VIDEO CLASSIFICATION USING TRANSFORM COEFFICIENTS

Andreas Girgensohn and Jonathan Foote

FX Palo Alto Laboratory
3400 Hillview Avenue
Palo Alto, CA 94304
{andreasg, foote}@pal.xerox.com

ABSTRACT

This paper describes techniques for classifying video frames using statistical models of reduced DCT or Hadamard transform coefficients. When decimated in time and reduced using truncation or principal component analysis, transform coefficients taken across an entire frame image allow rapid modeling, segmentation, and similarity calculation. Unlike color-histogram metrics, this approach models image composition and works on grayscale images. Modeling the statistics of the transformed video frame images gives a likelihood measure that allows video to be segmented, classified, and ranked by similarity for retrieval. Experiments are presented that show an 87% correct classification rate for different classes. Applications are presented including a content-aware video browser.

1. INTRODUCTION

Automatic classification of video is useful for a wide variety of applications, for example, automatic segmentation and content-based retrieval. Applications using automatic classification can support users in browsing and retrieving digitized video [1]. Other applications include identifying close-up video frames before running a computationally expensive face recognizer.

In this paper, we describe methods for classifying video frame images using statistical models. Images are transformed using a discrete cosine transform (DCT) or a Hadamard transform (HT) [1]. For accurate modeling, we reduce the dimensionality of the resulting coefficients by truncation or principal component analysis (PCA). Video frame classification consists of two steps: in the first step, model parameters for a particular class are estimated from a number of example training frames. In the second step, unseen test data is compared with each class model. Using a Gaussian model, the likelihood that the model generates the unknown data can be computed. This can be used as a distance measure to judge how similar the test data is to the example training data. Or, if a number of class models are used, the one with the highest probability of generating the unknown data can indicate the class of the unknown frame or image. Gaussian models can capture the characteristic composition and shape of an image class, while modeling the variation due to motion or lighting differences. Unlike many similarity measures based on color histograms, this approach models image composition features and works on black-and-white as well as color sources.

To assess our approach, we conducted a number of experiments on a corpus of videotaped staff meetings. We categorized the video shots into six categories and divided the corpus into a training and a test set. The next section, discusses related work, while following sections describe our approach in more detail and present experimental procedure and results.

2. RELATED WORK

Swanberg *et al*. [10] analyze individual image frames with a combination of color histogram and pixel-domain template matching. Zhang *et al*. [12] use color histograms, as well as motion and texture features, to segment video. Several researchers [3, 12] have looked at indexing video in the compressed domain, using the sub-block and motion information already present in MPEG-encoded video. Mohan [9] has done video shot matching by comparing time sequences of rank-based frame “fingerprints.” Many image retrieval systems use statistics of block-transform coefficients [6].

The exception to block transforms seems to seems to be wavelet approaches, which typically analyze an entire image using a wavelet basis (such as the Haar [7]). Quantizing and truncating higher-order coefficients reduces the dimensionality, while the similarity distance measure is just a count of bitwise similarity [7]. This approach apparently has not been used with more traditional transforms such as the DCT or Hadamard, nor has it been applied to video. Neural-network and decision-tree approaches have been used to classify images, but in the spatial (pixel intensity) domain [8]. A radial projection of FFT coefficients has been used as a signature for image retrieval [5].

Hidden Markov models have been used to segment video, but not on transform features, which is surprising given the ubiquity of this approach in the speech recognition domain. One approach uses color histogram features and motion cues [4]. Another approach uses a Markov-like finite state machine on principal components of pixel intensities [11].

3. VIDEO CLASSIFICATION

Each frame image is transformed, using either the DCT or HT. For many applications, a full video frame rate is not necessary, and frames can be decimated in time such that only one of several frames is transformed. This can reduce storage costs and computation times dramatically. The transform is applied to the frame image as a whole, rather than to small sub-blocks as is common for image compression. The transformed data is then reduced by discarding less significant information. This can be done using one of a number of techniques, for example, truncation, principal component analysis (PCA), or linear discriminant analysis (LDA). For this application, PCA works especially well, as it tends to decorrelate feature dimensions, thus the data better matches the diagonal-covariance assumption of the Gaussian models used in the Section 4 experiments. This results in a com-

pact feature vector (the reduced coefficients) for each frame. This representation is appropriate for classification, because frames of similar images will have similar features.

