K-Space at TRECVid 2007

Peter Wilkins, Tomasz Adamek, Daragh Byrne, Gareth J.F.Jones, Hyowon Lee,
Gordon Keenan, Kevin McGuinness, Noel E. O'Connor, Alan F. Smeaton
Centre for Digital Video Processing & Adaptive Information Cluster
Dublin City University (DCU), Ireland

Alia Amin, Zeljko Obrenovic
CWI Amsterdam, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Rachid Benmokhtar, Eric Galmar, Benoit Huet
Département Communications Multimédia
Institut Eurécom
2229, route des Crêtes, 06904 Sophia-Antipolis, France

Slim Essid, Rémi Landais, Félicien Vallet
GET-ENST, ParisTech, LTCI/TSI
37 rue Dareau, 75014 Paris, France

Georgios Th. Papadopoulos, Stefanos Vrochidis, Vasileios Mezaris, Ioannis Kompatsiaris
Informatics and Telematics Institute (ITI),
1st Km Thermi-Panorama Road, Thessaloniki, GR-57001, Greece

Evaggelos Spyrou, Yannis Avrithis
Image Video and Multimedia Laboratory National Technical University of Athens (ITI)
9 Iroon Polytechniou Str., 157 80 Athens, Greece

Roland Mörzinger, Peter Schallauer, Werner Bailer
Institute of Information Systems and Information Management
Joanneum Research (JRS)
Steyrergasse 17, 8010 Graz, Austria

Tomas Piatrik, Krishna Chandramouli, Ebroul Izquierdo
Department of Electronic Engineering
Queen Mary, University of London (QMUL), United Kingdom

Martin Haller, Lutz Goldmann, Amjad Samour, Andreas Cobet, Thomas Sikora
Technical University of Berlin, Department of Communication Systems (TUB)
EN 1, Einsteinufer 17, 10587 Berlin, Germany

Pavel Praks
Department of Information and Knowledge Engineering
Faculty of Informatics and Statistics, University of Economics, Prague (UEP)
W. Churchill Sq. 4, 130 67 Prague 3, Czech Republic

October 22, 2007

Abstract

In this paper we describe K-Space participation in TRECVid 2007. K-Space participated in two tasks, high-level feature extraction and interactive search. We present our approaches for each of these activities and provide a brief analysis of our results.

Our high-level feature submission utilized multi-modal low-level features which included visual, audio and temporal elements. Specific concept detectors (such as Face detectors) developed by K-Space partners were also used. We experimented with different machine learning approaches including logistic regression and support vector machines (SVM). Finally we also experimented with both early and late fusion for feature combination.

This year we also participated in interactive search, submitting 6 runs. We developed two interfaces which both utilized the same retrieval functionality. Our objective was to measure the effect of context, which was supported to different degrees in each interface, on user performance. The first of the two systems was a 'shot' based interface, where the results from a query were presented as a ranked list of shots. The second interface was 'broadcast' based, where results were presented as a ranked list of broadcasts. Both systems made use of the outputs of our high-level feature submission as well as low-level visual features.

1 Overview of K-Space

K-Space is a European Network of Excellence (NoE) in semantic inference for semi-automatic annotation and retrieval of multimedia content [1] which is in the second year of its three year funding. It is coordinated by Queen Mary University of London (QMUL) and the partner responsible for coordinating the K-Space participation in TRECVid is Dublin City University. K-Space is focused on the research and convergence of three themes: content-based multimedia analysis, knowledge extraction and semantic multimedia.

This paper describes the K-Space participation in TRECVid 2007. TRECVid ([39]) is an annual benchmarking evaluation campaign for research groups to use common data and queries to assess the relative performance of their techniques in an open, metrics-based forum. 2007 marks the 7th year of TRECVid.

2 Audio-Visual Features

The K-Space submission in both feature detection and interactive search made use of several feature detectors developed by K-Space partners in prior work. Later in this section we outline the specific concept detectors contributed by individual K-Space partners used during K-Space participation in TRECVid 2007. First, though, we present some detail on the low-level visual features used. For the remainder of this paper, the term 'concept' will refer to a high-level feature (e.g. 'Car').

As no common keyframe set was released as part of the TRECVid 2007 collection, we extracted our own set of keyframes. Our keyframe selection strategy was to extract every second I-Frame from each shot. This gives us far more keyframes than the usual one-keyframe-per-shot which has been the norm in previous TRECVids and in fact gives us about 1 keyframe per second of video. For the remainder of this paper, we will refer to these images as K-Frames.

We extracted low-level visual features from K-frames using several feature descriptors based on the MPEG-7 XM. These descriptors were implemented as part of the aceToolbox, a toolbox of low-level audio and visual analysis tools developed as part of our participation in the EU aceMedia project [2]. We made use of six different global visual descriptors. These descriptors were Colour Layout, Colour Moments, Colour Structure, Homogenous Texture, Edge Histogram and Scalable Colour. A complete description of each of these descriptors can be found in [24].

We also segmented each of the K-frames into regions. We considered several approaches to image segmentation [3], [18], [7], [23], when selecting the method for automatically partitioning K-frames into large regions which reflect the objects (or their parts) present in the image. We considered not only the accuracy of segmented regions in terms of how well they mapped to object, but also the typical number of regions produced by a given algorithm and its computational cost to execute. The number of regions was a particularly relevant factor since large regions are typically more suited to subsequent robust feature estimation.

We decided to use the approach proposed in [3] which is based on the well known Recursive Shortest Spanning Tree (RSST) method utilizing the more perceptually uniform L*U*V* color model and syntactic visual features to improve the quality of the segmentation. The syntactic features represent geometric properties of regions and their spatial configurations. This approach allowed satisfactory segmentation of various types of scenes into a set of large and typically meaningful regions without adjustment to algorithm parameters.

