

Video shot retrieval using a kernel derived from a continuous HMM

ABSTRACT

In this paper, we propose a discriminative approach for retrieval of video shots characterized by a sequential structure. The task of retrieving shots similar in content to a few positive example shots is more close to a binary classification problem. Hence, this task can be solved by a discriminative learning approach. For a content-based retrieval task the twin characteristics of rare positive example occurrence and a sequential structure in the positive examples make it attractive for us to use a learning approach based on a generative model like HMM. To make use of the positive aspects of both discriminative and generative models for our task, we derive Fisher and Modified score kernels for a continuous HMM and incorporate them into SVMs. A video shot is ranked based on its proximity to the positive hypothesis of SVM classifier. We evaluate the performance of the derived kernels by retrieving video shots of airplane takeoff. The retrieval performance using the derived kernels is found to be much better compared to linear and RBF kernels.

1. INTRODUCTION

There has been a large increase in the amount of digital video data over the past decade. As the increasing amount of multimedia information is driving the demand for content-based access to video data, new challenges have been created.

It is difficult to have a structured representation of large quantity of video data, but without a structured representation, any type of search over the video collection cannot succeed. In the ideal case, we would understand what the image content is, and translate it into words for search. In such a situation it is convenient to specify example data and retrieve other video information similar in content. Hence, content-based retrieval of video data is an ideal way to access the desired video information.¹ However, full, in-depth video understanding and effective search has proven to be an elusive goal.³ The video analysis community has long struggled to bridge the gap from successful, low-level feature analysis (color histograms, texture, shape) to semantic content description of video.

Video classification is arguably the first step toward video content understanding, and has been an active sub-field of video analysis research.

The National Institute of Standards and Technology has sponsored the Text Retrieval Conference (TREC) since 1992 as a means of encouraging research in information retrieval from large test collections. In 2001, the TREC Video Track began with the goal to promote progress in content-based retrieval from digital video via open, metrics-based evaluation. The corpora have ranged from documentaries to advertising films to broadcast news, with international participation growing from 12 to 24 companies and academic institutions from 2001 to 2003, now referred to as "TRECVID".² A number of tasks ranging from shot detection to story segmentation to semantic feature extraction to information retrieval are defined in the TRECVID forum. In this paper, we will focus on video classification into semantic feature categories, in particular when dynamic motion is a key element of the semantic class.

Examples of high-level semantic features are topics such as: group of people, Indoor/outdoor, airplane takeoff, Beach, Train, Car/truck/bus, sporting events. In the semantic feature extraction task the goal is to retrieve all the shots which contain specific topics such as those mentioned above. One class of topics involves dynamic aspects that cannot be derived from an analysis of a single keyframe. Examples of such dynamic or motion semantic topics are e.g. "rocket launches", "a soccer goal being scored", "people eating" or "a person walking", etc.

In this paper we concentrate on one particular such dynamic topic, "airplane takeoff", which was one of the high-level semantic features in the NIST TRECVID evaluation. For this dynamic, high-level semantic feature, the system has to retrieve all the shots containing video of an airplane taking off, moving away from the viewer.

Discriminative classifiers like SVM⁵ have been successfully used for image retrieval⁷ by learning a classifier on a labeled dataset. Our classification task is discriminative in nature because we are interested in retrieving only one class or topic, so we need to train a classifier which is suited to discriminate between the airplane

takeoff(positive class) and rest of data(negative class). In⁸ the properties of both generative and discriminative classifiers are combined by using a Fisher score kernel. A similar approach based on score-spaces uses SVMs for speech recognition.¹² We derive a Fisher score kernel for a continuous HMM and incorporate it into SVM classifier and rank the test set shots based on the distance from the decision boundary of Positive and Negative Hypothesis. Proximity to the Positive Hypothesis gives a higher rank to a shot. A new kernel called Modified score kernel is introduced. The aim of the Modified score kernel is to increase the rank of the positive examples which are at lower ranks for a Fisher kernel. Similar to the Fisher kernel we derive Modified score kernel for a continuous HMM and incorporate it into SVM classifier and rank the test set shots based on the distance from the decision boundary of Positive and Negative Hypothesis. The two important observations motivating our approach are:

1. The positive example shots have a rare occurrence in the training set. They are less than 0.1 percent of the total number of training set shots .
2. There is a strong sequential characteristic in the airplane takeoff topic shots which is better captured by a generative model such as HMM.

