEval 2012 Tagging Task", Working Notes Proceedings of the MediaEval 2012 Workshop, CEUR-WS.org, ISSN 1613-0073, http://ceur-ws.org/Vol-927/mediaeval2012_submission_2.pdf, 4-5 October, Pisa, Italy, 2012.

[39] A. García Seco de Herrera, J. Kalpathy-Cramer, D. Demner Fushman, S. Antani, H. Müller, "Overview of the ImageCLEF 2013 medical tasks", Working Notes of CLEF 2013 (Cross Language Evaluation Forum), Valencia, Spain, 2013.

[40] MediaEval 2013 Workshop, Eds. M. Larson, X. Anguera, T. Reuter, G.J.F. Jones, B. Ionescu, M. Schedl, T. Piatrik, C. Hauff, M. Soleymani, co-located with ACM International Conference on Multimedia, CEUR-WS.org, ISSN 1613-0073, 1043, http://ceur-ws.org/Vol-1043/, 18-19 October, Barcelona, Catalunya, Spain, 2013.

[41] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results", http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[42] P. Dollár, V. Rabaud, G. Cottrell, S. Belongie, "Behavior Recognition via Sparse Spatio-Temporal Features", IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, pp. 65-72, 15-16 October, Beijing, China, 2005.

[43] O. Kliper-Gross, Y. Gurovich, T. Hassner, L. Wolf, "Motion Interchange Patterns for Action Recognition in Unconstrained Videos", European Conference on Computer Vision, pp. 256-269, 7-13 October, Firenze, Italy, 2012.

[44] S.H. Cha, "Comprehensive Survey on Distance/Similarity Measures Between Probability Density Functions", International Journal of Mathematical Models and Methods in Applied Sciences, pp. 300-307, 2007.

[45] Y. Rubner, C. Tomasi, L. J. Guibas, "A Metric for Distributions with Applications to Image Databases", International Conference on Computer Vision, pp. 59-66, 4-7 January, Bombay, India, 1998.

[46] E. Deza, M.M. Deza, "Dictionary of Distances", Elsevier Science, 1st edition, ISBN: 978-0-444-52087-6, 2006.

[47] M. Hatzigiorgaki, A. N. Skodras, "Compressed Domain Image Retrieval: A Comparative Study of Similarity Metrics", SPIE Visual Communications and Image Processing, 5150, pp. 439-448, 2003.

[48] K. Seyerlehner, M. Schedl, T. Pohle, P. Knees, "Using Block-Level Features for Genre Classification, Tag Classification and Music Similarity Estimation", 6th Annual Music Information Retrieval Evaluation eXchange, 9-13 August, Utrecht, Netherlands, 2010.

[49] B.Mathieu, S.Essid, T.Fillon, J.Prado, G.Richard, "YAAFE, an Easy to Use and Efficient Audio Feature Extraction Software", International Society for Music Information Retrieval Conference, pp. 441-446, 9-13 August, Utrecht, Netherlands, 2010.

[50] T. Sikora, "The MPEG-7 Visual Standard for Content Description - An Overview", IEEE Transactions on Circuits and Systems for Video Technology, 11(6), pp. 696-702, 2001.

[51] O. Ludwig, D. Delgado, V. Goncalves, U. Nunes, "Trainable Classifier-Fusion Schemes: An Application To Pedestrian Detection", IEEE International Conference On Intelligent Transportation Systems, pp. 1-6, 4-7 October, St. Louis, MO, USA, 2009.

[52] C. Rasche, "An Approach to the Parameterization of Structure for Fast Categorization", International Journal of Computer Vision, 87(3), pp. 337-356, 2010.

[53] S. Nowak, M. Huiskes, "New Strategies for Image Annotation: Overview of the Photo Annotation Task at ImageClef 2010", Working Notes of CLEF 2010.

[54] J.R.R. Uijlings, A.W.M. Smeulders, R.J.H. Scha, "Real-Time Visual Concept Classification", IEEE Transactions on Multimedia, 12(7), pp. 665-681, 2010.

[55] S. Lazebnik, C. Schmid, J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories", Computer Vision and Pattern Recognition, pp. 2169 - 2178, 17-22 June, New York, NY, USA, 2006.