This set of K-frames and their features and regions were distributed to all K-Space TRECVid partners so that each could run their own feature detectors on the video and send the output back to DCU for coordination. The remainder of this section describes the partner contributions.

2.1 Institute EURÉCOM

The Eurécom system for the high level features extraction task was responsible for six semantic concepts within the K-Space project (sports, outdoor, building, mountain, waterscape, maps). The Eurécom approach is based on a multi-descriptor system. The following 4 experiments were submitted to the collaborative system:

- Run 1: MPEG-7 global descriptors,
- Run 2: MPEG-7 region descriptors using the region based automatic segmentation method RBAS (A region merging approach incorporating geometric properties [4]),
- Run 3: Color and texture descriptors were extracted using three segmentation methods (A fixed image grid, watersheds [42] and a technique based on minimum spanning trees MST[17]),
- Run 4: Combination of global and regions descriptors.

These descriptors were then introduced in separate SVM classification systems (one classifier per feature) trained using the first half of the development data set. The fusion of classifiers outputs was finally provided by training a neural network based on evidence theory [10] on the second half of the training data. More details about this entire framework and its performance can be found in the notebook paper [9].

2.2 GET

GET features used in TRECVid 2007 are the outputs of a face detection module and an audio classification module which are described below.

2.2.1 Face detection

Face detection is performed thanks to the fusion of the results of two different systems. The first one, the classic Viola and Jones algorithm [44] is based on the estimation of a "strong" classifier composed of a cascade of many weak classifiers, each of these weak classifiers being attached to a particular Haar feature. A classifier dedicated to frontal faces and a second one dealing with profile faces have been applied.

The second system may be considered as a probabilistic equivalent of the Viola and Jones method [16]. While this system still relies on the estimation of a strong classifier, the difference is that the underlying classifier function is then used to estimate the distribution of the object of interest (faces in our case), that is to model the generation of such objects within images (such a model is called a "*generative model*"). As this distribution is computed, many partitions of the input images are considered and the patches they are composed of are assigned a label (*"object of interest"* versus *"background"*) depending on the estimation of likelihoods.

The results of these systems (bounding boxes) are fused thanks to geometric constraints considering the size of the bounding boxes and regarding overlaps between bounding boxes produced by different systems.

In order to reduce the number of false alarms, a colour filter (concerning chrominance channels in the YCrCb colour space) was applied. Values of this filter are derived from [28]. As such a filter would not perform efficiently on graylevel-like video frames (still shot on a graylevel picture, ...), we do not apply it on such frames using a list of graylevel-like frames provided by DCU.

2.2.2 Audio classification

The audio classification system developed is able to discriminate 17 different classes of sound, namely clean speech, noisy speech, music, music and speech, silence/pause and various environmental sounds (*i.e.* airplane, helicopter, applause, crowds, dogs, explosion, gunshot, car, race-car, siren, truck/lorry/bus, motorcycle). It relies on an efficient subset of 40 audio features which was obtained by automatic feature selection [20] from a wide set of candidate features (most of which are described in [15, 27]). The feature selection technique used is inertia ratio maximisation [26]. Features are extracted over 32-ms length overlapping analysis windows with a 50% hop-size. Temporal feature integration is used whereby features are temporally averaged over 0.5-s length texture windows. The actual classification is performed thanks to one-class Support Vector Machines following the approach presented in [35]. The fraction of each class positive outputs over a video shot length are then used as audio features.

2.3 ITI

In this section the approach followed for the detection of 10 high-level features by ITI is described. Two high-level concept detection approaches were applied. The first by ITI for the detection of *Building*, *Car* and *Waterscape-Waterfront*, and the latter by the National technological University of Athens (NTUA), for the detection of *Desert*, *Road*, *Sky*, *Snow*, *Vegetation*, *Explosion/Fire* and *Mountain*. In order to detect the aforementioned high-level features, the following MPEG-7 descriptors, which were extracted from all the available K-Frame keyframes, are used: *Scalable Color*, *Homogeneous Texture*, *Edge Histogram* and *Color Layout*.

Within the ITI approach, a *Support Vector Machines*

(SVM) structure is utilized to detect instances of the features of concern. This comprises 3 individual SVMs, one for every feature, each trained under the '*one-class*' approach [37]. After extensive experimentation, a polynomial function was used as a kernel function by each SVM. For the purpose of training the SVMs, the common TRECVID annotations [5] were employed. Then, for every keyframe the extracted, low-level descriptors were combined and their values were normalized to the interval $[-1, 1]$. The latter constitute the input to each SVM, which at the evaluation stage returns for every keyframe a numerical value in the range $[0, 1]$. This value denotes the degree of confidence to which the corresponding keyframe is assigned the high-level feature associated with the particular SVM. The metric adopted is defined as follows: For every input sample the distance z from the corresponding SVM's separating hypersphere is initially calculated. This distance is positive in cases of positive sample detection and negative otherwise. Then, a sigmoid function [41] is employed to compute the respective degree of confidence, h, as follows:

$$h = \frac{1}{1 + e^{-t \cdot z}} \tag{1}$$

where the slope parameter t is experimentally set. The SVM structure employed for high-level features detection, was realized using the SVM software libraries of [13].

Within the NTUA approach [40], all images were firstly segmented using a coarse segmentation algorithm, tuned to produce coarse segments. The aforementioned MPEG-7 descriptors were then extracted from each region. Then, K-means clustering is performed on the descriptions of all regions of the training set. After some experiments, the number of K is set to 100. Then, each cluster may or may not represent a high-level feature and each high-level feature may be represented by one or more clusters.