The sequential structure we refer to is the airplane initially moving on the runway and later rising into the sky. It has been shown that a simple generative classifier like naive-bayes performs better than its discriminative counterpart(logistic regression) when the amount of labeled data is small.⁴ These two observations imply that in our case a generative model is ideal for learning the characteristics of positive examples. On the other hand as stated previously discriminative classifiers are ideal for retrieval. This is exactly where the idea of deriving kernels from a generative model and incorporating them into SVMs fits in. The intuition behind using HMM as a generative model is that 2 states of a HMM capture the sequential structure of airplane takeoff. The first state capturing the airplane movement on a runway. While the second state capturing the moving away or moving towards sky path of airplane. We evaluate our approach on a set of videos taken from the TRECVID 2003 dataset and show that the 2 kernels derived from a generative model perform much better than the standard linear and RBF kernels. The rest of the paper is organized as follows. In Section 2 we give a general definition for Fisher score kernel and Modified score kernel. Section 3 explains the principle of SVM, in Section 4 partial derivatives w.r.t to continuous HMM parameters are given. Section 5 gives the scoring technique for different kernels, Section 6 describes the features used to represent a shot and Section 7 explains the dataset and Section 8 the procedure for Learning classifiers. In Section 9 we give the results and conclude the paper in Section 10.

2. KERNELS DERIVED FROM GENERATIVE MODELS

In general when we refer to an example we refer to its feature representation.

2.1. Fisher score kernel

Let us denote by \mathcal{X} , the space of all examples. Let $X \in \mathcal{X}$. The Fisher score⁸ for X w.r.t the parameters θ of a generative model with probability $P(X|\theta)$ is:

$$U_X = \nabla_{\theta} \log P(X|\theta)$$

The Fisher information matrix is denoted as I_F , where $I_F = E_{X|\theta}\{U_X U_X^T\}$. The expectation is taken over $P(X|\theta)$.

The Fisher kernel between 2 examples X_i and X_j is $K(X_i, X_j) = U_{X_i}^T I_F^{-1} U_{X_j}$. This kernel is stated to be a valid kernel function because of the positive definite property of I_F .⁸

It is known that asymptotically I_F is immaterial⁸ and hence, $K_U(x_i, x_j) \propto U_{X_i}^T U_{X_j}$ can be used as a kernel instead of the original Fisher kernel.

2.2. Modified score kernel

Consider an example $X \in \mathcal{X}$. We define a *Modified score vector* for X as

$$L_X = \begin{bmatrix} 1 \\ \nabla_{\theta_1} \end{bmatrix} \log \frac{P(X|\theta_1)}{P(X|\theta_2)} \quad (1)$$

In the above equation θ_1 indicates the parameters of the positive Hypothesis of the generative model and θ_2 indicates the parameters of the negative Hypothesis of the generative model. We are hoping to get a better discriminative kernel by utilizing θ_2 through $\log \frac{P(X|\theta_1)}{P(X|\theta_2)}$.

Let $\mu_{L_X} = E_{X|\theta_1} \{L_X\}$. Then, we define the *Modified information matrix* as

$$I_M = E_{X|\theta_1} \{(L_X - \mu_{L_X})(L_X - \mu_{L_X})^T\} \quad (2)$$

where the expectation is taken over $P(X|\theta_1)$

Further assume that $X_i, X_j \in \mathcal{X}$. We then define the *Modified score kernel* between the examples X_i and X_j as

$$\tilde{K}(X_i, X_j) = L_{X_i}^T I_M^{-1} L_{X_j} \quad (3)$$

Theorem 1.

Let \mathcal{X} be the space of all examples. Let X, X_i , and X_j be elements of \mathcal{X} . Further assume that the *Modified Score Vector*, L_X , *Modified Information Matrix*, I_M , and *Modified Score Kernel*, \tilde{K} , be as defined in equations (1), (2) and (3). Then I_M is a positive definite matrix and \tilde{K} is a valid kernel.

