Classification of Video Events using 4-dimensional time-compressed Motion Features

Alexander Haubold
Department of Computer Science
Columbia University
New York, NY 10027
ahaubold@cs.columbia.edu

Milind Naphade
IBM Thomas J. Watson Research Center
Hawthorne, NY 10532
naphade@us.ibm.com

ABSTRACT

Among the various types of semantic concepts modeled, events pose the greatest challenge in terms of computational power needed to represent the event and accuracy that can be achieved in modeling it. We introduce a novel low-level visual feature that summarizes motion in a shot. This feature leverages motion vectors from MPEG-encoded video, and aggregates local motion vectors over time in a matrix, which we refer to as a motion image. The resulting motion image is representative of the overall motion in a video shot, having compressed the temporal dimension while preserving spatial ordering. Building motion models using this feature permits us to combine the power of discriminant modeling with the dynamics of the motion in video shots that cannot be accomplished by building generative models over a time series of motion features from multiple frames in the video shot. Evaluation of models built using several motion image features in the TRECVID 2005 dataset shows that use of this novel motion feature results an average improvement in concept detection performance by 140% over existing motion features. Furthermore, experiments also reveal that when this motion feature is combined with static feature representations of a single keyframe from the shot such as color and texture features, the fused detection results in an improvement between 4 to 12% over the fusion across the static features alone.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models.* I.2.6 [**Artificial Intelligence**]: Learning – *concept learning, parameter learning*. I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding – *motion, video analysis*.

General Terms

Algorithms, Performance, Design, Experimentation, Verification.

Keywords

MPEG motion vectors, motion features, TRECVID, LSCOM.

CIVR '07, July 9-11, 2007, Amsterdam, The Netherlands.
Copyright 2007 ACM 978-1-59593-733-9/07/0007…$5.00.

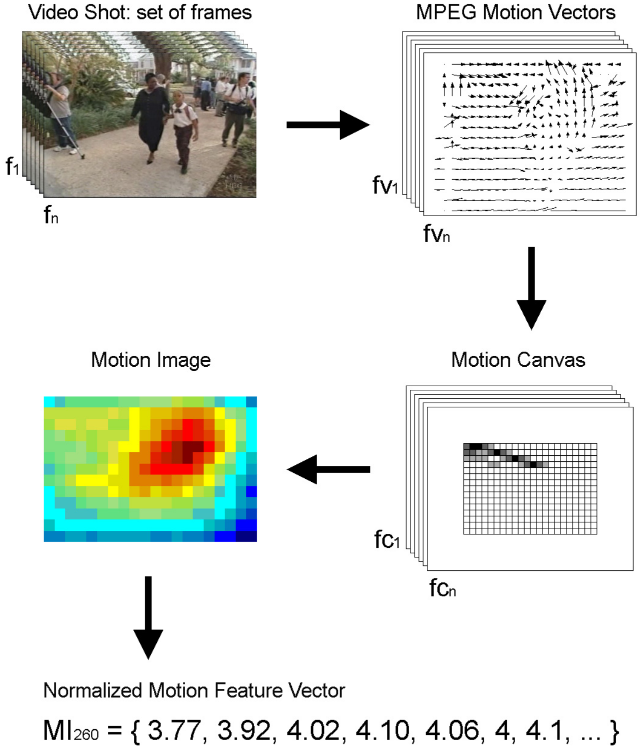
Video Shot: set of frames
f_1
f_n
MPEG Motion Vectors
fv_1
fv_n
Motion Image
Motion Canvas
fc_1
fc_n
Normalized Motion Feature Vector
MI_{260} = { 3.77, 3.92, 4.02, 4.10, 4.06, 4, 4.1, ... }

Figure 1. Overview of motion image extraction. Clock-wise: Frames in a video shot; MPEG motion vectors for P- and B-frames; motion image with compressed dimension of time; trimmed motion image to reduce noise along edges; 260-feature motion image vector.

1. INTRODUCTION

Semantic multimedia management is necessary for the effective and widespread utilization of multimedia repositories and realizing the potential that lies untapped in the rich multimodal information content. This challenge has driven researchers to devise new algorithms and systems that enable automatic or semi-automatic tagging of large-scale multimedia content with rich semantics. Detecting a predetermined set of semantic concepts that can act as semantic filters and aid in search, and manipulation is now a well-accepted approach for automatic indexing. These semantic concepts that are detected are mostly of the generic variety and can be categorized as belonging to sites, objects, people, events, etc. The TRECVID benchmark run by the National Institute of Standards and Technology has been evaluating the state of the art techniques in semantic concept detection [4]. A review of various techniques and their efficacy in concept detection can be found in Naphade et. al. [13]. Of the various concept types, events pose the greatest challenge on account of their complexity, difficulty in getting informative and discriminating representations as well as the limited success of temporal modeling and detection techniques.