From each of the formed clusters, the region that lies closest to the centroid is selected and will be referred to as the "Region Type". An image will then be described semantically in terms of the region types it is composed of. Next, for each one of the keyframes, a model vector is formed. More specifically, let: $d_i^1, d_i^2, ..., d_i^j, i = 1, 2, \ldots, N_R$ and $j = N_C$, where N_C denotes the number of region types, N_R the number of the regions within the image and d_i^j is the distance of the i-th region of the image to the j-th region type. The model vector D_m is formed in the way depicted in equation 2.

$$D_m = \left[min\{d_i^1\}, min\{d_i^2\}, ..., min\{d_i^{N_C}\} \right], i = 1, 2, \ldots, N_R \tag{2}$$

For each semantic concept, a separate neural network-based detector was trained. Its input was the model vector and the output represents the distance of each region to the corresponding semantic concept. For the training of these detectors the common annotation has been used.

2.4 JRS

In the feature extraction task, JRS contributed with the extraction of a number of visual indexes, as described in the following.

The extraction algorithm for camera motion is the same that we used for the TRECVID 2005 camera motion task [8]. It is based on feature tracking using the Lucas-Kanade tracker, which is a compromise between spatially detailed motion description and performance. The feature trajectories were then clustered by similarity in terms of a motion model. The clustering algorithm is an iterative approach of estimating a motion parameter sequence for a set of trajectories and the re-assigning trajectories to the best matching parameter sequence. The cluster representing the global motion is selected. The decision is based on the size of the cluster and its temporal stability. Based on the parameter sequence representing the dominant motion, the presence of pan, zoom and tilt is detected. For one or more segments per shot, the following types of motion are described: pan left/right, tilt up/down, zoom in/out and static.

The level of visual activity is computed by temporally subsampling the video and computing the mean absolute frame differences (MAFD). The description then contains statistics about minimum, maximum, mean and median MAFD per shot.

For each shot, the number of faces is detected on the temporally subsampled video, by using the face detection method implemented in OpenCV. The mode of the number of detected faces in the frames is described for each shot. Additionally, information about the biggest face size is given.

For calculating shot and keyframe similarity, we use four different global image features, specifically the MPEG-7 features [22] ColorLayout, DominantColor, ColorStructure and EdgeHistogram. The description of each shot contains a list of relations to similar shots.

The description of all feature extraction results is in MPEG-7 format compliant to the Detailed Audiovisual Profile (DAVP) we have specified [6].

2.5 QMUL

In the feature extraction task, QMUL extracted the following six features: "Maps", "Sky", "Weather", "US-Flag", "Boat/Ship" and "Vegetation". These features were extracted by two classification modules developed within MMV group in QMUL: Particle Swarm Optimisation based image classifier and Ant Colony based image

classifier. The feature vectors used for feature extraction were extracted from the segmentation of video frames. A brief introduction to each module is given as follows: the Particle Swarm Optimisation (PSO) technique is one of the meta-heuristic algorithms inspired by Biological systems. The image classification is performed using the Self Organising Feature Map (SOFM) and optimising the weight of the neurons by PSO [12]. To improve the performance of the classification algorithm, fuzzy inference rules are constructed along with Binary Particle Swarm to merge the classification results from multiple MPEG - 7 descriptors [11]. The rules were explicitly weighted based on the ability of the descriptor to classify different features/concepts. The PSO based classifier was used for extraction of "Maps", "Sky" and "Weather" features. Next module is the Ant Colony based image classifier where the Ant Colony Optimisation (ACO) and its learning mechanism is integrated with the COP-K-Means to address image classification problem [29]. The COP-K-Means is a semi-supervised variant of K-Means, where initial background knowledge is provided in the form of constraints between instances in the dataset. The integration of ACO with a COP-K-Means makes the classification process less dependent on the initial parameters, so that it becomes more stable. An 'ACO based classifier was used for extraction of "US-Flag", "Boat/Ship" and "Vegetation" features. Both modules are designed to handle the very large TRECVid dataset, considering both the classifier performance and the processing time.

2.6 TUB

2.6.1 Speaker change detection

The goal of the speaker change detection developed by K-Space partner TUB, is to detect change points between individual speakers and segment the audio stream into nonoverlapping speaker segments. It was realized using the Bayesian information criterion (BIC), a parametric model selection method which was first proposed in [38]. To apply BIC for speaker change detection, the audio stream was first divided into homogenous audio segments with a length of 2s. A sliding window divided these audio segments into overlapping frames with a length of 40 ms and an overlap of 20 ms. For each frame a feature vector consisting of 13 mel frequency cepstral coefficients (MFCC's) [14] and the log energy of the frame are extracted. With the assumption that an audio segment consisting of a set of consecutive feature vectors is drawn from an independent multivariate Gaussian process and contains at most one speaker change point at a certain time, the segmentation problem can be treated as a model selection problem between the models of two contiguous audio segments. A frame is a good candidate segment boundary if the variation between two consecutive BIC values is larger than 0. The final change point decision is made via Maximum Likelihood Estimation (MLE).

2.6.2 Optical character recognition

In a video OCR program a sequence of frames is used for the detection of text. This sequence of frames carries informations about moving objects and static text. This information is not available on a single image. For example in a soccer game the camera is moving all the time or the players are moving, but the text with the score, time, and teams are always on the same place in the frames. To find the text in this example video it is necessary to remove all moving objects. To remove the moving objects all frames of one sequence are used. After the edge filter is done, each of these frames is used for a logical multiplication. Thus most of the moving objects are removed and only the static edges of the text areas are left. In this way it is possible to detect the text regions.

