Video Classification and Retrieval with the Informedia Digital Video Library System

A. Hauptmann, R. Yan, Y. Qi, R. Jin, M. Christel, M. Derthick,
M.-Y. Chen, R. Baron, W.-H. Lin, and T. D. Ng.

Carnegie Mellon University,
School of Computer Science,
Pittsburgh PA, 15213-3891 USA

This paper is organized in three parts. The first part details some of the lower level shot classification work, the second part describes the 'manual' retrieval systems while the last section details the interactive retrieval system for the Carnegie Mellon University TREC Video Retrieval Track runs. The description of the data can be found elsewhere in the proceedings of the 2002 TREC conference video track overview.

Classification

In the TREC02 video track, one of the main tasks is to detect various semantic features concepts such as "Indoor/Outdoor", "People" etc. This part contains the description of the classification tasks. We submitted runs for the following classification concepts in the TREC 2002 Video Track. To obtain training data, we manually annotated each I-frame of the 23.26 hours feature development collection for each category.

a. **Outdoors**: Segment contains a recognizably outdoor location, i.e., one outside of buildings. Should exclude all scenes that are indoors or are close-ups of objects (even if the objects are outdoor).
b. **Indoors**: Segment contains a recognizably indoor location, i.e., inside a building. Should exclude all scenes that are outdoors or are close-ups of objects (even if the objects are indoor
c. **Cityscape**: Segment contains a recognizably city/urban/suburban setting.
d. **Monologue:** an event in which a single person is at least partially visible and speaks for a long time without interruption by another speaker
e. **Face:** at least one human face with nose, mouth, and both eyes
f. **People: a** group of two more humans
g. **Text Detection:** superimposed text large enough to be read
h. **Speech:** human voice uttering recognizable words
i. **Instrumental Sound:** sound produced by one or more musical instruments, including percussion instruments

Feature Extraction

It is a critical challenge to find a good feature set for image classification. A number of image features based on color and texture attributes have been reported in the literature for image retrieval. We tried several of them and explored some new features at the same time.
Color Histograms. We used the histogram of 3*3 image regions in HSV color space for each MPEG I-frame. The color features were derived from a histogram in the quantized HSV color space.
Textures. We use the mean and variance of a texture orientation histogram for each of the 3*3 regions as texture feature.
Edge features. We used a feature called the Edge Direction Histogram. A Canny edge detector was used to extract the edges from an image. A total of 73 bins were used to represent the edge direction histogram of an image; the first 72 bins are used to represent the edge directions quantized at 5^o intervals and the last bin represents a count of the number of pixels that didn't contribute to any edge.
Edge direction coherence vector. This feature stores the number of coherent versus non-coherent edge pixels with the same edge directions (considering only horizontal and vertical axis within a range of +/- 5^o,). We thresholded on the size of every 8 connected components of edges in a given direction to decide whether the region could be considered coherent or not. This feature was used to distinguished structured edges (like edges of buildings) from arbitrary edge distributions.

Camera motion. We used statistical distribution patterns to detect the pan/tilt/zoom camera operations based on the motion vectors of MPEG encoding. The resulting features encoded the presence/absence of these six kinds of camera operations (pan left, pan right, tilt up, tilt down, zoom in, zoom out) as a new type of feature for image classification.

MPEG motion vectors. We transformed the motion vectors directly encoded in the MPEG-compressed video into a different kind of feature, namely a histogram of the motion vector angle and velocity, as well as the wavelet coefficients of motion vectors..

Although we experimented extensively with the features derived from camera motion analysis and the raw MPEG motion vectors, these additional features did not contribute to overall classification accuracy.

Classification Algorithms

We experimented with several classification tools for these tasks, including SVM, KNN, Adaboosting and Decision Trees. Comparing their performance using cross validation on a comparative large data set, we reached the conclusion that support vector machine learning was best, with the power=2 polynomial as the kernel function. Nonlinear functions usually performed better than linear SVM kernel functions. The trade-off is that for nonlinear functions, the parameter space can be huge and therefore it may cause overfitting for small training datasets.

Among the tasks, the cityscape classification suffered from the problem of insufficient positive training examples, which is also the reason why we did not submit a landscape classification for evaluation. For the cityscape classification training data, the positive examples (that is, the cityscape images vs. the non-cityscape images) comprised only 12% of the whole data set. Such small ratios of positive examples in the training set cannot be well represented by the classification methods we attempted. In addition, we investigated using the chi-square function as distance function based on published literature. Contrary to published claims, the chi-square function was not superior to any other functions.

Cross_validation

Due to the temporal correlation between adjacent images in a video, an initial cross validation based on random sampling of shots gave much better performance than appropriate for the true prediction capability of the models. This was due to the fact that similar shots appeared throughout a single video or 'movie'. So we performed a video based cross validation based, using 30 complete videos as training and then testing on the remaining 11 videos.

