A Motion Activity Descriptor and Its Extraction in Compressed Domain

Xinding Sun, Ajay Divakaran[+], B. S. Manjunath**
*Department of Electrical and Computer
Engineering, University of California,
Santa Barbara, CA 93106
{xdsun, manj}@ece.ucsb.edu
[+]Mitsubishi Electric Research Laboratories,
571 Central Avenue, #115
Murray Hill, NJ 07974
ajayd@merl.com

Abstract. A novel motion activity descriptor and its extraction from a compressed MPEG (MPEG-1/2) video stream are presented. The descriptor consists of two parts, a temporal descriptor and a spatial descriptor. To get the temporal descriptor, the "motion intensity" is first computed based on P frame macroblock information. Then the motion intensity histogram is generated for a given video unit as the temporal descriptor. To get the spatial descriptor, the average magnitude of the motion vector in a P frame is used to threshold the macro-blocks into "zero" and "non-zero" types. The average magnitude of the motion vectors and three types of runs of zeros in the frame are then taken as the spatial descriptor. Experimental results show that the proposed descriptor is fast, and that the combination of the temporal and spatial attributes is effective. Key elements of the intensity parameter, spatial parameters and the temporal histogram of the descriptor have been adopted by the draft MPEG-7 standard [10].

1. Introduction

How to characterize the degree or intensity of a scene change and the corresponding temporal pattern of that intensity is of great significance in content-based system applications. Most previous work such as UCSB Netra V [2], IBM CueVideo [9], Columbia Video Search Engine[1], etc. has applied motion feature for content analysis, but none of the systems have addressed the above issue. It is the first question addressed in this paper. The second question addressed in this paper is how to describe the spatial distribution of the motion in a scene, i.e. how the motion varies within a given frame. To solve the above problems, a motion activity descriptor is proposed. The descriptor is a combination of two descriptors: a temporal descriptor and a spatial descriptor, which address the temporal and spatial distribution of motion respectively.

In obtaining the temporal descriptor for motion activity, we first characterize a scene change into different intensity levels, called *motion intensity*. This is based on the following observation. When describing video scene motion intensities, a person usually uses several levels of description, for example, high, low, medium, etc. In sports videos, a play can be characterized in terms of these intensity levels very clearly. At the beginning of the play, the intensity of motion is small, and it goes up and down with the progression of the play. This pattern is very similar to that of audio, which can be characterized by rhythm. After we get the intensity level of a scene, we can further characterize the temporal change of the scene. Suppose that the video sequence has been segmented into small units based on different measures [7], we can then use the histogram of the intensity, here we call it *motion intensity histogram* (MIH), to characterize the change of it. The MIH is taken as our temporal descriptor.

In obtaining the spatial descriptor for motion activity, we use motion vectors to characterize the video frames into spatial regions. Motion segmentation of image into regions tends not to be robust in practical applications. Therefore, instead of working on the segmentation, we pursue a statistical approach. The general idea is to threshold the macroblocks into zero and non-zero types based on whether it is above or below the average motion magnitude. The average magnitude and the runs of zero types can then be used to describe the spatial distribution of the motion and taken as our spatial descriptor.

The primary reason for compressed domain video processing is that it can achieve high speeds (see for example [8]). Therefore, in this paper we extract features from compressed domain information directly. Since the P frames of a video are good sub-samples of the original video, we confine our extraction to them.

2. Extraction of Temporal Descriptor for Motion Activity

2.1 Motion Intensity

To exploit temporal redundancy, MPEG adopts macroblock level motion estimation. In order to reduce the bit rate, some macroblocks in the P or B frames are coded using their differences with corresponding reference macroblocks. Since only P frames are used for later discussion of camera control, the following discussion applies to P frames only. During motion estimation, the encoder first searches for the best match of a macroblock in its neighborhood in the reference frame. If the prediction macroblock and the reference macroblock are not in the same positions of the frames, motion compensation is applied before coding. No_MC means no motion compensation. When a macroblock has no motion compensation, it is referred to as a No_MC macroblock. Generally, there are two kinds of No_MCs: one is the No_MC intra-coded and the other is the No_MC inter-coded. In typical MPEG encoder architecture, there exists an inter/intra classifier. The inter/intra classifier compares the prediction error with the input picture elements (pels). If the mean squared error of the prediction exceeds the mean squared pel value then the macroblock is intra-coded, otherwise it is inter-coded. The No_MC intra-coded and inter-coded scheme can be obtained correspondingly.