2.6.3 Face detection

The goal of face detection is to detect and localize frontal faces within the keyframes of a shot and provide some face statistics for the subsequent fusion and search modules. Since the images of the TRECVID 2007 data have only a low resolution (CIF) the holistic approach proposed by Viola & Jones [43] was adopted. In order to decrease the number of false positives it was combined with a postfiltering step based on skin color probabilities. Although it performed quite well under various controlled conditions, a lot of real faces were discarded in uncontrolled scenarios. Since the dataset contained also a lot of monochrome sequences the postfiltering was not used for the submitted results. So almost the same approach as described in [45] was applied to the TRECVid 2007 data. In contrast to TRECVid 2006 the face detection was applied only both to the keyframes and the K-frames within a shot.

2.6.4 Audio classification/segmentation

Audio classification/segmentation determines temporal audio segments. The membership of each audio segment is given as confidence value for the six audio classes pause, clean speech, noisy speech, pure music, music and speech as well as environmental sound. Almost the same approach as described in [45] for audio classification/segmentation was applied to the audio streams of the TRECVid 2007 video data. The minor difference for this year is the change of the sub-segment duration from 0.5 s to 1 s. This change is motivated by a higher stability of results determined by a maximum likelihood (ML) classification for each sub-

segment. Finally, the audio segments are formed by joining sub-segments that are belonging to the same audio class.

2.7 UEP

2.7.1 Latent Semantic Indexing for automated image retrieval

Originally, Latent Semantic Indexing (LSI) [19] was developed for retrieval of large amounts of text documents especially because of difficulties in effective matching of terms due to polysemy and synonymy. We extended the original LSI for intelligent image retrieval in [30]. In our previous approach [30, 33], a raster image was coded as a sequence of pixels and then the coded image can be understood as a vector in an m-dimensional space, where m denotes the number of pixels (attributes). We successfully used our approach especially for surveillance [34, 31] and as an automated tool for the large-scale iris recognition problem [33]. We also showed that image retrieval can be powered very effectively when the time-consuming Singular Value Decomposition of the original LSI is replaced by the partial symmetric eigenproblem which can be solved very effectively by using fast iterative solvers [33]. However, our previous approach produced large non-sparse document matrices which is why we used a novel sparse image representation for the TRECVid 2007 image similarity task in matching K-frames against query images. Our novel sparse approach is based on the Fast Fourier transform (FFT) and also on a statistically-based model for the efficient dimensional reduction of sparse FFT data [32]. These reduced sparse data are analyzed by the fast SVD-free LSI approach [33].

Although images can be represented very effectively by FFT sparse coefficients, the sparsity character of these coefficients is destroyed during the LSI-based dimension reduction process represented by the sparse partial eigenproblem. In our TRECVid 2007 approach, we kept the memory limit of the decomposed data by a statistical model of the sparse data. Each analyzed image was processed by the following steps:

1. Representation of the image by $352 \times 288 = 101\ 376$ dense "keywords". We use the approach described in [33].

2. Generation of an FFT sparse representation of these dense "keywords". After FFT, only 1 % of these coefficients remain as non-zeros.

3. Automated reduction of the size of these FFT-based sparse coefficients by a statistical model of data [32]. Each image is finally represented by only 3% "keywords" (i.e. by only 3 % of the original number of keywords.). Moreover, these coefficients remain sparse [32].

4. Finally, the sparse document matrix is constructed and analyzed by the SVD-free algorithm [33]. For numerical results related to the video BG_38002 see Table 1.

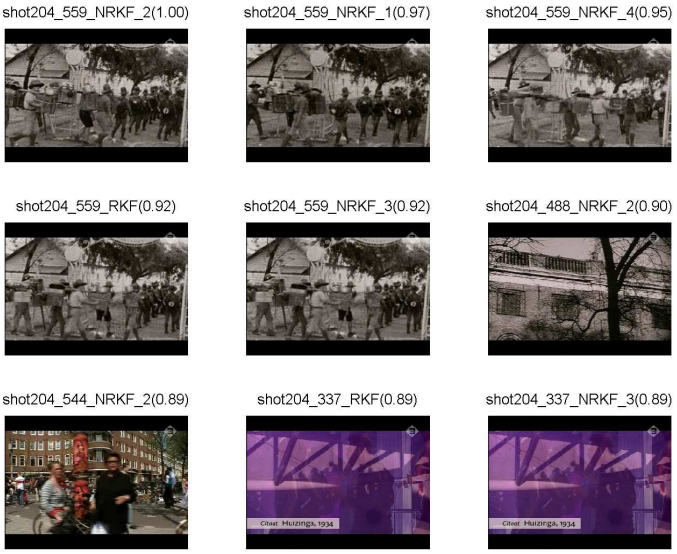

Figure 1: An example of the SVD-free LSI keyframe similarity user-interface. The query image (shot204_559_NRKF_2.jpg) is situated in the left upper corner and has the similarity coefficient 1. All of the 5 most similar images are related to the same topic. Images are automatically sorted in the same way as it would be sorted by a human user. More remote images are not related to the query image at all.

2.7.2 K-Space TRECVid 2007 Keyframe similarity results

For TRECVid 2007 we processed each video of the test collection separately by our SVD-free LSI approach, see Figure 1. This meant that we created 109 separate document matrices, see Table 1. We used the MATLAB's sparse matrix storage format for document matrices, for details see Table 1. All computations were stable and fast on a notebook with Intel(R) Pentium(R) M processor 1.86GHz with 0.98 GB RAM. One of the reasons for this is that singular values of TRECVid 2007 keyframes tend to decrease quite fast so that only 8 extremal eigenvalues and corresponding eigenvectors of the sparse partial symmetric eigenproblem were computed and stored in memory in all cases. The second reason for the fast execution is that we

Properties of the document matrix *A*	
Number of keywords:	3 042
Number of documents:	6 605
Size in memory:	76.67 MB
The SVD-Free LSI processing parameters	
Dim. of the original space	6 605
Dim. of the reduced space (k)	8
The total time	30.1 secs.