Feature Selection

It is a challenge to select a good feature set for image classification. Qualifying their discrimination ability of each feature in the given classification problem is difficult. We performed video-based cross validation on training sets and compared the different features' performance based on the resulting classification error and precision / recall of each task.

For the camera motion related features and the MPEG motion vector related features, we explored numerous experiments to test their usefulness to the image classification task. However, they did not give conclusive results clearly. Finally, we ended up not using the camera or motion features in the final submission.

To get the best feature combination for each task, we performed a 6 folder movie based cross validation on the three training sets on different feature combinations. The best feature combinations were always included texture, edge and color features . Since the results were submitted as shot based features and not classification from individual images, we integrated all I frame classification results in a shot into this shot's feature detection result. The confidence of a particular feature detection is the ratio between number of feature presenting I frames vs. number of feature absent I frames in this shot.

Our results showed huge difference of the performance of different classifiers. The reason of this discrepancy is possibly caused by the variability of the training sets, the inconsistency between training set and the test sets, or the varying difficulty of the different classifications.

Non-standard classification for text, faces, people, monologues, speech and music

Variations of the classification approach were used for the face, people, monologue and audio categories.

For the **face** category, we used the Schneiderman face detector [19], exclusively.

For **people**. we extracted the following features

At the level of shots:

- Number of frames in a shot

- Number of faces detected by the face detector
- Number of faces with high confidence
- Number of faces with low confidence
- Average confidence score of the faces in a shot,
- The standard deviation of the face scores,
- A smoothed minimum face score,
- A smoothed maximum score,
- Average pixel area for each detected face.

For each I-frame within a shot we also extracted these frame-based features:

- Average number of faces per frame,
- Average number of faces per frame with high confidence
- Average number of faces per frame with low confidence

Since the total number of features was fairly low, we trained a decision-tree based classifier (C4.5), which outperformed SVM on this task in cross-validation experiments.

Our contrasting **people** classification submission merely counted the number of faces visible in each I-frame, and averaged this over the whole shot. This baseline approach performed significantly worse with a classification error of 0.403 vs. 0.498.

The task of **text-overlay** classifier is to find scenes with superimposed texts. Simply predicting a scene to be a text overlay based on whether or not the OCR engine is able to find text is not good enough because that OCR engine is quite error-prone. The features extracted were:

1. time: related to the whole movie, when is the OCR detected texts are found
2. #terms_within_a_shot
3. #dictionary_words_within_a_shot
4. average_popularity_valid_trigram_in_a_shot
5. average_popularity_valid_4gram_in_a_shot
6. average_no_alphabets_found_in_a_term
7. ratio_dictionary_words_to_detected_terms
8. ratio_length_of_all_dictionary_words_to_length_of_detected_terms

For classification, similar to the people classifier, a decision tree (C4.5) was used instead of a SVM.

For **monologues**, we used as features:

1. The portion of time where a least one (face) was detected.
2. The confidence of the face in every I-frame.
3. The number of speaker voice changes in one shot
4. The confidence in any significant audio change during this shot.
5. The number of faces present in one image.

These features were also fed into an decision tree classifier.

Speech and music were classified by the same speaker identification code as in the 2001 TREC video track.

‘Manual’ video retrieval with classification pseudo-relevance feedback

Example-based image retrieval task has been studied for many years. The task requires the image search engine to find the set of images from a given image collection that is similar to the given query image. Traditional methods for content-based image retrieval are based on a vector model. These methods represent an image as a set of features and the difference between two images is measured through a (usually Euclidean) distance between their feature vectors. While there have been no large-scale, standardized evaluations of image retrieval systems, most image retrieval systems are based on features such as color, texture, and shape that are extracted from the image pixels.

In our system two kinds of low-level features are used for finding similar images: color features and texture features. The color features are the cumulative color histograms for each separate color channel, where the three channels are derived from the HSV color space. We use 16 bins for hue and 6 bins for both saturation and value. We generate a texture feature for each subblock of a 3*3 image tessellation. The texture features are obtained through the convolution of the subblock with various Gabor Filters. In our implementation, 6 angles are used and each filter output is quantized into 16 bins. We compute a histogram for each filter and generate their central and second-order moments as the texture features. We concatenate all the features into a longer feature vector for every image; i.e. one vector for all color features and one vector for all texture features. We use a simple nearest neighbor (NN) image matching algorithm on both color and texture to produce the initial similarity results. In a preprocessing step, each element of the feature vectors is scaled by the covariance of its dimension. We adopted the Euclidean distance as the similarity measure between two images.