Given sufficient data reduction, it is simple to train a classifier to discriminate between typical meeting video scenes such as presentation slides, presenter, or audience. Besides our domain of meeting videos, this approach should work well whenever images in a particular class have a similar composition, for example shots of a news anchor.

Given feature data, an image class can be modeled with a multidimensional Gaussian distribution. We assume a diagonal covariance matrix, i.e. the off-diagonal elements are zero so the model will be robust in high dimensions (we present results for dimensionality up to 1000). To model a class using Gaussian models, the mean and covariance across a set of training images is computed. If single-mixture Gaussian models are used, this can be rapidly done in one pass over the data. More sophisticated models can use Gaussian mixtures, given the well-known Expectation-Maximization algorithm to estimate the multiple parameters and mixture weights, though this requires iteration. For this reason, we use single-mixture Gaussian models which can be computed rapidly on the fly. The log-likelihood alone is a useful measure of similarity to a particular class model, and can be used directly in applications such as the video browser of [1]. Given an unknown frame and several models, the unknown frame can be classified as to which model would produce it with the maximum likelihood.

4. EXPERIMENTS

4.1 Setup

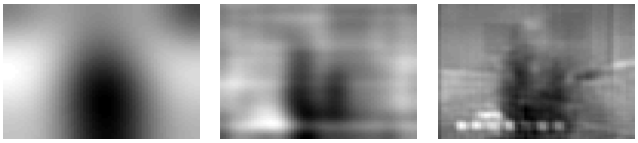
Video classification experiments were performed on a corpus of video-recorded staff meetings held over a six-month period. Each video is produced by a camera operator, who switches between video from three cameras with controllable pan/tilt/zoom, and the video signals from a PC and rostrum camera. The latter device allows presentation graphics such as transparencies and opaque materials to be displayed on a rear-projection screen. Thus video shots typically consist of presenters, audience shots, and presentation graphics such as PowerPoint slides or transparencies. The resultant video is MPEG-1 encoded and stored on a server.

There were 21 meeting videos in our corpus, for a total of more than 13 hours of video. The corpus was arbitrarily segmented into

Shot Category	Training Data	Test Data
slides	16,113	12,969
longsw	9,102	5,273
longsb	6,183	5,208
crowd	3,488	1,806
figonw	3,894	1,806
figonb	5,754	1,003
not categorized	13,287	10,947
Total	**57,821**	**39,047**

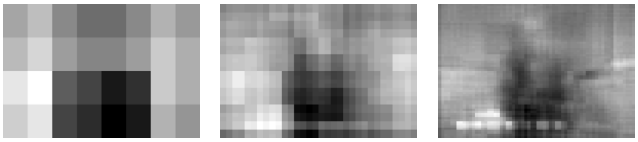
Table 1. Number of training and test frames (at 0.5 Hz)

Example training frames



mean of DCT Gaussian models

10 100 1000



mean of HT Gaussian models

Figure 1. Example training frames and model means.

testing and training segments by taking alternate meeting videos. We labeled both testing and training data into six classes of Table 1, which also shows the number of frames in each training and test set. A significant amount of data did not fit into any category and was left unlabeled. We chose six classes to represent presentation graphics, (slides), long shots of the projection screen both lit (longsw) and unlit (longsb), long shots of the audience (crowd) and medium close-ups of human figures on light (figonw) and dark (figonb) backgrounds. When a single category (such as screen shots) had significantly different modes (such as lit and unlit) we used a separate model for each mode. This ensured a better match with our single-Gaussian models, though another approach might use a Gaussian mixture to model the combined classes. Different models can be combined when they are intended to model the same logical class For example, we combine the figonw and figonb classes when presenting classification results, as the background color doesn't matter when the intent is to find human figures.