Table 1: Image retrieval using the SVD-free Latent Semantic Indexing method related to the BG_38002 video; Properties of the document matrix (up) and LSI processing parameters (down). Decompressing of original JPGs onto bitmaps required 1437.73 secs (i.e. 4.56 images per sec.).

used efficient implementation of linear algebra algorithms with assuming several key implementation details [33] and also the new developed sparse approach. This new approach is very efficient in sense of the computer memory and computer time and also applicable for large-scale full automated image retrieval applications. The effectiveness of our novel approach was demonstrated by the large scale image similarity task of the TRECVid 2007.

3 High-Level Feature Extraction

For our participation in the High-Level Feature Extraction task, we created five runs. The majority of these runs utilized some aspect of machine learning, for which we used the WEKA toolkit [47]. All of our approaches used the training data generated through the collaborative annotation activity organized by LIG [5]. We will describe each of our approaches below.

KSpace_1 Baseline: Our baseline approach was to take the six global visual features, and perform early fusion to create a single attribute vector for each K-Frame. Using logistic regression, we classified each test K-Frame for each concept. Aggregation back to the shot level was achieved by using MAX on the set of K-Frame predictions for that shot, and using the highest positive prediction as the representative for that shot.

KSpace_2 Combination of K-Space concept detectors: In this approach, we first took each of the K-Space concept detectors (such as ITI's waterscape detector) and using cross-validated logistic regression, examined the concept's ability to detect each of the TRECVid concepts on the training data. For example, we measured how well TUB's face detector was at detecting people, cars, urban etc. Then for each TRECVid concept, we selected those K-Space concepts which maximized the true positive and false positive rate, combined these K-Space detectors through early fusion and trained SVM's on these combinations. For instance, for the 'Car' feature, we combined GET's audio features and JRS motion features to produce our classifier.

KSpace_3 Lightweight audio visual detection: This run experimented with producing a fast classification which made use of multiple modalities to aid in classification. It was comprised of global visual features Colour Layout and Homogenous Texture, as well as JRS motion and GET audio classifiers. Each of these features were fused together, then we used logistic regression to perform the classification on the TRECVid concepts. Both Colour Layout and Homogenous Texture are our most compact colour and texture visual features.

KSpace_4 Single K-Space concept detectors: For this run, we selected a single K-Space concept detector which matched a TRECVid concept. E.g. for the waterscape concept we submitted the results from the QMUL waterscape detector. Where there was not a K-Space concept detector for a given concept, we substituted in the results from the baseline visual classification. This gives us a baseline performance for the K-Space detectors.

KSpace_6 Single feature SVM, late fusion: In our final run, we first created individual SVM's for each of our six global visual features. The predictions of each of these SVM's were then late fused via a MAX operator. Classification was at the K-Frame level with aggregation to the shot level the same as per the baseline run.

Our results for these submissions can be found in Table 2. The first observation we can make is that overall performance is down as compared to last year and this is probably because the data is more difficult and challenging. Second, that our approaches this year were not as successful as last year. Our most successful approach last year was based on global visual features fused together through early fusion, then classification was performed through the use of SVM's. The baseline run this year was approximately the same except that we used logistic regression instead of SVM's. We could speculate that this approach would have performed better if we had utilized SVMs.

The most interesting result for us was that our lightweight multi-modal approach was our best-performing run. This approach was also one of our quickest to train, with classification and test being achieved within 24 hours on a dual core Pentium Xeon 5160. We believe that if we were to extend this approach to include more visual features, that we would see an increase in performance. This run also emphasizes the benefits that can be gained from not restricting our training data to the visual domain, but to also incorporate audio and temporal features.

Concept	median	KSpace 1	KSpace 2	KSpace 3	KSpace 4	KSpace 6
1	0.028	0.015	0.004	0.008	0.000	0.022
3	0.002	0.003	0.000	0.003	0.000	0.005
5	0.061	0.046	0.023	0.079	0.015	0.030
6	0.053	0.065	0.086	0.122	0.024	0.022
10	0.008	0.001	0.000	0.001	0.001	0.005
12	0.003	0.009	0.003	0.028	0.006	0.021
17	0.167	0.151	0.151	0.149	0.021	0.157
23	0.003	0.002	0.002	0.012	0.005	0.006
24	0.005	0.012	0.007	0.008	0.002	0.001
26	0.074	0.077	0.005	0.074	0.047	0.108
27	0.051	0.035	0.035	0.021	0.003	0.019
28	0.000	0.000	0.000	0.000	0.000	0.000
29	0.022	0.045	0.002	0.009	0.006	0.013
30	0.082	0.076	0.035	0.117	0.010	0.059
32	0.026	0.018	0.018	0.036	0.016	0.026
33	0.083	0.020	0.020	0.071	0.071	0.033
35	0.028	0.002	0.012	0.015	0.012	0.013
36	0.005	0.004	0.001	0.006	0.007	0.001
38	0.035	0.015	0.002	0.018	0.002	0.071
39	0.017	0.006	0.006	0.021	0.004	0.014
average	0.039	0.030	0.021	0.040	0.013	0.031

Table 2: 2007 K-Space Feature Detection Results

4 Interactive Search

For interactive search, we defined our experimental objectives as investigating the role of context in the user experience. We created two distinct user interfaces to investigate the role of context, whilst keeping the functionality of both systems the same. In this way we limited the scope of the experiment to just the user interface, which is a departure from some of the K-Space partner's earlier interactive experiments where the objective was often to measure the impact of some functionality on system performance (e.g. the effect of text in retrieval performance versus visual-only retrieval).