Although nearest neighbor search is the most straightforward approach to finding the matching images, it suffers from two major drawbacks. First, irrelevant features in the vector are given equal weight to important features, and thus retrieval accuracy will hurt decrease dramatically. Feature selection is therefore a necessary step prior to computing the nearest neighbor images. In theory, relevance feedback, through re-weighting and query refinement, is a powerful tool to refine the feature weighting so as to provide more accurate results. However, it is impossible to obtain the user judgment information in most automatic retrieval tasks. A second negative aspect is the unjustified distance function. Since an appropriate distance measure is a function of both the characteristics of the dataset and of the queries, a simple Euclidean distance function is unlikely to work for all the queries and images. Another concern is the normalization of the different dimension of a feature vector. To mitigate all these issues, we propose a classification-based pseudo-relevance feedback approach to refine the initial retrieval result. Support Vector Machines (SVMs) are used as our basic classifier mechanism, since SVMs are known to yield good generalization performance compared to other classification algorithms.

The basic idea for this approach is to augment the retrieval results by incorporating the classification output value through Pseudo-Relevance Feedback (PRF). The input data for the classifier is based on the information provided by our initial retrieval results. Standard PRF methods, which originated in the text information retrieval community, utilize the top-ranked documents as positive examples to improve the accuracy. The idea is to re-weight the words in the document feature vector based on the words in the top ranked documents, which are assumed to be positive examples. However, due to the poor initial performance of current video retrieval system, even the very top-ranked results are not always the correct ones that meet the users' information need. Unlike in text retrieval methods, it is more appropriate to make use of the *lowest* ranked documents in the collection after the initial search, which are more likely to be the negative examples. Therefore, we construct a classifier where the positive data are the query image examples and the negative data are sampled from the least confident image examples in the initial retrieval results.

Since the number of positive examples in our retrieval task is always much smaller than the number of the negative examples, we cast the problem into the imbalanced dataset classification framework. To sample more negative examples but achieve an overall balanced distribution of negative and positive examples in the classifier training set, we apply an ensemble of SVMs to tackle the rare class problem. The overall procedure can be summarized as follows,

1. Generate the initial classification results by nearest neighbor retrieval for all the images in the collection.
2. Choose all the query images as positive data. Denote the number of query images as *m*.
3. Construct a negative sub-collection based on the initial retrieval results, which are defined by the lowest 10% of the retrieved data from the collection. We sample *k* groups of negative data from the negative sub-collection, where each group contains *m* query images. Combine each group of negative data and all the positive data as a training set.
4. Build a classifier from each training set to produce new relevant score for any images *x* $f_i(x)(1 \le i \le k)$, where *i* is the index of training set
5. Combine the results in form of logistic regression, which is

$$P(+|x) = \frac{\exp(\beta_0 + \sum_{i=1}^{k} \beta_i f_i(x))}{1 + \exp(\beta_0 + \sum_{i=1}^{k} \beta_i f_i(x))}$$

In our system, we simply set β_0 as 0, $\beta_i(1 \le i \le k)$ as equal values.

Our approach presented here utilizes the collection distribution knowledge to refine the final result. Due to the good generalization ability of the SVM algorithm, the most relevant features are selected automatically. Also the approach yields a better distance function based on the probability estimation compared with the simple Euclidean distance.

Combination of multiple agents

As the first step to integrate different types of agents, all the relevance scores of the agents are converted into posterior probability. For each agent other than the classification-based PRF agent, the posterior probability is generated by a linear transformation of their rank and scaled to the range of [0, 1]. All these posterior probabilities are simply linear combinations as follows:

$$Score = a_I(b_c P_{color}(+|x) + b_t P_{texture}(+|x) + b_{PRF} P_{PRF}(+|x)) + a_T P_{text}(+|x) + a_m P_{movie}(+|x)$$

where a_I, a_T, a_m is the weight for image agent, text agent, movie information agent respectively, which are set to be 1, 1, 0.2. b_c, b_t, b_{PRF} are the weights for the three search agents for image retrieval: NN on color, NN on texture and classification PRF, which are either set to be 0 or 1 in our contrastive experiments reported below.

Speech Recognition

The audio processing component of our video retrieval system splits the audio track from the MPEG-1 encoded video file, and decodes the audio and downsamples it to 16kHz, 16bit samples. These samples are then passed to a speech recognizer. The speech recognition system we used for these experiments is a state-of-the-art large vocabulary, speaker independent speech recognizer. For the purposes of this evaluation, a 64000-word language model derived from a large corpus of broadcast news transcripts was used. Previous experiments had shown the word error rate on this type of mixed documentary-style data with frequent overlap of music and speech to be 35 – 40%.

Text Retrieval

All retrieval of textual material was done using the OKAPI formula. The exact formula for the Okapi method is shown in Equation (1)

$$Sim(Q,D) = \sum_{qw \in Q} \left\{ \frac{tf(qw,D)\log(\frac{N - df(qw) + 0.5}{df(qw) + 0.5})}{0.5 + 1.5\frac{|D|}{avg\_dl} + tf(qw,D)} \right\} \qquad (1)$$

where *tf*(*qw*,*D*) is the term frequency of word *qw* in document *D*, *df*(*qw*) is the document frequency for the word *qw* and *avg_dl* is the average document length for all the documents in the collection.