Proof. From equation (2), it is clear that I_M is a covariance matrix. Hence, I_M is positive definite. For \tilde{K} to be a valid kernel, it is sufficient that I_M is positive definite,^{8,9} implying that \tilde{K} is a valid kernel. \square

As in the case of Fisher kernel, asymptotically I_M is immaterial i.e. $K_L(X_i, X_j) \propto L_{X_i}^T L_{X_j}$.

3. PRINCIPLE OF SVM

SVM is a classifier⁵ which aims to find a decision surface that “best” separates the data points into two classes based on the Structural Risk Minimization Principle. The decision function is of the form

$$y = \text{sign} \left(\sum_{i=1}^N y_i \alpha_i K(X, X_i) + b \right)$$

where X is the d-dimensional vector of a test example, $y_i \in \{-1, 1\}$ is the class label of the i^{th} example, $y_i = 1$ for a positive example and $y_i = -1$ for a negative example. X_i is the vector for the i^{th} training example, N is the number of training examples, $K(X, X_i)$ is a kernel function between X and X_i , $\alpha = \{\alpha_1, \dots, \alpha_N\}$ and b are the parameters of the model. These α_i can be learned by solving following quadratic programming (QP) problem,

$$\max W(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(X, X_i)$$

subject to $\sum_{i=1}^N \alpha_i y_i = 0$ and $0 \leq \alpha_i \leq C, \forall i$. C is the regularization constant.

4. PARTIAL DERIVATIVES FOR CONTINUOUS HMM PARAMETERS

We denote the parameters of HMM by $\lambda = (A, B, \pi)$. $A = \{a_{ij}\}$, $\pi = \{\pi_i\}$, $B = \{c_{jk}, \mu_{jk}, \Sigma_{jk}\}$. Let q_t denote the state at time t , and $\pi_i = P[q_1 = i]$. Let Q be the number of states of HMM.

$$a_{ij} = P[q_{t+1} = j | q_t = i], \quad 1 \leq i, j \leq Q.$$

a_{ij} is the probability of transition from state i at time t to state j at time $t+1$. In our case we model the observation vectors using a continuous probability distribution. $b_j(O_t)$ is the probability density function in state j associated with the observation vector O_t at time t . In our case $b_j(O_t)$ is a mixture of Gaussians.

$$b_j(O_t) = \sum_{k=1}^K c_{jk} \mathcal{N}(O_t, \mu_{jk}, \Sigma_{jk}), \quad 1 \leq j \leq Q$$

In equation (6), c_{jk} is the mixture coefficient of the k^{th} Gaussian mixture component in state j . The coefficients satisfy the constraints: $\sum_{k=1}^K c_{jk} = 1$ and $c_{jk} \geq 0$, $1 \leq k \leq K$. μ_{jk} is the mean vector of the k^{th} Gaussian mixture component in state j . Σ_{jk} is the covariance matrix of the k^{th} Gaussian mixture component in state j . The parameters λ of the HMM are estimated from the training set data using the Baum-Welch re-estimation procedure, outlined in.⁶ $O = O_1, \dots, O_T$ is an observation vector sequence of length T .

We define $\xi_t(i, j)$ as the probability of being in state i at time t and state j at time $t+1$ given the model λ and the observation sequence O . $\gamma_t(j, k)$ is the probability of being in state j at time t with the k^{th} Gaussian mixture component accounting for O_t .

$$\xi_t(i, j) = P(q_t = i, q_{t+1} = j | O, \lambda)$$

$$\gamma_t(j, k) = P(q_t = i, M_{jt} = k | O, \lambda)$$

M_{jt} is a random variable indicating the mixture component at time t , while in state j . The partial derivatives w.r.t the parameters $\{a_{ij}, c_{jk}, \mu_{jk}, \Sigma_{jk}\}$ are given in equations (4) to (7).