[56] J. Van de Weijer, C. Schmid, J. Verbeek, D. Larlus, "Learning Color Names for Real-World Applications", IEEE Transactions on Image Processing, 18(7), pp. 1512-1523, 2009.

[57] B. Lucas, T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", International Joint Conference on Artificial intelligence, pp. 674-679, Vancouver, Canada, 1981.

[58] L. Lamel, J.-L. Gauvain, "Speech Processing for Audio Indexing", Advances in Natural Language Processing, LNCS, 5221, pp 4-15, 2008.

[59] I. Mironică, B. Ionescu, P. Knees, P. Lambert, "An In-Depth Evaluation of Multimodal Video Genre Categorization", IEEE International Workshop on Content-Based Multimedia Indexing, pp. 11-16, 17-19 June, Veszprém, Hungary, 2013.

[60] C.G.M. Snoek, K.E.A. van de Sande, O. de Rooij, B. Huurnink, J.C. van Gemert, J.R.R. Uijlings, J. He, X. Li, I. Everts, V. Nedovic, M. van Liempt, R. van Balen, F. Yan, M.A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, A.W.M. Smeulders, D.C. Koelma, "The MediaMill TRECVID 2008 Semantic Video Search Engine", 6th TRECVID Workshop, Gaithersburg, USA, 2008.

[61] S. Khurram, A. R. Zamir, and M. Shah: "Ucf101: A dataset of 101 human actions classes from videos in the wild." CoRR, abs/1212.0402, 2012.

[62] R. Messing, C. Pal, and H. Kautz. Activity recognition using the velocity histories of tracked keypoints. In ICCV, 2009.

[63] A. Vedaldi and K. Lenc, "MatConvNet - Convolutional Neural Networks for MATLAB", In Proceedings of the arXiv:1412.4564, 2014.

[64] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks", In Conference on Neural Information Processing Systems (NIPS), 2012.

[65] J. Uijlings, I. Duta, E. Sangineto, and N. Sebe, "Video classification with densely extracted HoG/HoF/Mbh features: an evaluation of the accuracy/computational efficiency trade-off", International Journal of Multimedia Information Retrieval, pages 112, 2014.

[66] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele, "A database for fine grained activity detection of cooking activities", In International Conference of Computer Vision and Pattern Recognition, CVPR, 2012.

[67] N. Rostamzadeh, G. Zen, I. Mironica, J. Uijlings, N. Sebe, "Daily Living Activities Recognition via Efficient High and Low Level Cues Combination and Fisher Kernel Representation", IEEE International Conference on Image Analysis and Processing, ICIAP, 2013.

[68] Z. Xu, Y. Yang, A. G. Hauptmann, "A discriminative CNN video representation for event detection.", Computer Vision and Pattern Recognition (CVPR), 2015.

[69] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice", CoRR , abs/1405.4506, 2014.

[70] Y. Yang, Z. Ma, Z. Xu, S. Yan, A. G. Hauptmann, "How related exemplars help complex event detection in web videos?", International Conference on Computer Vision (ICCV), pp. 2104-2111, 2013.

[71] Zhongwen Xu, Yi Yang, Alexander G. Hauptmann, "A Discriminative CNN Video Representation for Event Detection", International Conference on Computer Vision (ICCV), 2015.

[72] K. Yanai, "Automatic extraction of relevant video shots of specific actions exploiting Web data", Computer Vision and Image Understanding (CVIU), vol. 118, pp. 2-15, 2014.

[73] L. Shao, S. Jones, X. Li, "Efficient search and localization of human actions in video databases", IEEE Transactions on Circuits and Systems for Video Technology, vol. 24(3), pp. 504-512, 2014.

[74] L. Shao, X. Zhen, D. Tao, X. Li, "Spatio-temporal Laplacian pyramid coding for action recognition", IEEE Transactions on Cybernetics, vol. 44(6), pp. 817-827, 2014.

[75] L. Liu, L. Shao, F. Zheng, X. Li, "Realistic action recognition via sparsely-constructed Gaussian processes", Pattern Recognition, vol. 47(12), pp. 3819-3827, 2014.

[76] I. Mironică, I. Duta, B. Ionescu, N. Sebe, "A Modified Vector of Locally Aggregated Descriptors Approach for Fast Video Classification", Multimedia Tools and Applications (MTAP), 2015.