Results

We report our results in terms of mean average precision in this section, as shown in Table 1. Four different combination of the retrieval agents are compared in this table, including the combination of text agents (Text), movie agents (Movie), nearest neighbor on color (Color), nearest neighbor on texture (Texture) and classification-based PRF (Classification). The results show a significant increase in retrieval quality using classification-base PRF technique. While the text information from the speech transcript accounts for the largest proportion of the mean average precision (0.0658), only a minimal gain was observed in the mean average precision when the 'movie title' and abstract were also searched (0.0724) in addition to the speech transcripts. The image retrieval component provided further improvements in the scores to a mean average precision of 0.1046. Finally, the PRF technique managed to boost the mean average precision to the final mean average precision score of 0.1124.

Approach	Precision	Recall	Mean Average Precision
Text only (ASR)	0.0348	0.1445	0.0658
Text + Movie information (Abstract and Title)	0.0348	0.1445	0.0724
Text + Movie + Image retrieval (Color + Texture)	0.0892	0.220	0.1046
Text + Movie + Color + Texture + PRF Classification	0.0924	0.216	0.1124

Table 1 Video Retrieval Results on the 25 queries of the 2003 TREC video track evaluation.

Interactive Video Retrieval

For the 2002 TREC video track interactive condition, we used the basic Informedia Digital Video Library system, as in the 2001 TREC Video TREC. A few refinements to the interface are discussed and illustrated below.

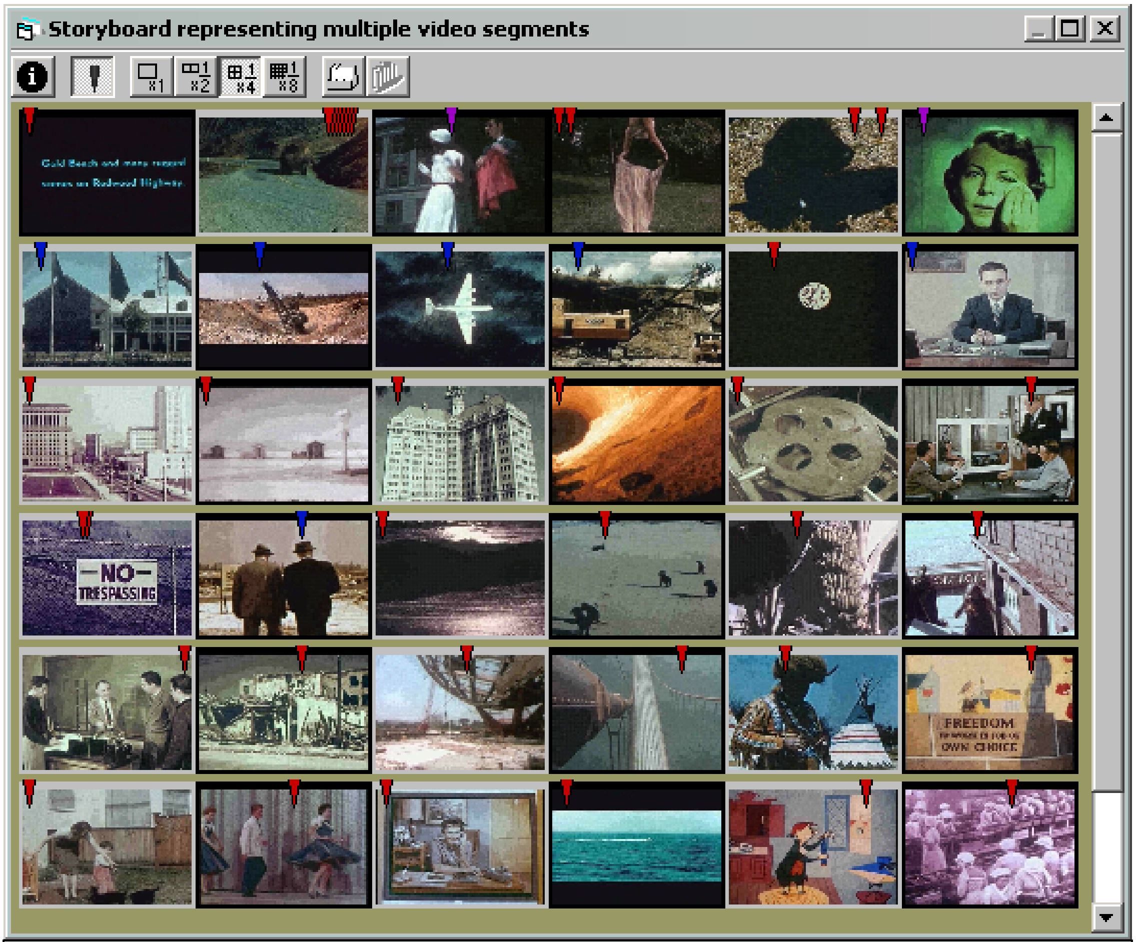


Figure 1. Multi-document storyboards combine all shots from highly relevant segments into one display.