Event modeling using temporal and dynamic graphical model is expensive. While effective in gesture and speech recognition, event modeling using graphical dynamic models including hidden Markov models of various flavors have tasted mixed success in modeling visual events. Use of hierarchical HMMs for combining modalities and modeling temporal video events includes [14, 11, 15, 9]. Despite this application of dynamic graphical modeling, the performance for event modeling and detection continues to be a challenge in scenarios where a very large number of training samples are not available. It is in situations like these that the need for event models that are built using discriminant classifiers is acute and the need for well designed features that can capture motion information of video shots into a small number of feature dimensions is required. Previous work includes the use of motion magnitude and motion direction histograms [14], which have been commonly used to combine the motion information of an entire shot or a series of frames into a single feature vector of smaller dimensionality. However, these features compress several dimensions of motion, including spatial location, time, and either direction or time. Ideally none of the five dimensions would be compressed in order to best represent motion patterns in a series of video shots. Motion textures [10] retain magnitude, direction, and time, but compress the two dimensions of space. Previous work on using motion for a variety of human-specific motion detection includes motion history images for detection of human motion in videos [16] and unsupervised learning of human action categories [17] to name a few. Event detection in video using motion has also been discussed recently in [18].

In this paper we investigate a new kind of motion feature that captures the dynamism over a series of frames (Figure 1) and is richer than motion magnitude and direction histograms. We then leverage well-known discriminant learning techniques in the form of support vector machines to train discriminant models of temporal events based on these features. We compare the performance of discriminant models built using the novel feature with that of those built using existing popular aggregate motion features. We then combine the detection based on this novel motion feature with that based on static features representing localized and global color and texture extracted from a representative keyframe per shot. Finally we evaluate detection performance on a number of LSCOM-lite concepts on the TRECVID 2005 corpus to evaluate performance (Table 1).

2. MOTION IMAGES

2.1 Overview

A single instance of motion is a three-dimensional quantity described by direction, magnitude, and time. MPEG video takes advantage of the observation that between groups of consecutive frames, portions of visual regions are shifted in relatively small

Table 1: The LSCOM-lite Lexicon [12] designed for the TRECVID 2005 Benchmark consists of more than 40 concepts spread across multiple concept-types such as objects, events, sites, etc. Of these, 39 concepts were annotated and made available for training models in TRECVID 2005. Highlighted are concepts, which are motion driven (with the exception of "Mountain". Marked in orange (dark) are concepts for which motion features improve detection significantly.

	Category	Concepts
Broadcast News	Setting/Scene/Site	Building
		Court
		Desert
		Meeting
		Mountain
		Office
		Outdoor
		Road
		Sky
		Snow
		Studio
		Urban
		Vegetation
		Waterscape
	People	Crowd
		Face
		Person
		Roles
		Government Leader
		Corporate Leader
		Police/Security
		Military
		Prisoner
	Objects	Airplane
		Animal
		Boat/Ship
		Bus
		Car
		Computer
		Flag-US
		Truck
		Vehicle
	Activities and Events	Explosion / Fire
		March
		Natural Disaster
		People Related
		Walk / Run
	Program Category	Entertainment
		Sports
		Weather
	Graphics	Charts
		Map