4.1 Experiment Design

Our experiment was designed to examine the role of context in the user interface, where context can be described as showing for a given shot, temporally adjacent shots which a retrieval engine may or may not have ranked. An example of the display of temporal context would be to issue to a retrieval engine, a visual query of an anchor person from a news broadcast. A temporal context response would be to return to the user, not just matching shots of anchor persons, but also the news story shots that they were presenting, which would not be visually similar to the initial query.

To examine the role of context, we designed two user interfaces, known as the 'shot based' system, and the 'broadcast based' system. Both systems, apart from sharing the same retrieval engine, also shared a common query-panel, topic description panel and saved shot area. The major difference was in the presentation of the results from the retrieval engine.

The 'shot based' system presented to the user the ranked list of shots direct from the retrieval engine. Presented in Figure 2 it can be seen that the ranked shots are organized left to right, top to bottom. It can be thought of as the more traditional result display that has been used for content-based retrieval interfaces. This interface displays no context for any of the returned results.

The 'broadcast based' system takes the idea of context to its maximum by ranking not shots, but broadcasts. If we were to take the assumption that the corpus for this year will have broadcasts which are more homogenous on a subject (i.e. a documentary may be about one major subject, whilst in previous years a news broadcast could be seen as containing many subjects), then ranking broadcasts as opposed to shots appears as an interesting alternative. In Figure 3 we can see a horizontal line of shots in rows across the results area. Each of these rows is a ranked entire broadcast, with the best-matching broadcast being the first row. When a user issues a query, the ranked list of broadcasts is presented, and within each broadcast's row the row will be centered on the highest matching shot

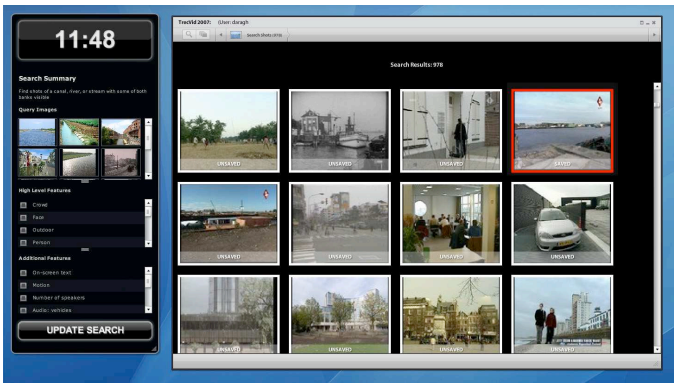

Figure 2: Shot-based user interface

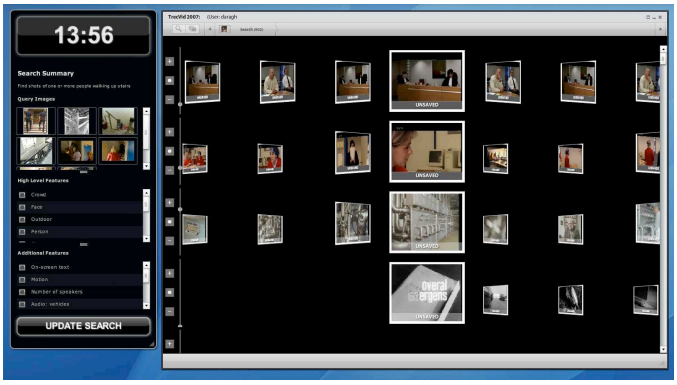

Figure 3: Broadcast-based user interface

within that broadcast.

For the user experiment, we had two distinct groups of users. We had two 'expert' users who were intimately involved in the design and development of the user interfaces and retrieval engine. These users however were not exposed to the test collection. This allowed us to conduct two 'expert' retrieval runs, where one expert used only the shot system, and the other expert used only the broadcast system.

We also had 8 users from the Centre for Digital Video Processing (CDVP) in DCU, who whilst familiar of the concept of content-based retrieval had not been strongly involved in the TRECVid 2007 activity. These users we classify as non-expert users as they would not have been aware of the details of either of the interfaces or the retrieval engine. Using a Latin squares arrangement, we were able to perform two runs on each system with these users, with each user performing 12 topics, six on each system.

4.2 Retrieval Engine

For interactive search we had seven retrieval experts available for use by the user. These included the six global visual features as identified in Section 2. We also made use of the Machine Translated (MT) Automatic Speech Recognition (ASR) output [21]. The alignment of this data to the shot boundaries was performed by ITI. The text was then indexed by Terrier [25], with retrieval results provided through a vector space model [36].

For our visual experts, ranking within each was handled by the similarity measures as specified by the MPEG7 specification [22]. These measures for the most part are similar to Euclidian distance.

When specifying a multi-modal query, the user can select to use any or all of these seven experts to retrieve a response. When a query is issued, it goes to each of the retrieval experts and a ranked list is returned. Using a variation on DCU's query-time weight generation techniques [46], these result lists are merged at query time with weights being assigned to each expert which approximate that experts likelihood of providing the most relevant responses to the query.

The user interfaces also allows the user to make use of the concept detectors developed in Section 3. These concepts are used as filters by the user after a content-based query has been issued. These filters could be set to 'positive', 'negative' or 'off'. For instance, for a query about cars, a user may set a 'car' concept to 'positive' and a face concept to 'negative'. To allow for false positives by the concept detectors, when they were used by a user, they did not alter the result list by removing non-matching elements, rather non-matching elements were greyed out but were still visible. The intention was to allow the user to scroll down a result list with their attention drawn to the brighter elements. In hindsight it would have been better if these non-matching results were removed.