Only P frames of MPEG macroblocks have No_MC inter-coded macroblocks. In fact, in a special case, when the macroblock perfectly matches its reference, it is skipped

and not coded at all. To simplify the illustration, the skipped frames are categorized the same as No_MC inter-coded frames as shown in figure 1.

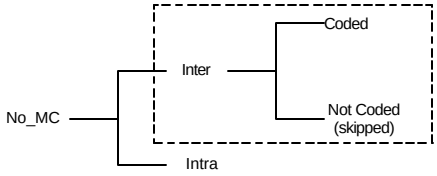


Figure 1. No_MC in MPEG video macroblocks.

According to the definition of inter No_MC, we can see that when the content of video changes are not too significant, and thus many macroblocks can be matched by their reference frames, the number of inter No_MC macroblock in a P frame would be high. For example, pauses in sports games often coincide with small object motion and fixed cameras in videos, so the corresponding number of inter No_MC macroblocks would be very high. On the other hand, when the content of the video changes rapidly, and thus many macroblocks cannot be matched by their reference frames, the number of inter No_MC macroblock in a P frame would be small. Here, we define the α-ratio of a P frame as:

$$a = \frac{Numberof\ \text{inter}\ No\_MC\ Macroblocks}{TotalNumberof\ FrameMacroblocks} \quad . \tag{1}$$

From our experiments, we found that this ratio is a good measure of scene motion intensity change and it conforms with human perception very well. The higher the ratio is, the lower the scene motion intensity change is. Figure 2 shows two frames from a football video, the first one extracted from the start of a play, which has a high α = 86%, and the second one corresponding to the play in progress, which has a low α = 5%.

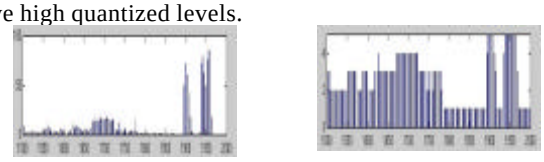
Frame 1. α=0.86 Frame 2. α=0.05

Figure 2. Two video frames with different inter No_MC ratios

As our objective is to find motion intensity levels, it is not necessary to use α-ratios directly for video motion description. So, we further quantize the ratio into several levels. Here we use the logarithmic compandor [4] that has been widely applied to speech telephony for quantization. First we compress the ratio into $G_u(a)$ using the μ_law characteristic. By using this method, we can keep quantization steps higher for high ratio values. Next, we use vector quantization methods to transform $G_u(a)$ into N_l quantized change levels. A codebook of N_l entries is extracted from the $G_u(a)$ data set first, then $G_u(a)$ values are indexed using this code book. In our experiments,

we set N_l =5. We use the index of $G_u(a)$ as the quantized level, therefore the motion intensity of a scene can be characterized by a level L =i, where i=1,2,3,4, 5.

Figure 3 shows such quantization results on P frames 1500 to 2000 of a soccer video (from MPEG 7 test data V18). The I and B frames in this interval are not used. The original α-ratios are shown in figure 3.a. The quantized change levels are shown in figure 5.b. Within the time range betweenframes 1880 to 1970, there is a pause of the play. During the pause, the scene change has a very low value and the P frames within the pause have high quantized levels.



a. Original α-Ratio b. Quantized α Level

Figure 3. The α-ratios and their quantized levels from part of the MPEG-7 data set.

2.2 Motion Intensity Histogram

Assume a video has been segmented into temporal segments (called *video units*), where these video units can be a video sequence, a shot, or small temporal segments. Then the histogram of the above levels can be used to characterize the segments' temporal intensity distributions. Note that the histogram is not dependent on the video segment size, therefore it can be easily scaled to multiple video levels. Therefore, it supports hierarchical video content description. While much research effort has been expended on frame level motion feature description, we believe that we discuss the temporal intensity distribution for the first time. The temporal intensity distribution is similar to the histogram analysis that has been used for region based image processing [6].