Since IDVLS was designed to return ‘stories’, which can encompass multiple shots as retrieval results, we modified the interface to allow a shot-based presentation of the results which we called “*Multiple document storyboards*”. The text was retrieved in roughly 3-minute story chunks, and all shots for that story were presented to the user. A storyboard display, which concatenated the top N relevant stories and their shots, was used [Figure]. Thus a user could visually scan for relevant images from a fairly large storyboard display of the top relevant stories and their shots. Selecting a shot as relevant placed this shot onto an answer set display, which could again be edited before final submission [Figure].

Because of the large number of shots on the result storyboard, we placed the resolution of the keyframe size and the layout under user control. Thus a user can shrink or enlarge the size of the keyframes displayed on the storyboard, depending on the desire to visually inspect the keyframes more closely, or to view the complete set. The size of the window, and the total number of results displayed could also be modified. We found that the query context plays a key role in filtering image sets to manageable sizes. The TREC 2002 image feature set offered filtering capabilities for the classified categories of indoor, outdoor, faces, people, etc. The user interface provided for a display of the classified feature values for every shot [Figure]. The user was also able to control the threshold values for each of the feature categories. This enabled the display to be more manageable by filtering out shots that were more likely to

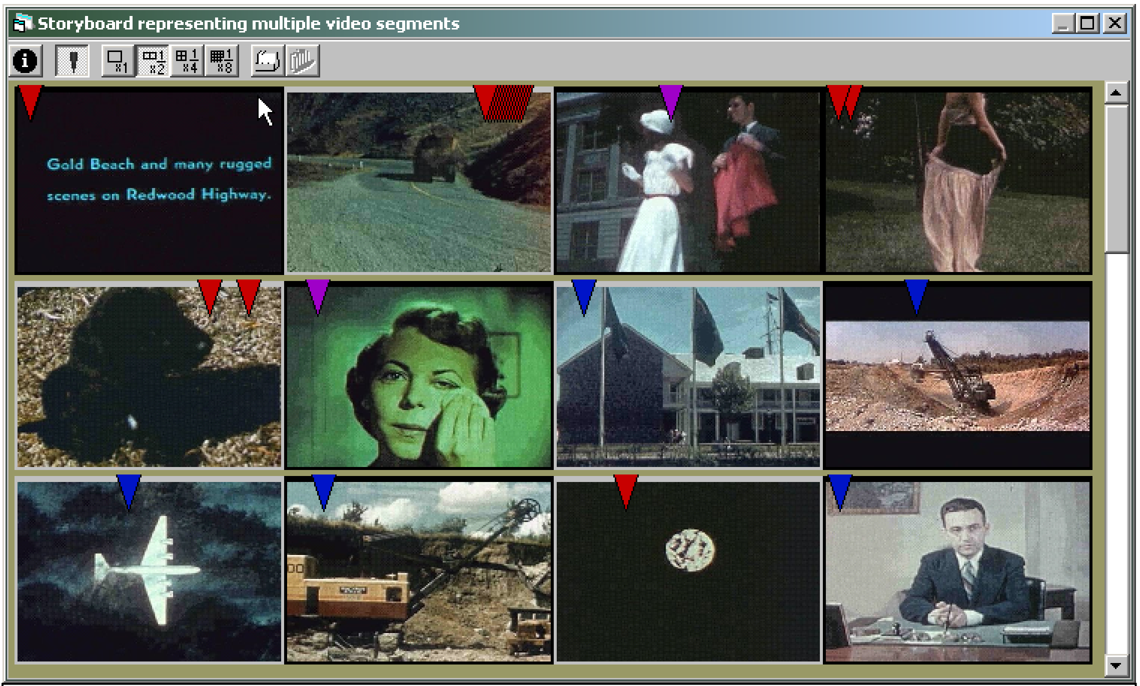
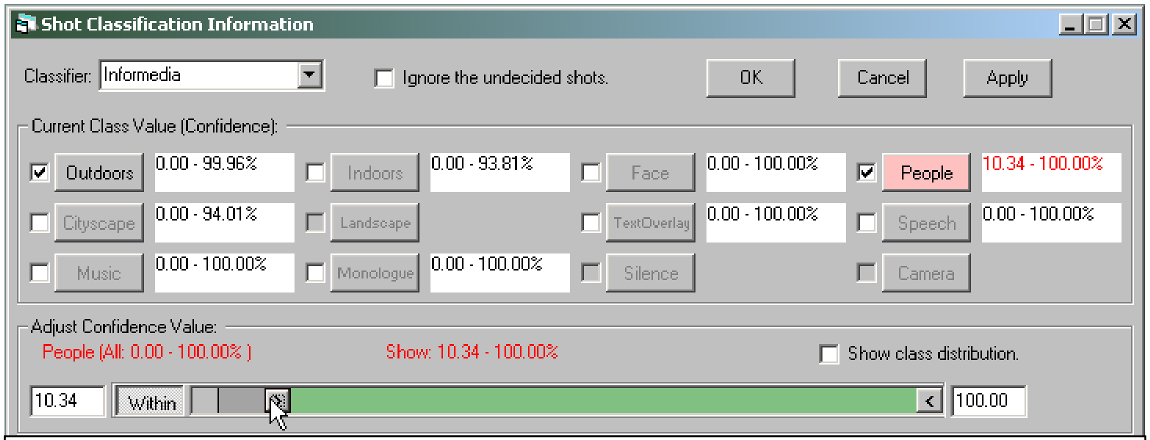