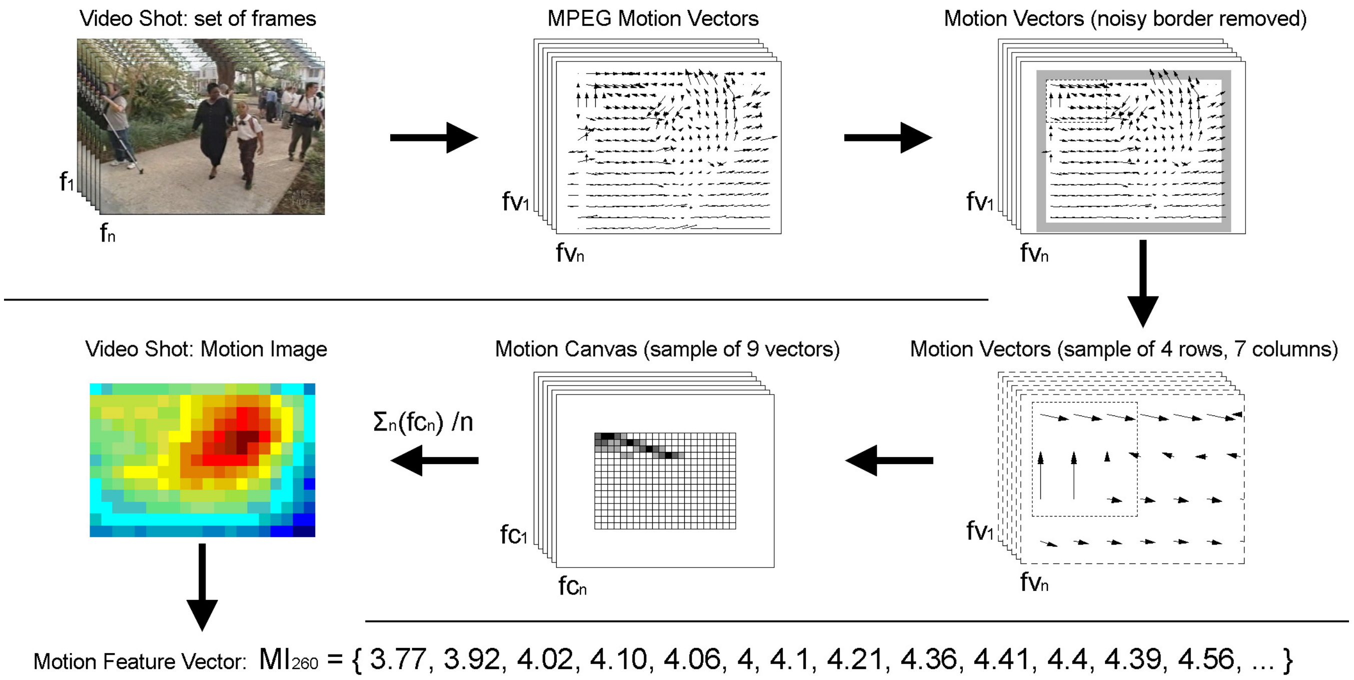
Video Shot: set of frames
f1
fn
MPEG Motion Vectors
fv1
fvn
Motion Vectors (noisy border removed)
fv1
fvn
Video Shot: Motion Image
Σn(fcn) /n
Motion Canvas (sample of 9 vectors)
fc1
fcn
Motion Vectors (sample of 4 rows, 7 columns)
fv1
fvn
Motion Feature Vector: MI260 = { 3.77, 3.92, 4.02, 4.10, 4.06, 4, 4.1, 4.21, 4.36, 4.41, 4.4, 4.39, 4.56, ... }

Figure 2. Detailed extraction of motion images. Motion vectors are added as intensity lines to a frame's representative motion canvas. Motion images are computed as the normalized sum of motion canvases.

vicinity. Motion vectors describe this movement between frames. Predictive and bi-directional MPEG video frames contain a two-dimensional grid of these motion vectors, consequently increasing the number of dimensions representing motion in MPEG video to five. Reduction in dimensionality to a small number of features while retaining all dimensions is a difficult problem. With variable duration of video shots, the temporal development of a single motion instance must be compressed into a constant quantity.

Motion images represent motion features by retaining direction and magnitude in a two dimensional space while aggregating the third dimension of time. The canvas of a motion image for a series of video frames is initially empty and assumes a size equal to the grid of motion vectors. Motion vectors are treated as lines with start and end points, which are added to this canvas as intensity pixels. The resulting motion image then describes the global motion pattern in a video shot. Once linearized, a motion image becomes a comparatively small feature vector.

2.2 Implementation

Motion vectors are present for all macroblocks (MB) in predictive (P) and bi-directional (B) frames of MPEG video. For intra-frames (I), which start a group of pictures (GOP) sequence of P and B frames, motion vectors have zero magnitude. Motion of a macroblock is defined as having a forward direction from a past reference frame or a backward direction from a future reference frame. Motion images do not encode vectors with start and end points, but instead the tentative direction of object motion in a video shot. In computing a motion image, vectors are added to the two-dimensional canvas as lines with constant intensity. This section describes the implementation outlined in Algorithm 1.

We extract motion vectors from P, and B frames of an MPEG video shot. A separate motion image is defined for each video shot. The image assumes the size equal to the number of macroblocks defined in the MPEG frames. An MPEG-1 video with 352 rows and 240 columns and single motion vectors per 16x16 pixel macroblock contains 15 MB rows and 22 MB columns. The corresponding motion image assumes a size of 330 pixels in 15 rows and 22 columns. Row and column indices of macroblock motion vectors are mapped directly to the xy space in the motion image.

A motion image is constructed by incrementally adding a frame's motion vectors to the motion image's canvas. Initially, the motion image for a shot is an empty canvas. Motion vectors from each video frame are added to the motion image as constant intensity lines. We apply Bresenham's fast line drawing algorithm [5] to compute the points of the line representing the motion vector, and add them to the aggregate sum of intensity values in the motion image. Points of the line, which exceed the canvas area of the motion image, are not considered. With the addition of motion vectors, areas of significant and characteristic motion are emphasized.

Magnitude of motion vectors defines the absolute pixel distance of displacement for 16x16 pixel macroblocks. Because motion images represent macroblocks as single pixels, magnitudes of motion vectors are scaled by a factor of 1/16 to match the reduced dimension. In a subsequent step, we scale each vector by some constant factor F, which represents the predicted future direction of that vector over F-many frames. This prediction approximates the tentative trajectory of the object represented by the macroblock. We stipulate that a factor of 15 frames, the equivalent of half a second, is reasonable for motion of typical objects without overestimation. For reasons of computational efficiency, we scale the vectors by a factor of 16, which cancels out the previous division. Consequently, we retain the original vector magnitude in the motion image as an amplified predictor of a macroblock's motion.