Given a video unit, we define our temporal descriptor as the corresponding motion intensity histogram of the unit: MIH=$[p_0, p_1, p_2, p_3, \ldots p_{N_l}]$. Where p_i is the percentage of the quantized motion corresponding to the *i*-th quantization level, and $\sum_{i=1}^{N_l} p_i = 1$. Here we set N_l =5. The intensity level within a small video temporal region usually keeps stable. Therefore this vector also conforms to human perception very well.

3. Extraction of Spatial Descriptor for Motion Activity

We use the magnitude of motion vectors with a run-length framework to form a descriptor [2]. The extraction is as follows:

For a given P frame, the "spatial activity matrix" C_{mv} is defined as: $C_{mv} = \{R(i,j)\}$, where $R(i,j) = \sqrt{x_{i,j}^2 + y_{i,j}^2}$ and $(x_{i,j}, y_{i,j})$ is the motion vector associated with the $(i,j)th$ block. For Intra-coded blocks, $R(i,j) = 0$.

The average motion vector magnitude per macro-block of the frame/object C_{mv}^{avg} is given by: $C_{mv}^{avg} = \frac{1}{MN}\sum_{i=0}^{M}\sum_{j=0}^{N} C_{mv}(i,j)$ where M and N are the width and height of the macroblocks in the frame.

We use C_{mv}^{avg} as a threshold on C_{mv} to get a new matrix as:

$$C_{mv}^{thresh}(i,j) = \begin{cases} C_{mv}(i,j), \text{ if } C_{mv}(i,j) \geq C_{mv}^{avg} \\ 0, \text{ otherwise} \end{cases}$$

Then we compute lengths of runs of zeroes in the above matrix, using a raster-scan order. Next we classify the run-lengths into three categories, short, medium and long, which are normalized with respect to the object/frame width. In this case we have defined the short runs to be 1/3 of the frame width or lower, the medium runs to be greater than 1/3 but less than 2/3 of the frame width, and the long runs to be all runs that are greater than or equal to the width. N_{sr} is the number of short runs, with N_{mr}, N_{lr} similarly defined. We use such "quantization" of runs to get some invariance with respect to rotation, translation, reflection etc.

The spatial descriptor can then be constructed as $SD = (C_{mv}^{avg}, N_{sr}, N_{mr}, N_{lr})$. Note that the descriptor indirectly expresses the number, size, and shape of distinct moving objects in the frame, and their distribution across the frame. For a frame with a single large object such as a talking head, the number of short run-lengths is high, whereas for a frame with several small objects, such as an aerial shot of a soccer game, the number of short run-lengths is lower.

4. Similarity Measure

After we obtain the motion activity descriptors for different video data sets, the MPEG-7 enabled applications can be performed based on them. However, in order to compare feature vectors, we need to provide a similarity measure. Generally, the feature vector components propose above are correlated. Therefore, when computing the similarity between two feature vectors, we use the Mahalanobis distance. The Mahalanobis distance between two feature vectors: Q_1 and Q_2 is given by:

$$D_M(Q_1,Q_2) = [Q_1\text{-}Q_2]M^{-1}[Q_1\text{-}Q_2] \ . \qquad (2)$$

Where M is the covariance matrix of the feature vector. Since M^{-1} is symmetric, it is a semi or positive matrix. So we can diagonalize it as $M^{-1} = P^T \Lambda P$, where Λ is a diagonal matrix, and P is an orthogonal matrix. Then computation of (2) can be simplified in terms of Euclidean distance as follows:

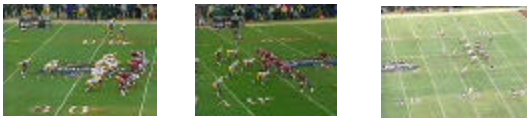
$$D_M(Q_1,Q_2) = D_E\left(\sqrt{\Lambda}PQ_1, \sqrt{\Lambda}PQ_2\right) \ . \qquad (3)$$

Since Λ and P can be computed directly from M^{-1}, the complexity of the computation of the vector distance can be reduced from $O(n^2)$ to O(n).