MPEG frames taken at ½-second intervals were decoded and reduced to 64×64 grayscale intensity images. The resulting frame images were DCT and HT coded. Both the coefficients with the highest variance (rank) and the most important principal components were selected as features. Gaussian models were trained on the training set using a variable number of dimensions between 1 and 1000. Figure 1 shows samples for one of the feature categories (figonw). That category consists of close-ups of people against a lighter (white) background[1]. The mean and covariance were trained using the highest-variance DCT and HT coefficients. Each model has been imaged by inverse-transforming the

[1]Note how the images for this class are highly variable in camera angle, lighting, and position, perhaps more than images of a typical news anchor.

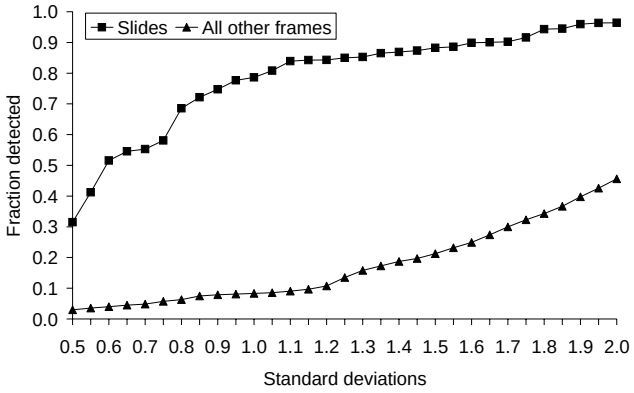


Figure 2. Slide detection performance vs. threshold.

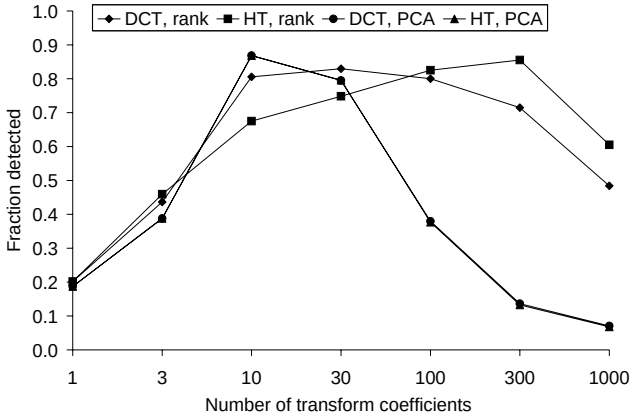


Figure 3. Classification vs. number of coefficients.

mean with the discarded coefficients set to zero. Though the covariance is not shown, it is clear the mean captures the major feature—the dark central figure—from the training data.

5. RESULTS

Thresholding the likelihood at a multiple of the standard deviation (from the covariance $|\Sigma|^{1/2}$) has shown to be quite effective in detecting class membership. Such a threshold is also fairly independent of the number of coefficients used. Figure 2 shows how the slide detection rate varies across different thresholds. The graph indicates that a threshold around 1.1 standard deviation results in an 84% correct slide recognition rate with few (9%) false positives. The likelihood, when normalized by the standard deviation, is useful by itself as an indication of a given frame's similarity to a class model, as discussed in Section 6.

All classes have similar detection rates, however, the number of false positives varies among the different classes. To show the different model results without thresholding, we used a maximum-likelihood approach to classify labeled test frames. Table 2 shows the results from using the 30 highest-variance DCT coefficients. The class **fig** is a superset of the combined **figonw** and **figonb** classes. Each column is the ground-truth label of the test frames; the rows indicate the fraction of the samples in the test set that are recognized as the row class. Non-zero off-diagonal elements represent classification errors. Columns sum to 1 as every labeled frame has a maximum-likelihood class even if different from the label.