Storyboard representing multiple video segments
Gold Beach and many rugged scenes on Redwood Highway.

Figure 2. Resolution and layout of the storyboard can be modified by the user.

be feature X, and unlikely to be feature Y, depending on the query context. Since the display showed the number of active results, and provided direct feedback on the distribution of the data, the large number of irrelevant shot could easily be filtered down to a manageable number, that was then visually scanned by the user.

The multi-document storyboard facilitated quick inspection of many images. A first-order filtering by query text provided an initial set of images that constituted potential results. The multi-document storyboard based on 3-minute segments and shots enabled the user to find relevant shots, which were temporally near shots where query-words had been matched. The keyframe ordering by video segment and time useful. The classified shot features were useful for filtering, but needed to be manually adjusted depending on the particular queries. Users were able to drill-down to details, going from keyframe images to observing video, which was often necessary to eliminate uncertainty that could not be resolved by looking at a still image frame.



Shot Classification Information

Classifier: Informedia   □ Ignore the undecided shots.   OK   Cancel   Apply

Current Class Value (Confidence):

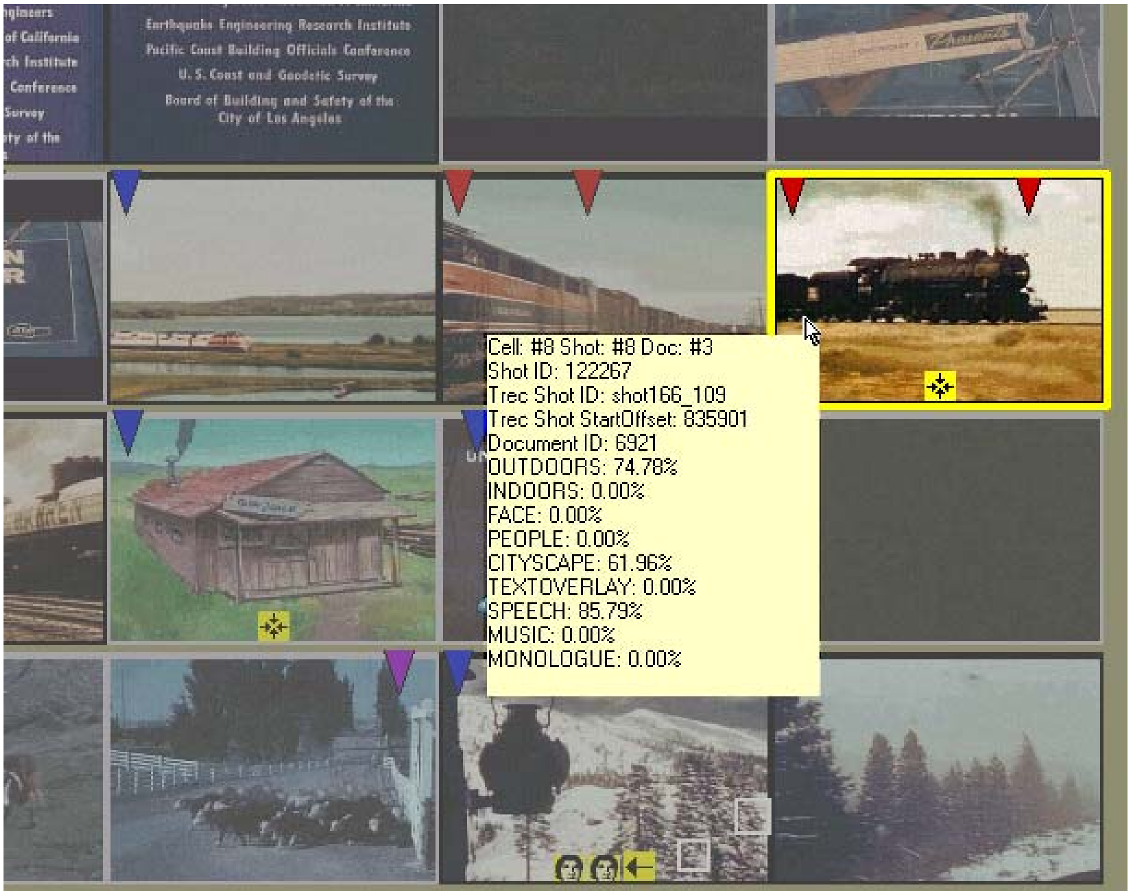
☑ Outdoors 0.00 - 99.96%	□ Indoors 0.00 - 93.81%	□ Face 0.00 - 100.00%	☑ People 10.34 - 100.00%
□ Cityscape 0.00 - 94.01%	□ Landscape	□ TextOverlay 0.00 - 100.00%	□ Speech 0.00 - 100.00%
□ Music 0.00 - 100.00%	□ Monologue 0.00 - 100.00%	□ Silence	□ Camera