Features aggregated in a motion image are normalized by the number of frames in the video shot. A one-pixel border in the motion image, the equivalent to a 16-pixel border in the original video frame, is removed. We have observed that video content in this area is not indicative of shot content because production of highly edited video ensures that important visual material is centered in the frame. Motion estimation along the frame border is also very noise, mainly due to the presence of static. Finally, motion estimation in border regions is limited to three directions, and in corner regions only to two directions. The large concentration of horizontal and vertical motion vectors in these areas distorts the computed data in the motion image. In related work [7] motion vectors along the border are discarded for similar reasons.

The trimmed motion image reduces the number of pixels from 330 (15 * 22) to 260 (13 * 20), a reasonably sized feature vector for SVM. Figure 2 outlines the aggregation of motion vectors from several frames into one motion image. Figure 3 presents examples of video shots and their computed motion images for the concepts: Car, Mountain, Sports, Walking/Running, and Waterscape/Waterfront.

2.3 Classifiers from SVM

We have experimented with the use of support vector machines for building discriminant classifiers for semantic concepts for several years across multiple TRECVID cycles [1, 2, 3, 4] as have others [13] and it is a well established fact that SVM classifiers when used to build models for features individually and then combined through various fusion strategies (early vs. late fusion) provide the most consistent top performing systems evaluated. We will therefore experiment with the well-tested approach of building SVM models for each feature separately followed by a late fusion strategy that aggregates at the decision level using classification outputs of distance from the SVM hyperplanes mapped appropriately. Figure 4 shows the strategy used for optimizing the model for each feature across the several optimization parameters of the SVM.

We partitioned the development data set provided by NIST into the internal partitions for facilitating hierarchical processing experiments and selection by randomly assigning videos from the development set to each partition. All shots from videos in a partition were then assigned to that partition and one keyframe per shot was in turn used for feature extraction of static features whereas each entire shot was used for extracting motion features.

The training corpus consisted of 41000 annotated keyframes, one for each shot. A validation set of 7000 shots was used to tune the parameters and evaluate best-case performance for each feature (all motion features as well as all static features including color correlogram co-occurrence texture, grid based color moments and grid based wavelet texture). Based on this comparison the best motion feature was then selected for naive fusion with all the static features. The fusion in turn was evaluated on a third held out set of 7000 shots and results of fusion are reported on this third set referred to as the Selection set.

```
S = video shot (collection of video frames)
F = video frame
MI = motion image
MV = motion vector
($MV_{row}$, $MV_{col}$) = motion vector origin
($MV_h$, $MV_v$) = motion vector magnitude (horizontal, vertical)
MB = macro block
```

```
for each S
  // Initialize MI
  for r = 0 .. $MB_{rows}$ - 2
    for c = 0 .. $MB_{cols}$ - 2
      MI(r,c) = 0;
    }
  }
  // Compute motion image
  for each $F \in S$
    for each $MV \in F$
      drawLine(MI, $MV_{row}$, $MV_{col}$, $MV_h$, $MV_v$)
    }
  }
  // Normalize motion image
  for r = 0 .. $MB_{rows}$ - 2
    for c = 0 .. $MB_{cols}$ - 2
      MI(r,c) = MI(r,c) / |S|;
    }
  }
}

drawLine(MI, r, c, h, v) {
  // Add line to MI from position (r,c) with
  // magnitude (h,v). Each pixel of line is
  // added to existing values in MI canvas.
  // Points exceeding the image's area are
  // not considered. We use Bresenham's fast
  // line drawing algorithm [5].
}
```

Algorithm 1. Implementation of motion images for a collection of video frames (a video shot)