Let \otimes denotes the kronecker product. If F is a matrix of size $M \times N$, $\text{vec}(F) = [F_{11}, F_{12}, \dots, F_{M1}, \dots, F_{MN}]^T$.

$$\nabla_{a_{ij}} \log P(O | \lambda) = \sum_{t=1}^{T-1} \frac{\xi_t(i, j)}{a_{ij}} \quad (4)$$

$$\nabla_{c_{jk}} \log P(O | \lambda) = \sum_{t=1}^T \frac{\gamma_t(j, k)}{c_{jk}} \quad (5)$$

$$\nabla_{\mu_{jk}} \log P(O | \lambda) = \sum_{t=1}^T \gamma_t(j, k) \left[(O_t - \mu_{jk})^T \Sigma_{jk}^{-1} \right]^T \quad (6)$$

$$\nabla_{\text{vec}(\Sigma_{jk})} \log P(O | \lambda) = \sum_{t=1}^T \gamma_t(j, k) 0.5 \left[-[\text{vec}(\Sigma_{jk}^{-1})]^T + [(O_t - \mu_{jk})^T \Sigma_{jk}^{-1} \otimes (O_t - \mu_{jk})^T \Sigma_{jk}^{-1}]^T \right]^T \quad (7)$$

5. SCORING TECHNIQUES FOR RANKING SHOTS

We take all the positive example shots in the training set and estimate the parameters λ_1 of a continuous HMM using the Baum-Welch re-estimation technique. The partial derivatives are taken w.r.t the parameters $\widehat{\lambda}_1 = \{a_{ij}, c_{jk}, \mu_{jk}\}$ while calculating the Fisher score and the modified score vector. The covariance matrix parameter is not used because we feel the parameters in $\widehat{\lambda}_1$ have sufficient discriminative power and also to reduce the computational cost. The SVM parameter $b=1$ in our evaluation. The test set shots are ranked on the basis of a score value. A shot with a higher score gets a higher rank. We now explain the score calculation for different kernels used in our evaluation.

5.1. Fisher kernel and Modified score kernel

In this case we use the entire observation sequence to represent a shot. The score of a shot with observation sequence O is

$$score(O) = \sum_{i=1}^N y_i \alpha_i K(O, O^i) + b \quad (8)$$

Where O^i is the i^{th} observation sequence in the training set, $y_i \in \{-1, 1\}$ are labels associated with the training set examples, α_i are known by solving the optimization problem described in Section 3 and N is the number of examples in the training set. The Fisher score U_O of an observation vector sequence $O = O_1 \dots O_T$ is.

$$U_O = \nabla_{\widehat{\lambda}_1} \log P(O|\lambda_1). \quad (9)$$

We use the asymptotic approximation while obtaining the Fisher kernel. This kernel is used in equation 8, to obtain a score.

The modified score vector L_O of an observation vector sequence $O = O_1 \dots O_T$ is.

$$L_O = \left[\begin{array}{c} 1 \\ \nabla_{\widehat{\lambda}_1} \end{array} \right] \log \frac{P(O|\lambda_1)}{P(O|\lambda_2)} \quad (10)$$

Where λ_2 denotes the parameters of the continuous HMM estimated from the Negative example shots in the training set. We use the asymptotic approximation while obtaining the Modified score kernel. This kernel is used in equation 8, to obtain a score.

5.2. Linear and RBF kernel

In this case each shot is represented with a single vector X . The score of the shot X is

$$score(X) = \sum_{i=1}^N y_i \alpha_i K(X, X_i) + b \quad (11)$$

X_i is the vector for the i^{th} training example, N is the number of training examples. A linear kernel between X_i and X_j is $K_l(X_i, X_j) = X_i^T X_j$. The RBF kernel between X_i and X_j is $K_r(X_i, X_j) = \exp(-\gamma \|X_i - X_j\|^2)$. We set $\gamma = 0.01$.

6. FEATURES

In this section we discuss about the features used to represent a shot. The 2 features used to represent a shot are Color feature and Motion feature. These 2 features are used independent of each other i.e. we have an only Motion representation of a shot and a only color representation of a shot.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Figure 1. A video frame is divided into 25 regions

6.1. Motion feature for HMM based kernels

To capture the sequential characteristics of an airplane takeoff shot we use optical flow of the frames in a shot. We divide each video frame into a 5x5 image tessellation as shown in Figure 1. For each of these 25 regions we calculate the mean optical flow, in x,y directions. To avoid redundancy the optical flow feature is calculated for every third video frame, this sequence is represented as $Op_1.....Op_T$. We calculate an 8 directional motion histogram of the original optical flow feature and append it to the original feature to form a 58 dimensional feature vector. A sequence of these 58 dimensional feature vectors becomes the observation vector sequence $mO = mO_1.....mO_T$.