To study the influence of the number of transform coefficients for the different transform methods, we computed the overall correctness, i.e., the fraction of samples that were recognized in the correct category. Figure 3 shows the results. It is interesting to note that the recognition distribution for the principal components of the DCT and HT is virtually identical. The best performance (87% correct) was achieved using 10 principal components. Without PCA, variance-ranked DCT coefficients peak at 30 whereas HT coefficients achieve a slightly higher accuracy at 300. Though the HT is often criticized for not preserving perceptual features as well as the DCT, it appears to work somewhat better here, perhaps because the rectilinear HT basis functions match image features (such as slides or walls) better than the sinusoidal DCT bases.

	slides	longsw	longsb	crowd	fig
slides	0.872	0.017	0.000	0.000	0.000
longsw	0.009	0.900	0.000	0.000	0.000
longsb	0.000	0.002	0.749	0.000	0.000
crowd	0.001	0.042	0.014	0.848	0.010
fig	0.118	0.039	0.237	0.152	0.990

Table 2. Frame classification results. Columns are the true label; rows are the model with highest likelihood.

6. APPLICATIONS

We have developed an application that uses video classification to help users find interesting passages in video [1]. It is not simple to determine whether a long video contains desired information without watching it in its entirety. Our intelligent media browser allows fine-grained access to video by taking advantage of the metadata extracted from the video (see Figure 4). A confidence score for a particular video is displayed graphically on a timeline. The confidence score gives valuable cues to interesting regions in the source stream by using the time axis for random-access into the source media stream. For example, the normalized log-likelihood of the slide model is displayed on the timeline of Figure 4. Two areas of high likelihood (confidence) are visible as the grey or black regions: these correspond to slide images in the video. Selecting a point or region on the time axis starts media playback from the corresponding time. Thus time intervals of high potential interest can be visually identified from the confidence display and easily reviewed without a linear search.

6.1 Further Applications

The experiments of Section 5 shows how a Gaussian classifier can detect video frames from a particular class in the context of a longer video. This can be used to segment shots, defined as a region of similar frames, from a longer video. This can provide useful index points, for example the beginning of a shot containing slides. In the other direction, if shots have been already located, for example using frame or color differences, a shot model can easily be trained on all the frames from that shot. This allows shots to be retrieved by similarity, because the covariance

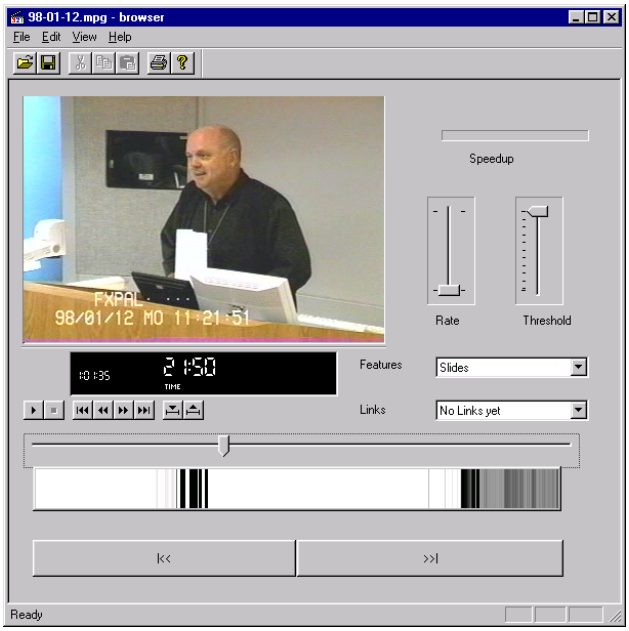


Figure 4. Browser displaying confidence score for slides.

will capture differences caused by motion or other changes. Keyframes to represent a given shot are easily found by finding the frame closest to the shot mean, using a likelihood distance metric. Because the number of coefficients that represent an image is extremely modest (as small as 10 per frame for the PCA features), it is possible to store the features alongside the video with virtually no overhead. Gaussian models are straightforward to compute, so models can be trained on-the-fly. This enables applications like interactive video retrieval, where the user could indicate the desired class, for example, by selecting a video region by dragging across the timeline. A model could be rapidly trained on the features for this region, and the similarity of a large video corpus could be rapidly computed. Regions of high likelihood in the corpus are regions that match the selected video well, and would serve as indexes into the corpus.