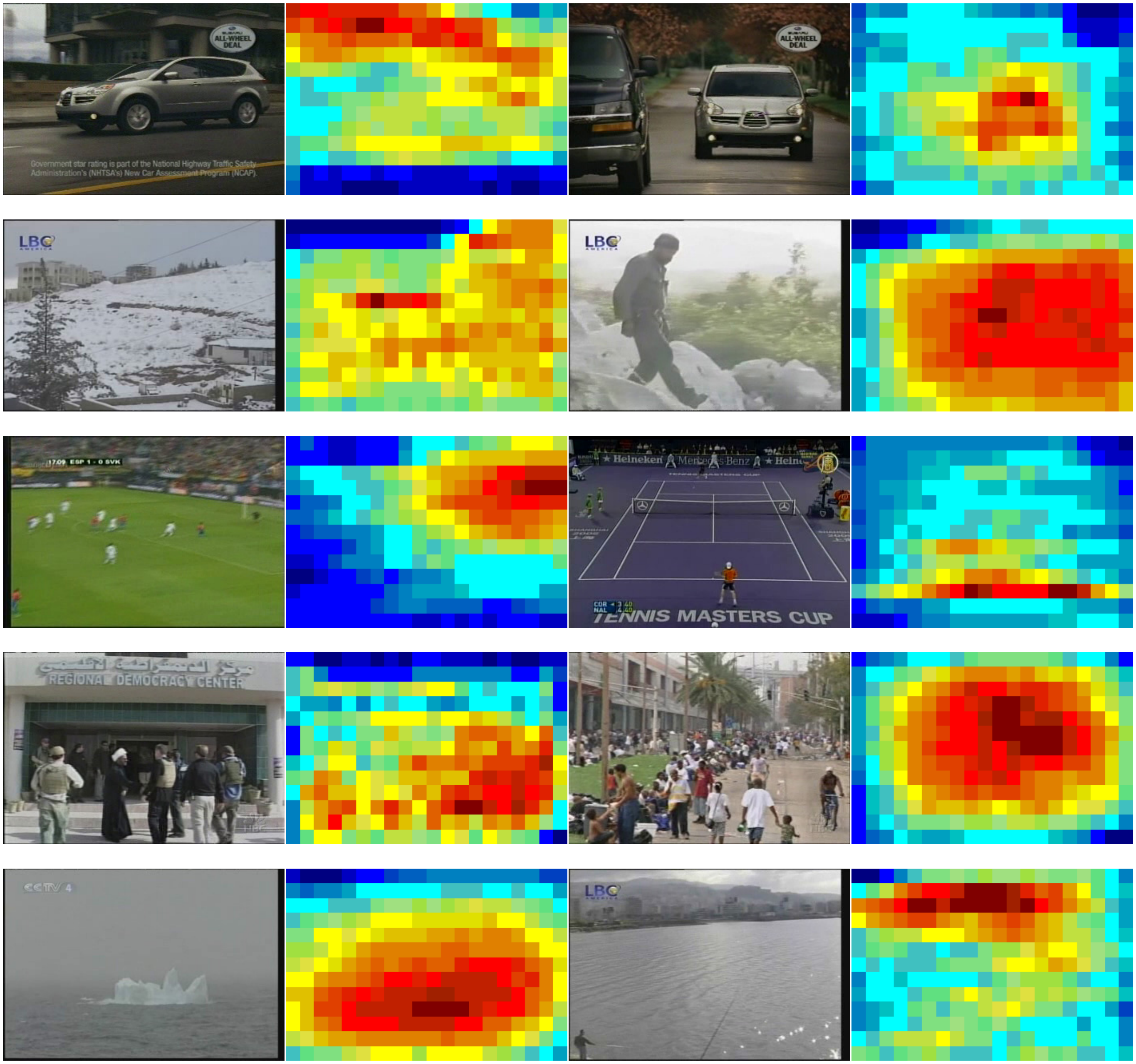
ALL-WHEEL DEAL
Government star rating is part of the National Highway Traffic Safety Administration's (NHTSA's) New Car Assessment Program (NCAP).
ALL-WHEEL DEAL
LBC
LBC
17:09 ESP 1 - 0 SVK
Heineken
Mercedes-Benz
TENNIS MASTERS CUP
REGIONAL DEMOCRACY CENTER
CCTV 4
LBC

Figure 3. Examples of video shots (representative keyframes) and their computed motion images. From top to bottom the concepts are: Car, Mountain, Sports, Walking/Running, and Waterscape/Waterfront. Motion images capture some of the characteristic movement in shots which exhibit these event concepts.

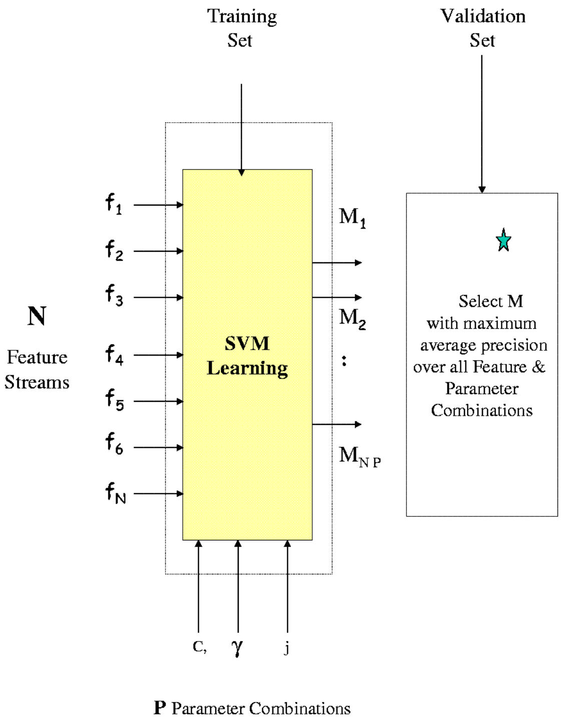

Figure 4: Various models for classifiers are built on individual features and are then fused to produce the final classifier.