6.2. Motion feature for Linear and RBF SVM kernels

In this case we need a single feature vector to represent the shot. As described for the previous motion feature, we first form a sequence of optical flow features $Op_1.....Op_T$. We get a single motion feature mX by taking mean of $Op_1.....Op_T$ and appending to it an 8 directional motion histogram of the mean vector.

6.3. Color feature for HMM based kernels

The Munsell(H,V,C)¹⁰ color space histogram is used as a visual feature for the airplane takeoff shot. We use 16 bins histogram for each component namely H, V, C. To reduce the dimensionality of this feature space, We only use the mean H, V, C values for each region in the 5x5 image tessellation. Hence we have 3 values for each of the 25 regions or a 75 dimensional feature vector for the entire image. To keep the length of color feature observation vector sequence same as motion feature observation vector sequence for a shot, we first extract 3 frames at equal intervals in the shot, find their color feature vectors and upsample them so that their total number is same as the total number of motion feature vectors. The color feature observation vector sequence is represented as $cO = cO_1.....cO_T$.

6.4. Color feature for Linear and RBF SVM kernels

In this case we need a single feature vector to represent the shot. We select the middle frame in the shot and using the procedure described in section 6.3 , its 75 dimensional color feature cX is obtained.

7. DATASET DETAILS

In our experiments we use 220 videos in the 2003 TRECVID dataset. Each video is approximately of 28 minutes duration. These 220 videos are of Broadcast news type. Out of the 220, we use 132 videos as training set and 88 videos as test set. The shot boundaries are provided by TRECVID National Institute of standards and technology. The number of positive examples in the training set is less than 0.1 percent of total number training set shots. We use 27 positive examples in the training set. The duration of all the 27 positive example shots put together is less than 200 seconds. Out of the entire training set, we select a much smaller set of 4626 shots as negative example training set. Though 4626 is much smaller than the total number of negative example shots in the 132 videos, we feel it is sufficient for training purpose when compared to the 27 positive examples.

The 88 test videos have a total of 61894 shots. The number of airplane takeoff shots or the positive examples is 0.15% of the total number of shots in the test videos. Computationally it is very expensive to calculate the optical flow values for video frames in all test set shots. To reduce the computational cost involved in calculating the features for the test set shots we introduce a filtering stage. The aim of the filtering stage is to eliminate all the shots in the test which have a frontal face in them or have a duration less than 1 second. Using a low threshold value for a face detection algorithm¹¹ we eliminate all the shots in the test set below that threshold.

Table 1. Comparison of different kernels for motion feature using Precision at multiple ranks

Rank	Precision			
	Fisher	Modified	linear	RBF
10	0.2	0.2	0	0.1
30	0.167	0.1	0	0.033
60	0.1	0.1	0	0.017
100	0.07	0.1	0	0.02

Table 2. Comparison of different kernels for color feature using Precision at multiple ranks

Rank	Precision			
	Fisher	Modified	linear	RBF
10	0.2	0.2	0.1	0.1
30	0.133	0.133	0.067	0.033
60	0.117	0.083	0.067	0.067
100	0.07	0.07	0.05	0.04

8. LEARNING CLASSIFIERS

8.1. Positive Hypothesis λ_1 HMMs

We calculate the motion feature observation vector sequence $mO = mO_1 \dots mO_T$ for each of the 27 shots in positive example training set and use the 27 observation sequences together to train the motion feature Positive Hypothesis HMM. This HMM has 2 states and a mixture model probability density function. The mixture model has 4 Gaussian components. The parameters of this HMM are used to derive the Fisher and modified score kernels described previously, using mO as the observation sequence.

We calculate the color feature observation vector sequence $cO = cO_1 \dots cO_T$ for each of the 27 shots in positive example training set and use the 27 observation sequences together to train the color feature Positive Hypothesis HMM. This HMM has 2 states and a mixture model probability density function. The mixture model has 4 Gaussian components. The parameters of this HMM are used to derive the Fisher and modified score kernels described previously, using cO as the observation sequence.