4.3 Results

The results of our interactive experiment can be seen in Table 3. Our analysis of these results has only just begun, and as such any conclusions we present now are only preliminary.

Our first major conclusion to draw is the difference in performance between the expert and non-expert submissions. This difference is more pronounced for the shot-based system than the broadcast system. It certainly highlights the user effect on the results of an interactive experiment.

System	MAP	Non-augmented
Expert Broadcast	0.167	0.138
Expert Shot	0.199	0.162
Non-Expert Broadcast 1	0.146	0.109
Non-Expert Broadcast 2	0.145	0.111
Non-Expert Shot 1	0.146	0.110
Non-Expert Shot 2	0.133	0.096

Table 3: 2007 K-Space Search Results

The second conclusion we can draw is that the examination of MAP performance is insufficient to draw any confident conclusions of the superiority of one system over another. In the case of the expert runs, the shot system had the advantage, however for non-expert users the broadcast system appears to have slightly better performance. We also note at this stage though, based on our follow-up questionnaires with experiment participants, that for several of the topics there was disagreement between what the user thought was the objective and the assessor. Because of this we will have to examine other metrics, such as pure recall of the systems do get a clearer picture of system performance, and perhaps on a per-topic basis.

Our augmentation process of user results provides a consistent boost to performance across all runs, adding an additional 0.03 and 0.04 to MAP in all cases, an average boos of 30%. As mentioned previously, this augmentation was the issuing of a 'more like this' query using whatever saved shots the user had selected. The consistent increase in performance demonstrates the boost that can be achieved using our query-time fusion techniques.

5 Conclusion

We have presented the K-Space participation in TRECVid 2007. This was our second participation in TRECVid as a large group of research teams drawn together in an EU-funded network, and our participation has developed on last year with participation in interactive search. Our results for the High-Level Feature Extraction task are down on our previous efforts, with further examination required to determine changes in our approach. The search task saw a marked improvement from our previous manual search participation, with the K-Space runs achieving good performance above the median. Deeper analysis will now be required to determine what drove user performance and to identify areas for future improvement. Our participation has again been a positive experience for our partners and we look forward to greater participation in next year's TRECVid activities.

6 Acknowledgments

The research leading to this paper was supported by the European Commission under contract FP6-027026 (K-Space).

References

[1] KSpace Network of Excellence, information at http://www.k-space.eu/.

[2] The AceMedia Project, available at http://www.acemedia.org.

[3] T. Adamek and N. O'Connor. Using dempster-shafer theory to fuse multiple information sources in region-based segmentation. In *ICIP 2007 - Proceedings of the 14th IEEE International Conference on Image Processing*, 2007.

[4] T. Adamek, N. O'Connor, and N. Murphy. Region-based segmentation of images using syntactic visual features. In *WIAMIS 2005 - 6th International Workshop on Image Analysis for Multimedia Interactive Services*, 2005.

[5] S. Ayache and G. Quénot. Evaluation of active learning strategies for video indexing. *Image Commun.*, 22(7-8):692–704, 2007.

[6] W. Bailer and P. Schallauer. The Detailed Audiovisual Profile: Enabling Interoperability between MPEG-7 based Systems. In *12^{th} International MultiMedia Modelling Conference (MMM'06)*, pages 217–224, Beijing, China, 2006.

[7] W. Bailer, P. Schallauer, H. B. Haraldsson, and H. Rehatschek. Optimized mean shift algorithm for color segmentation in image sequences. In A. Said and J. G. Apostolopoulos, editors, *Image and Video Communications and Processing 2005. Edited by Said, Amir; Apostolopoulos, John G. Proceedings of the SPIE, Volume 5685, pp. 522-529 (2005).*, volume 5685 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 522–529, Mar. 2005.

[8] W. Bailer, P. Schallauer, and G. Thallinger. Joanneum Research at TRECVID 2005 – Camera Motion Detection. In *Proceedings of TRECVID Workshop*, pages 182–189, Gaithersburg, Md., USA, 11 2005. NIST.

[9] R. Benmokhtar, G. Eric, and B. Huet. Trecvid 2007: high level features extractions. In *TRECVID 2007 Workshop*, Gaithersburg, MD, November 2007.

[10] R. Benmokhtar and B. Huet. Neural network combining classifier based on Dempster-Shafer theory for semantic indexing in video content. In *MMM 2007, International MultiMedia Modeling Conference, January 9-12 2007, Singapore - Also published as LNCS Volume 4351*, Jan 2007.

[11] K. Chandramouli, D. Djordjevic, and E. Izquierdo. Binary particle swarm and fuzzy inference for image classification. In *Proceedings of Proceedings of 3rd International Conference on Visual Information Engineering 2006*, pages 126–131, 1988.

[12] K. Chandramouli and E. Izquierdo. Image Classification using Self-Organising Feature Map and Particle Swarm Optimisation. In *Proceedings of 3rd International Conference on Visual Information Engineering 2006*, pages 313–316, 2006.

[13] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines, 2001.

[14] S. B. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366, Aug. 1980.

[15] S. Essid. *Automatic Classification of Audio Signals: Machine Recognition of Musical Instruments*. PhD thesis, Université Pierre et Marie Curie, 2005.

[16] I. Fasel, B. Fortenberry, and J. Movellan. A Generative Framework for Real-Time Object Detection and Classification. *Computer Vision and Image Understanding*, 98(1):182–210, 2004.

[17] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[18] E. Galmar and B. Huet. Graph-based spatio-temporal region extraction. In A. C. Campilho and M. S. Kamel, editors, *ICIAR (1)*, volume 4141 of *Lecture Notes in Computer Science*, pages 236–247. Springer, 2006.