3. EVALUATION

We evaluate the low-level motion image feature on selected concepts from the TRECVID 2005 dataset, most of which exhibit characteristic motion. The concepts include: Car, Sports, Walking/Running, Waterscape/Waterfront, and Mountain.

3.1 Comparison to other Motion Features

We have evaluated the effectiveness of motion images against three low-level motion features for concept detection in the TRECVID 2005 dataset. The various features have been generated under similar conditions. Specifically, we remove noisy features along the frame border, including the first and last row and first and last column. We find that motion images exceed average precision values for motion direction and motion magnitude histograms by up to 140%.

Motion direction histograms summarize observations of motion vector angles. We define 36 bins for this histogram, representing angle values of 0 to 350 in 10 degree increments. Motion vectors without magnitude are counted in the zero degree bin. Motion vectors from a series of video frames are aggregated in this histogram by angle, and the final value is normalized by the total number of motion vectors. Concept detection using the motion direction histogram alone resulted in an average precision of 11.5% over five selected concepts.

Motion magnitude histograms summarize observations of motion vector displacement. We define 30 bins for this histogram, representing absolute displacement of up to 30 pixels. Vectors with magnitude of more than 30 pixels are placed into the highest bin. The final histogram is again normalized by the total number of motion vectors. Concept detection on this feature resulted in an average precision of 7.5% over five selected concepts.

Motion direction and motion magnitude histograms retain only one dimension of motion and compress all others. We have also evaluated the combination of the two histograms, which retains two of the initial five dimensions, namely direction and magnitude. We define 35 bins for motion vector direction angle values of 10 to 350, and 30 orthogonal bins for vector magnitude. The resulting feature vector contains 1050 bins, improving coverage of dimensions, but also requiring substantially more computation for the SVM. Concept detection with this combined histogram yields as average precision of 10.7% over the five selected concepts.

Motion images with 260 features results in an average precision of 19% over the five selected concepts. This accounts for an improvement over motion direction, motion magnitude, and combined motion direction and magnitude histograms of 56.6%, 139.2%, and 67.7% respectively. See Table 2 and Figure 5.

3.2 TRECVID 2005

TRECVID chose the following 10 concepts for evaluation in the 2005 cycle: People, Walking/Running, Explosion or Fire, Map, US flag, Building, Exterior, Waterscape/Waterfront, Mountain, Prisoner, Sports, Car. Of these concepts we chose the 5 for which there might be some help from motion features: Car, Mountain, Sports, Walking/Running, Waterscape/Waterfront.We have evaluated various features for concept detection of five selected concepts and we report on the improvement of average precision values by additionally fusing motion image detection results.

The system extracts the following static features or visual descriptors from each keyframe in a shot. These features have been chosen based on years of experimentation and typically result in the best available consistent performance acrosss tasks such as detection and search as well as across concepts.

- Color Correlogram (CC) : global color and structure represented as a 166-dimensional single-banded auto-correlogram in HSV space using 8 radii depths [8].
- Color Moments (CMG) : localized color extracted from a 5x5 grid and represented by the first 3 moments for each grid region in Lab color space as a normalized 225-dimensional vector.
- Co-occurrence Texture (CT) : global texture represented as a normalized 96-dimensional vector of entropy, energy, contrast, and homogeneity extracted from the image gray-scale co-occurrence matrix at 24 orientations.
- Wavelet Texture Grid (WTG)---localized texture extracted from a 3x3 grid and represented by the normalized 108-dimensional vector of the normalized variances in 12 Haar wavelet sub-bands for each grid region.

With the addition of motion image features, average precision for concept detection increased between 7-12% on individual visual features, and 4% on a fusion of five visual features. See Table 3 and Figure 6.

Table 2. Evaluation of motion image features and other low-level motion features on detection of five selected concepts with characteristic motion. Motion images show substantial improvement in average precision over histogram-based motion features.

	Car (%)	Mountain (%)	Sports (%)	Walk/Run (%)	Waterscape/ Waterfront (%)	Average (%)	Improvement of Motion Image
Motion Image	0.15463	0.20723	0.18143	0.16781	0.18697	0.179614	---
Motion Direction Histogram	0.09922	0.07659	0.09169	0.15626	0.14956	0.114664	56.6 %
Motion Magnitude Histogram	0.04672	0.0346	0.11187	0.11608	0.06615	0.075084	139.2 %
Motion Magnitude Direction Histogram	0.08495	0.05486	0.12212	0.12379	0.14969	0.107082	67.7 %

Table 3. Evaluation of visual features and the addition of motion image features on detection of five selected concepts with characteristic motion. Motion images show improvement in average precision.