Adjust Confidence Value:

People (All: 0.00 - 100.00%)   Show: 10.34 - 100.00%   □ Show class distribution.

10.34   Within   100.00

Figure 3. Users can filter shots based on thresholds in any feature classification category.



Cell: #8 Shot: #8 Doc: #3
Shot ID: 122267
Trec Shot ID: shot166_109
Trec Shot StartOffset: 835901
Document ID: 6921
OUTDOORS: 74.78%
INDOORS: 0.00%
FACE: 0.00%
PEOPLE: 0.00%
CITYSCAPE: 61.96%
TEXTOVERLAY: 0.00%
SPEECH: 85.79%
MUSIC: 0.00%
MONOLOGUE: 0.00%

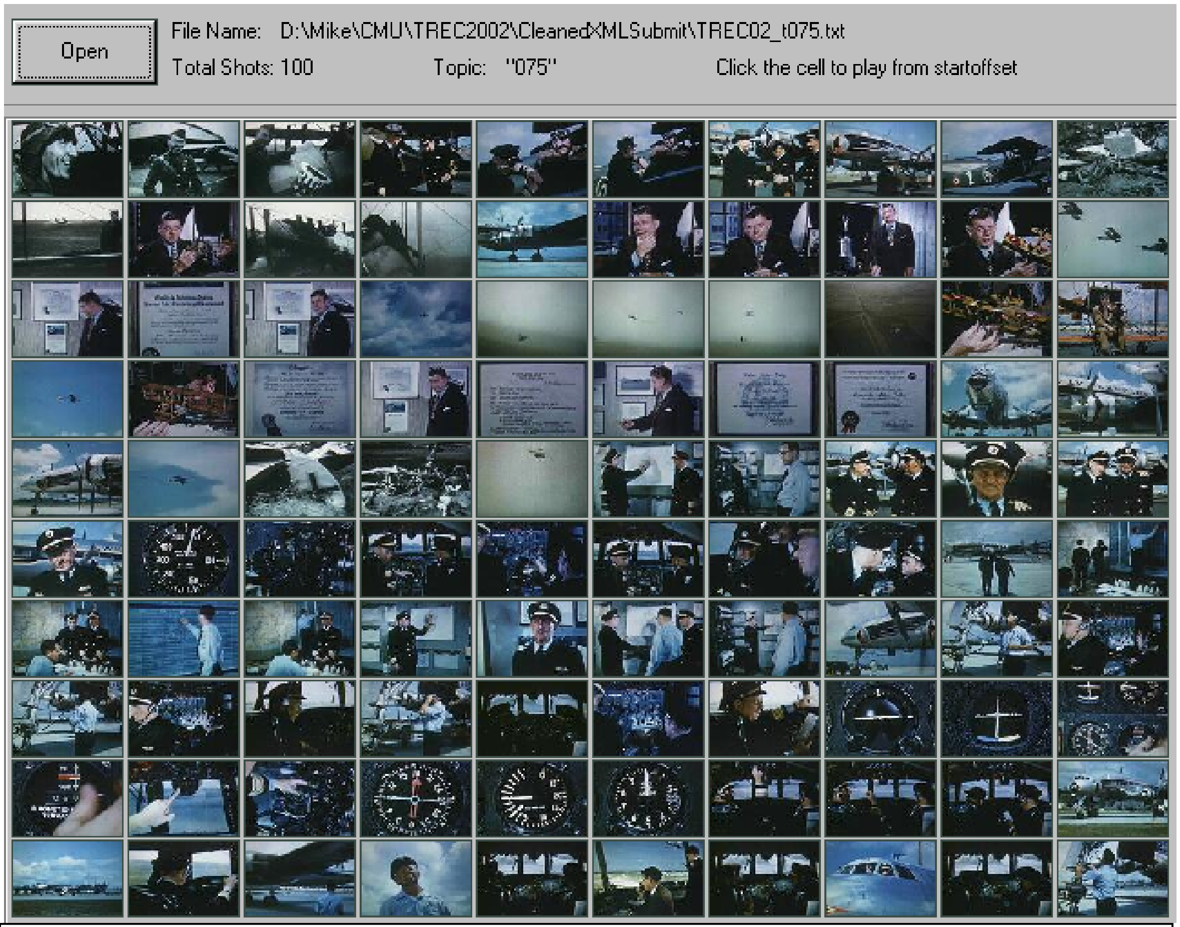
Earthquake Engineering Research Institute
Pacific Coast Building Officials Conference
U. S. Coast and Geodetic Survey
Board of Building and Safety of the City of Los Angeles