[19] D. Grossman and O.Frieder. *Information retrieval: Algorithms and heuristics*. Kluwer Academic Publishers, Second edition, 2000.

[20] I. Guyon and A. Elisseeff. An introduction to feature and variable selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[21] M. Huijbregts, R. Ordelman, and F. de Jong. Annotation of heterogeneous multimedia content using automatic speech recognition. In *Proceedings of the second international conference on Semantics And digital Media Technologies (SAMT)*, Lecture Notes in Computer Science, Berlin, December 2007. Springer Verlag.

[22] MPEG-7. Multimedia Content Description Interface. Standard No. ISO/IEC n°15938, 2001.

[23] R. Nock and F. Nielsen. Statistical region merging. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(11):1452–1458, 2004.

[24] N. O'Connor, E. Cooke, H. le Borgne, M. Blighe, and T. Adamek. The AceToolbox: Low-Level Audiovisual Feature Extraction for Retrieval and Classification. In *2nd IEE European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies*, 2005.

[25] I. Ounis, C. Lioma, C. Macdonald, and V. Plachouras. Research directions in terrier. *Novatica/UPGRADE Special Issue on Web Information Access, Ricardo Baeza-Yates et al. (Eds), Invited Paper*, 2007.

[26] G. Peeters. Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization. In *115th AES convention*, New York, USA, oct 2003.

[27] G. Peeters. A large set of audio features for sound description (similarity and classification) in the cuidado project. Technical report, IRCAM, 2004.

[28] S. Phung, A. Bouzerdoum, and D. Chai. A Novel Skin Color Model in YCrCb Color Space and its Application to Human Face Detection. In *International Conference of Image Processing*, volume 1, pages 289–292, 2002.

[29] T. Piatrik and E. Izquierdo. Image classification using an ant colony optimization approach. In *Proceedings of 1st International Conference on Semantic and Digital Media Technologies*, 2006.

[30] P. Praks, J. Dvorský, and V. Snášel. Latent semantic indexing for image retrieval systems. In *SIAM Linear Algebra Proceedings, Philadelphia, USA*. International Linear Algebra Society (ILAS), http://www.siam.org/meetings/la03/proceedings/-Dvorsky.pdf, July 2003.

[31] P. Praks, M. Grzegorzek, R. Moravec, L. Válek, and E. Izquierdo. Wavelet and eigen-space feature extraction for classification of metallography images. In *17th European-Japanese Conference on Information Modeling and Knowledge Bases (EJC 2007)*. Juvenes Print-TTY, Tampere, Finland, June 2007.

[32] P. Praks, E. Izquierdo, and R. Kučera. The sparse image representation for automated image retrieval. Work in progress, 2008.

[33] P. Praks, L. Machala, and V. Snášel. *On SVD-free Latent Semantic Indexing for Iris Recognition of Large Databases*. Springer-Verlag London, In: V. A. Petrushin and L. Khan (Eds.) Multimedia Data mining and Knowledge Discovery (Part V, Chapter 24), 2007.

[34] P. Praks, J. Černohorský, V. Svátek, and M. Vacura. Human expert modelling using semantics-oriented video retrieval for surveillance in hard industry. In *ACM MobiMedia 2006: 2nd International Mobile Multimedia Communications Conference*. K-Space special session on Automatic Annotation and Retrieval of Multimedia Content, Alghero, Sardinia, Italy, September 2006.

[35] A. Rabaoui, M. Davy, S. Rossignol, Z. Lachiri, and N. Ellouze. Using one-class svms and wavelets for audio surveillance systems. *submitted to IEEE trans. on Information Forensic and Security*.

[36] G. Salton. *Automatic Text Processing*. Addison–Wesley, 1989.

[37] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

[38] G. Schwarz. Estimation the dimension of a model. In *Ann. Stat.*, volume 6, pages 461–464, 1978.

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

[40] E. Spyrou and Y. Avrithis. A region thesaurus approach for high-level concept detection in the natural disaster domain. In *2nd International Conference on Semantics And digital Media Technologies (SAMT)*, 2007.

[41] D. Tax and R. Duin. Using two-class classifiers for multiclass classification. In *Int. Conf. Pattern Recognition, Quebec City, Canada vol. 2*, 2002.

[42] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *PAMI*, 13(6):583–598, June 1991.

[43] P. A. Viola and M. J. Jones. Robust real-time object detection. In *IEEE Workshop on Statistical and Computational Theories of Computer Vision*, 2001.

[44] P. A. Viola and M. J. Jones. Robust Real-Time Object Detection. *International Journal of Computer Vision*, 57(2):137–154, 2002.

[45] P. Wilkins, T. Adamek, P. Ferguson, M. Hughes, G. J. F. Jones, G. Keenan, K. McGuinness, J. Malobabic, N. E. OConnor, D. Sadlier, A. F. Smeaton, R. Benmokhtar, E. Dumont, B. Huet, B. Merialdo, E. Spyrou, G. Koumoulos, Y. Avrithis, R. Moerzinger, P. Schallauer, W. Bailer, Q. Zhang, T. Piatrik, K. Chandramouli, E. Izquierdo, L. Goldmann, M. Haller, T. Sikora, P. Praks, J. Urban, X. Hilaire, and J. M. Jose. K-space at trecvid 2006. In *Proceedings of the TRECVid Workshop*, Gaithersburg, Maryland, USA, Nov. 2006.

[46] P. Wilkins, P. Ferguson, and A. F. Smeaton. Using score distributions for querytime fusion in multimedia retrieval. In *MIR 2006 - 8th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2006.

[47] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, second edition, June 2005.