	MAP (avg. prec.)	MAP with Motion Image (avg. prec.)	Improvement
260 dimensional Motion Image	0.170672	---	---
166 dimensional HSV Color Correlogram	0.270124	0.289172	7.1 %
225 dimensional Color Moments Grid	0.262652	0.294532	12.1 %
96 dimensional Cooccurrence Texture	0.27692	0.303466	9.6 %
108 dimensional Wavelet Texture Grid	0.248372	0.27859	12.2 %
Naïve Late fusion across detectors of all five features	0.369926	0.384228	3.9 %

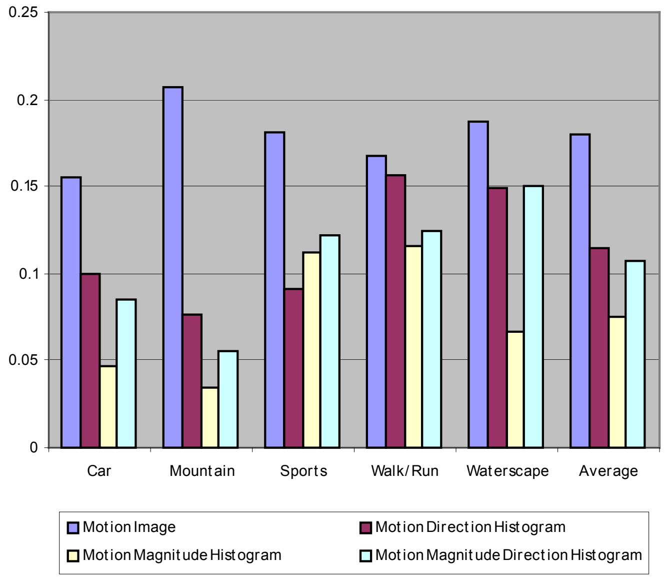

Figure 5. Comparison of average precision of low-level motion features for detection of five concepts Motion images perform more than 50% better than the next best feature.

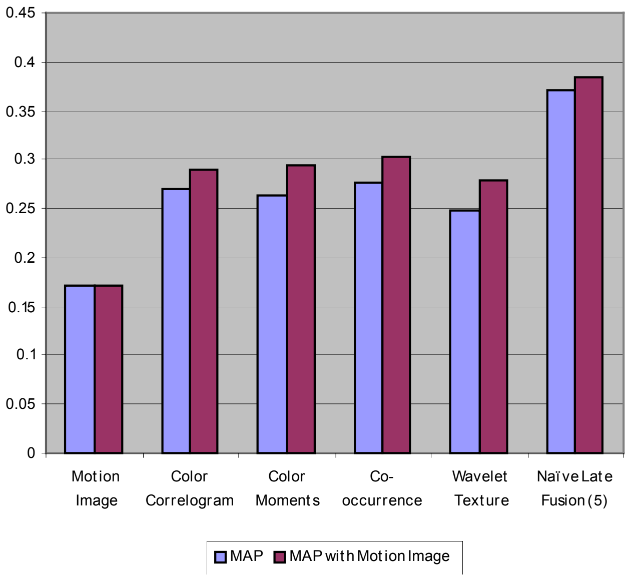

Figure 6. Comparison of average precision for concept detection of low-level visual features with and without the fusion of motion images. Average precision increases reasonably with the addition of this new feature.

4. CONCLUSIONS

We have presented a novel low-level motion feature for the classification of video events. Motion vectors from MPEG macroblocks are aggregated in a two dimensional motion image, which preserves spatial location, direction, and magnitude of the vectors. Time is compressed into intensity in the motion image.

We have successfully evaluated this feature on the TRECVID 2005 [4] dataset, and have found that our motion features are especially useful for concepts with characteristic motion patterns, including the concepts “Car”, “Sports”, “Walking/Running”, and “Waterscape/Waterfront”. Additionally, the motion feature improved detection results for the concept “Mountain”.

In future work we intend on evaluating the effect of the prediction factor of motion vectors beyond the empirically determined value used at present. We also intend on testing new models for motion images, including directional approaches in lieu of the current uni-directional method.

5. ACKNOWLEDGMENTS

This material is based upon work funded in part by the U.S. Government. Opinions, findings and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the U.S. Government.

6. REFERENCES

[1] Adams, W.H., Amir, A., Dorai, C., Ghoshal, S., Iyengar, G., Jaimes, A., Lang, C. Lin, C.Y., Naphade, M.R., Natsev, A., Neti, C., Nock, H.J., Permutter, H., Singh, R., Srinivasan, S., Smith, J.R., Tseng, B.L., Varadaraju, A.T., and Zhang, D. IBM Research TREC-2002 Video Retrieval System. In *Proceedings of the Text Retrieval Conference (TREC)* (Gaithersburg, MD, November 2002), NIST Special Publications, SP 500-251, 2002, 289-298.