6.2 Application to Motion Analysis

Simple Gaussian models as above compute the mean or average of the training frames, and so lose any time-varying information associated with the video sequence. To capture dynamic or sequential information, models can be enhanced in a number of ways. By training models on the frame-to-frame difference or trend of the reduced features, time-varying effects such as motion or fades can be modeled. To find the similarity between video sequences, a correlation score can be computed by summing the frame-by-frame inner product of the two sequences. Similar sequences will have a large correlation. Dynamic programming can be used to find the best match between two sequences of dissimilar length. A better way of capturing dynamic events would be a hidden Markov model, using Gaussian mixtures to model feature output probabilities. Given the efficient training and recognition algorithms developed for speech recognition, this is a promising area for future investigation.

7. CONCLUSIONS

The experiments presented here demonstrate that statistical models of transform coefficients can rapidly classify video frames with low error rates. The computational simplicity and low storage requirements of this approach enable novel applications such as interactive video retrieval. We are exploring more sophisticated statistical models to increase the versatility of our approach.

8. ACKNOWLEDGMENTS

Thanks to John Doherty for producing the meeting videos in our corpus, and to Lynn Wilcox for comments on the manuscript.

9. REFERENCES

[1] J. Foote, J. Boreczky, A. Girgensohn, and L. Wilcox, “An Intelligent Media Browser using Automatic Multimodal Analysis,” in *Proc. ACM Multimedia*, Bristol, UK Sept. 1998.

[2] A. Rosenfield and A. Kak, *Digital Picture Processing*, Academic Press, 1982.

[3] F. Arman, A. Hsu, and M.-Y. Chiu, “Image Processing on Encoded Video Sequences”, *Multimedia Systems* (1994) Vol. 1, No. 5, pp. 211-219.

[4] J. Boreczky and L. Wilcox, “A Hidden Markov Model Framework for Video Segmentation Using Audio and Image Features”, in *Proc ICASSP ’98*, IEEE, May 1998, Seattle.

[5] A. Centennial and V. Lecce, “A FFT based Technique for Image Signature Generation,” in *Proc. SPIE Vol. 3022, Storage and Retrieval for Image and Video Databases V*, pp. 457-466, Feb. 1997.

[6] Y. S. Hsu, S. Prum, J. Kagel, and H. Andrews, “Pattern recognition Experiments in the Mandala/Cosine Domain,” *IEEE Trans. PAMI*, Vol. PAMI-5, No. 5, Sept. 1983.

[7] C. Jacobs, A. Finkelstein, D. Salesin, “Fast Multiresolution Image Querying.” In *Proc. SIGGRAPH ’95*, Los Angeles, CA, August 1995.

[8] B. Moghaddam, W. Wahid and A. Pentland, Beyond Eigenfaces: Probabilistic Matching for Face Recognition, Proc. *IEEE Intl. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 1998.

[9] R. Mohan, “Video Sequence Matching,” in *Proc ICASSP ’98*, IEEE, May 1998, Seattle.

[10] D. Swanberg, C.-F. Shu, and R. Jain, “Knowledge Guided Parsing in Video Databases,” in *Proc. SPIE Vol. 1908, Storage and Retrieval for Image and Video Databases*, pp. 13-24, Feb. 1993.

[11] A. Wilson, A. Bobick, and J. Cassell, “Recovering the Temporal Structure of Natural Gesture,” in *Proc. Second Int. Conf. on Automatic Face and Gesture Recognition*, Oct., 1996 (Also MIT Media Laboratory Technical Report No. 338).

[12] H.-J. Zhang, C.-Y. Low, S. Smoliar, and J.-H. Wu, “Video Parsing, Retrieval, and Browsing: an Integrated and Content-Based Solution,” in M. Maybury, ed., *Intelligent Multimedia Information Retrieval*, AAAI Press/MIT Press, 1997.