Figure 4. Feature classification statistics are accessible for any shot.

References

1. Hafner, J. Sawhney, H.S. Equitz, W. Flickner, M. and Niblack, W. "Efficient Color Histogram Indexing for Quadratic Form Distance," IEEE Trans. Pattern Analysis and Machine Intelligence, 17(7), pp. 729-736, July, 1995.
2. Robertson S.E., et al.. Okapi at TREC-4. In The Fourth Text Retrieval Conference (TREC-4). 1993.
3. Sato, T., Kanade, T., Hughes, E., and Smith, M. Video OCR for Digital News Archive. In *Proc. Workshop on Content-Based Access of Image and Video Databases.* (Los Alamitos, CA, Jan 1998), 52-60.
4. Singh, R., Seltzer, M.L., Raj, B., and Stern, R.M. "Speech in Noisy Environments: Robust Automatic Segmentation, Feature Extraction, and Hypothesis Combination," *IEEE Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, UT, May, 2001.
5. A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain, "Content-Based Image Retrieval at the End of the Early Years," IEEE Trans. Pattern Analysis and Machine Intelligence, 22(12), pp. 1349-1380, December, 2000.
6. Swain M.J. and Ballard, B.H. "Color Indexing," Int'l J. Computer Vision, vol. 7, no. 1, pp. 11-32, 1991.
7. Tague-Sutcliffe, J.M., "The Pragmatics of Information Retrieval Experimentation, revised," Information Processing and Management, 28, 467-490, 1992.
8. TREC 2002 National Institute of Standards and Technology, Text REtrieval Conference web page, http://www.trec.nist.gov/, 2002
9. The TREC Video Retrieval Track Home Page, http://www-nlpir.nist.gov/projects/trecvid/

10. Wactlar, H.D., Christel, M.G., Gong, Y., and Hauptmann, A.G. "Lessons Learned from the Creation and Deployment of a Terabyte Digital Video Library", *IEEE Computer* **32**(2): 66-73.
11. *Informedia Digital Video Library Project Web Site*. Carnegie Mellon University, Pittsburgh, PA, USA. URL http://www.informedia.cs.cmu.edu
12. A. Del Bimbo " Visual Information Retrieval", Morgan Kaufmann Ed., San Francisco, USA, 1999
13. Mojsilovic, J. Kovacevic, J. Hu, R.J. Safranek, and S.K. Ganapathy, "Matching and Retrieval Based on the Vocabulary and Grammar of Color Patterns," IEEE Trans. Image Processing, 9(1), pp. 38-54, 2000
14. Gong, Y. *Intelligent Image Databases: Toward Advanced Image Retrieval*. Kluwer Academic Publishers: Hingham, MA.
15. A. Vailaya, A. Jain, and H.J. Zhang, "On image classification: city images vs. landscapes", *Pattern Recognition*, 31(12)(1998) 1921-1935.
16. Y. Li and L. G. Shapiro. "Consistent Line Clusters for Building Recognition in CBIR," International Conference on Pattern Recognition, August 2002
17. M. Szummer, R. W. Picard, **Indoor-Outdoor Image Classification,** IEEE International Workshop on Content-based Access of Image and Video Databases, in conjunction with ICCV'98
18. Q. Iqbal and J. K. Aggarwal, "Applying perceptual grouping to content-based image retrieval: Building images," in IEEE International Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, vol. 1, pp. 42--48, June 1999.
19. H. Schneiderman, T. Kanade. "Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 45-51. 1998. Santa Barbara, CA.
20. A. G. Hauptmann, R. Jin, N. Papernick, D. Ng, Y. Qi, R. Houghton, and S. Thornton: Video Retrieval with the Informedia Digital Video Library System. The 10th Text REtrieval Conference (TREC 2001), National Institute of Standards and Technology (NIST) Gaithersburg, Maryland, 2001.


File Name: D:\Mike\CMU\TREC2002\CleanedXMLSubmit\TREC02_t075.txt

Total Shots: 100 Topic: "075" Click the cell to play from startoffset

Figure 5. The final result set can be reviewed and edited before submission.