[2] Amir, A., Berg, M., Chang, S.F., Iyengar, G., Lin, C., Naphade, M.R., Natsev, A., Neti, C., Nock, H., Hsu, W., Sachdev, I., Smith, J.R., Tseng, B., Wu, Y., and Zhang, D. IBM Research TRECVID-2003 Video Retrieval System. In *Proceedings of the TRECVID 2003 Workshop* (Gaithersburg, MD, November 2003), NIST Special Publications, 2003.

[3] Amir, A., Argillander J., Berg, M., Chang, S.F., Iyengar, G., Lin, C., Naphade, M.R., Natsev, A., Hsu, W., Smith, J.R., Tešić, J., Yan, R., Zhang, D. IBM Research TRECVID-2004 Video Retrieval System. In *Proceedings of the TRECVID 2004 Workshop* (Gaithersburg, MD, November 2004), NIST Special Publications, 2004.

[4] Amir A., Argillander J., Campbell M., Haubold A., Iyengar G., Ebadollahi S., Kang F., Naphade M.R., Natsev A., Smith J.R., Tešić J., and Volkmer T. IBM Research TRECVID-2005 Video Retrieval System. In *Proceedings of the TRECVID 2005 Workshop* (Gaithersburg, MD, November 2005), NIST Special Publications, 2005.

[5] Bresenham, J.E. Algorithm for computer control of a digital plotter. In *IBM Systems Journal, Vol. 4 (1)*, 1965, 25-30.

[6] Campbell M., Haubold A., Ebadollahi S., Naphade M.R., Natsev P., Smith J.R., Tešić J., and Xie L. IBM Research TRECVID-2006 Video Retrieval System. In *Proceedings of the TRECVID 2006 Workshop* (Gaithersburg, MD, November 2006), NIST Special Publications, 2006.

[7] Ewerth, R., Beringer, C., Kopp, T., Nievergall, M., Stadelmann, T., and Freisleben, B. University of Marburg at TRECVID 2005: Shot Boundary Detection and Camera Motion Estimation Results. In *Proceedings of the TRECVID 2005 Workshop*, NIST Special Publications, Gaithersburg, MD, Nov. 2005.

[8] Huang, J., Kumar, S., Mitra, M., Zhu, W., and Zabih, R. Spatial Color Indexing and Applications. In *International Journal of Computer Vision, Vol. 35 (3)*, Dec 1999, 245-268.

[9] Lu, C., Drew, M.S., and Au, J. Classification of Summarized Videos using Hidden Markov Models on Compressed Chromaticity Signatures. In *Proceedings of the ACM International Conference on Multimedia (MM ’01)* (Ottawa, CA, September 30 – October 4, 2001). ACM Press, New York, NY, 479-482.

[10] Ma, Y.-F., Zhang, H.-J. Motion Pattern-Based Video Classification and Retrieval. In *EURASIP Journal on Applied Signal Processing 2003:2*, 199-208.

[11] Naphade, M.R., Huang, M. Discovering Recurrent Events in Video Using Unsupervised Methods. In *Proceedings of the International Conference on Image Processing (ICIP ’02)* (Rochester, NY, September 22-25, 2002), IEEE Press, New York, NY, 2002, II-13 – II-16.

[12] Naphade, M.R., Kennedy, L., Kender, J.R., Chang, S.F., Smith, J.R., Over P., and Hauptmann, A. LSCOM-lite: A Light Scale Concept Ontology for Multimedia Understanding for TRECVID 2005. IBM Research Technical Report, RC23612 (W0505-104), May, 2005.

[13] Naphade, M.R. and Smith, J.R. On the Semantic Detection of Concepts at TRECVID. In *Proc. of the ACM International Conference on Multimedia (MM ’04)* (New York, NY, October 10-16, 2004), ACM Press, New York, NY, 660-667.

[14] Naphade, M.R., Wang, R., and Huang, T.S. Supporting audiovisual query using dynamic programming. In *Proc. of the ACM International Conference on Multimedia (MM ’01)* (Ottawa, Canada, September 30 – October 4, 2001). ACM Press, New York, NY, 2001, 411-420.

[15] Snoek, C.G.M., and Worring, M. Multimedia Event-Based Video Indexing using Time Intervals. In *IEEE Transactions on Multimedia, 7(4)* (Aug. 2005), 638-647.

[16] J.W. Davis. Hierarchical Motion History Images for Recognizing Human Motion. In *Proceedings of the IEEE Workshop on Detection and Recognition of Events in Video* (Vancouver, Canada, July 8, 2001). IEEE Press, New York, NY, 2001, 39-46.

[17] J.C. Niebles, H. Wang, L. Fei-Fei. Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words. In *Proceedings of the British Machine Vision Conference (BMVC ’06)* (Edinburgh, United Kingdom, September 4-7, 2006). British Machine Vision Association, 2001

[18] L. Zelnik-Manor, M. Irani. Statistical Analysis of Dynamic Actions. In *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, No. 9*, September 2006. IEEE Press, 1530-1535.