8.2. Negative Hypothesis λ_2 HMMs

We calculate the motion feature observation vector sequence $mO = mO_1 \dots mO_T$ for each of the 4626 shots in the negative example training set and use the 4626 observation sequences together to train a motion feature Negative Hypothesis HMM. This HMM has 2 states and a mixture model probability density function. The mixture model has 2 Gaussian components. The parameters of this HMM are used to derive the Fisher and Modified score kernels described previously, using mO as the observation sequence.

We calculate the color feature observation vector sequence $cO = cO_1 \dots cO_T$ for each of the 4626 shots in the negative example training set and use the 4626 observation sequences together to train a color feature Negative Hypothesis HMM. This HMM has 2 states and a mixture model probability density function. The mixture model has 2 Gaussian components. The parameters of this HMM are used to derive the Fisher and modified score kernels described previously, using cO as the observation sequence.

8.3. Details on Training SVMs

The motion feature mX described in section 6.2 is calculated for the 27 positive example shots and 4626 negative example shots. Motion feature SVM is trained for linear and RBF kernels. The color feature cX described in section 6.4 for a linear and RBF kernels is calculated for the 27 positive example shots and 4626 negative example shots. Color feature SVM is trained for linear and RBF kernels. While training SVMs the parameter $C = 8000$.

Table 3. A comparison of Average precision for different kernels using motion feature

Average Precision			
Fisher	Modified	linear	RBF
0.041	0.05	0.003	0.012

9. RESULTS

Figure 3 shows three video frames extracted towards the start, middle and end of a video shot. This video shot is a common to both the Fisher kernel and the Modified score kernel in the top ranks on using motion feature. The sequential structure in the shot is evident. In the first video frame(start) the airplane is still on the runway which can be thought of as the first state of the HMM. The second video frame(middle) shows the airplane slightly above the ground and moving in an inclination. While the third frame(end) shows the airplane clearly above the ground and moving into the sky. The second and third video frames can be thought to be in the second state of the HMM. We use the precision measure at different ranks to compare the performance of different kernels. Let $|r_i^+|$ denote the number of positive examples upto rank r_i . The precision at rank r_i is

$$\text{precision}(r_i) = \frac{|r_i^+|}{r_i}$$

In Table 1 we use the motion feature to compare the performance of different kernels. The linear kernel has a precision of 0 at all ranks up to 100, which means that no airplane takeoff shots are present in the top 100 shots. The Fisher kernel has a precision of 0.2 at rank 10, which means that 2 shots among the top 10 shots are airplane takeoff shots. The precision at 100 is 0.07 which means that 7 in the top 100 are airplane takeoff shots. Hence this indicates a remarkably better performance compared to the linear kernel. The Modified score kernel has a precision of 0.2 at rank 10, which means that 2 shots among the top 10 shots are airplane takeoff shots. The precision at the other 3 ranks is almost the same as the Fisher kernel. The RBF kernel has 1 airplane takeoff shot in the top 10 and only 2 airplane takeoff shots in the top 100. It is obvious that the performance of the Fisher and Modified score kernels is much better than linear and RBF kernels in terms of placing airplane takeoff shots in the top 100 ranks.

Figure 2 shows three video frames extracted towards the start, middle and end of a video shot. This video shot is a common to both the Fisher kernel and the Modified score kernel in the top ranks on using color feature. This shot also displays a sequential structure. In the first video frame(start) the airplane, runway, and the blue sky are prominent. In the second video frame(middle) the airplane becomes more prominent, while the runway and blue sky are same as before. These 2 video frames can be thought of being in the first state of HMM. In the third video frame(end) the sky becomes more bright the runway remains same as before, while the airplane has become less prominent as it is moving away. This video frame can be thought of being in second state of HMM. In Table 2 we use the color feature to compare the performance of different kernels. The linear and RBF kernels have a precision of 0.1 at rank 10, which means there is 1 airplane takeoff shot in the top 10. The Fisher kernel and the Modified score kernel have a precision of 0.2 at rank 10. It is to be noted for all the ranks in Table 2 the precision values of a Fisher kernel and Modified score kernel are greater than the precision of linear and RBF kernels. The precision values of the Fisher kernel and the Modified score kernel are equal at most ranks. The Fisher and Modified score kernels have a much better performance compared to linear and RBF kernels in terms of returning airplane takeoff shots among the top 100 ranks.

Table 1 and Table 2 gave a comparison of the performance within the top 100 ranked shots. We now move to a more compact comparison. In TRECVID evaluation the value of Average Precision¹³ has been used as a measure of retrieval performance. For a rank list of shots S , we denote the total number of shots as $|S|$ and the number of positive examples as $|S^+|$.

$$\text{Average precision} = \frac{\sum_{i=1}^{|S|} \text{precision}(r_i)}{|S^+|} \quad (12)$$



Figure 2. A shot retrieved using color feature is represented using 3 Frames extracted. It can be seen that the color feature captures the sequential nature of airplane takeoff.

Table 3 shows that in the case of the motion feature, the Average precision for the linear and RBF kernels are 0.003 and 0.012. The Average precision for the motion feature using the Fisher score kernel and the Modified score kernel are 0.041 and 0.05 respectively. It is to be noted that there is a 10 fold improvement in the performance of the Modified score kernel when compared to the linear kernel and a 4 fold improvement when compared to the RBF kernel.

Table 4 shows that in the case of the color feature, the average precision of the linear kernel and RBF kernel are 0.017 and 0.02. The average precision using the Fisher score kernel and the Modified score kernel are 0.035 and 0.042 respectively. It is to be noted that there is a 2 fold improvement in the performance of the Modified kernel when compared to the linear kernel and RBF kernel.



Figure 3. A shot retrieved using motion feature is represented using 3 Frames . It can be seen that the motion feature captures the sequential nature of airplane takeoff.

9.1. Performance based Kernel and feature combination

It is observed that the improvement in Average precision when using a Fisher kernel or a Modified score kernel is higher in the case of the motion feature as compared to the color feature. One possible reason is that the motion feature models the sequential structure of airplane takeoff better than color.

The Average Precision of the Modified score kernel is higher than that of the Fisher score kernel for both motion and color features. This is due to the fact that the $\log \frac{P(X|\theta_1)}{P(X|\theta_2)}$ component in the Modified score kernel helps in pushing the positive examples among the lower rank shots of a Fisher kernel to higher ranks. If the aim is to retrieve only the few top ranking shots, it does not matter whether we use the fisher kernel or the modified score kernel. But if one wishes to retrieve several shots say upto 1000, it is ideal to use the Modified score kernel.

REFERENCES

1. A. Yoshitaka and T. Ichakawa. A survey on content-based retrieval for Multimedia Databases. *Knowledge and Data Engineering, IEEE Transactions*, 11:81-93, 1999.
2. NIST, Digital Video Retrieval at NIST: TREC Video Retrieval Evaluation, 2001-2004, <http://www-nlpir.nist.gov/projects/trecvid/>.
3. Hauptmann, A.G., Christel, M.G., Successful Approaches in the TREC Video Retrieval Evaluations, Proceedings of ACM Multimedia 2004, New York City, NY, pp. 668-675, October 10-16, 2004
4. On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. A. Y. Ng and M. I. Jordan. In T. Dietterich, S. Becker and Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems (NIPS)* 14, 2002.
5. V.N Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.
6. L. Rabiner and B. H Juang, *Fundamentals of speech recognition*. New Jersey: Prentice Hall International Inc, 1993.
7. Support Vector Machine Active Learning for Image Retrieval, S. Tong and E. Y. Chang, *ACM International Conference on Multimedia*, pp.107-118, Ottawa, October 2001
8. Exploiting generative models in discriminative classifiers. T. Jaakkola and D. Haussler. In *Advances in Neural Information Processing Systems* 11, 1998.
9. G. Wahba. *Spline models for observational data*. CBMS-NSF Regional Conference Series in Applied Mathematics, 1990.
10. M. Miyahara and Y. Yoshida, Mathematical transform(R, G, B) color data to Munsell(H, V, C) color data, *SPIE Vol 1001, Visual Communications and Image Processing* 88, pp 650-7, 1988.
11. H. Schneiderman. *A Statistical Approach to 3D Object Detection Applied to Faces and Cars*. Ph.D. Thesis. Carnegie Mellon University. CMU-RI-TR-00-06.
12. N.D. Smith and M.J.F. Gales. Using SVMs and Discriminative Models for Speech Recognition. In *Proceedings, IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol 1, pages 77-80, May 2002.
13. Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley, 1999.