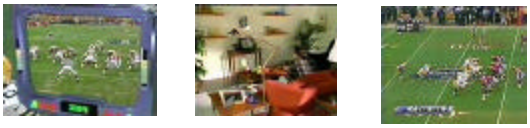
5. Experimental Results

We have implemented the descriptor for MPEG-7 data sets for the applications of video classification, retrieval, browsing, etc. Since the temporal and spatial descriptors cover two different aspects of motion activity, we first show experimental results of the two separately, then we show how to combine the two together for more powerful applications. To test the descriptors, a video is first segmented into small video units (clips) using the method proposed in [7]. The number of clips is 5% of the total video length. Therefore, a video with 100,000 frames will be segmented into 5000 small clips.

To get the temporal descriptor, the MIH is computed for each video clip. An example



(a)query clip 0427 (b)rank=1 clip 0006 (c)rank=2 clip 1284



(d)rank=3 clip 0718 (e)rank=4 clip 2043 (f)rank=5 clip 0423

Figure 4. Video query using temporal descriptor

of searching for similar video clips using the MIH descriptor and the similarity measure is shown in figure 4. The first P frame from each clip is used to represent the whole clip. The query frame shows a scene where a football game starts and there is very little motion in the scene. The 5 retrieved clips, ranked in order from 1 to 5, are displayed in the figure. As we expect, similar scenes are retrieved from the video.

Figure 5 shows the experimental results on a retrieval of a similar scene, based on the spatial descriptor SD. The first P frame from each video clip is shown in the figure. The query is an anchorperson in the news. We observe that the scenes with the anchorperson showing similar gestures are retrieved from the video stream.

A general motion activity descriptor can be constructed from its temporal and spatial parts as $Q = (\text{MIH}, SD)$. Now we explain how to use the general descriptor. First, note that the temporal attributes can apply to an entire video sequence, not necessarily to just a shot, and still be meaningful. For example, a high action movie like "The Terminator" could get a quantized intensity value that indicates high action. Moreover, the proposed temporal histogram would be even more meaningful in describing an entire movie or any other long video sequence. It immediately follows that the intensity histogram can serve to filter at the video program or sequence level. However, once we have located the program of interest, the spatial attribute becomes meaningful, since it effectively locates similar activities within a program, and thus

facilitates intra-program browsing. This is a capability unique to the motion activity descriptor among all other MPEG-7 visual descriptors.

(a) query (b) rank=1 frame 56127 (b) rank=2 frame 56952

(b) rank=3 frame 6816 (b) rank=4 frame 6891 (b) rank=5 frame 61302

Figure 5. Video query using spatial descriptor..

Based on the query results shown in figure 4, we further process the query video using the general descriptor. All the query results are used as candidates for further spatial processing. The spatial descriptor for all P frames in all candidate clips and the query clip are computed. The distance between a candidate and the query is computed as the smallest distance between two spatial descriptors, one is from the query P frames and the other is from the candidate P frames. Then we can reorder the candidates based on their distances to the query. The new order is shown in Figure 6. Clip0423 is our target, but it is the last one among the temporal query result candidates. After further spatial processing, it moves to rank 1 as expected.

(a)query clip 0427 (b)rank=1 clip 0423 (c)rank=2 clip 1284

(d)rank=3 clip 0718 (e)rank=4 clip 0006 (f)rank=5 clip 2043

Figure 6. Results from figure 4 re-sorted with the spatial

6. Conclusions

In this paper we present a novel motion activity descriptor, which includes both temporal and spatial features of motion. The proposed descriptor can be extracted using compressed domain information. It supports both similarity-based classification and retrieval, as well as other applications such as surveillance, video abstraction,

content re-purposing etc. The core of our descriptor, viz. the intensity, spatial and temporal attributes, has been accepted into the draft MPEG-7 standard motion activity descriptor.

The descriptor can be efficiently computed since only P frames of a video are processed and the descriptor can be processed directly in the compressed domain. While the temporal feature of the descriptor can be scaled to support multi-level representation and provides a basis for hierarchical video content analysis, the spatial feature can be easily scaled for different frame sizes as well.

Since the descriptor is a low level and simple descriptor, it does not carry out semantic matches. Therefore, as a standalone feature descriptor it only targets MPEG–7 enabled applications like media filtering, multimedia presentation etc. However, we can combine it with other visual features to make it more powerful in content-based multimedia applications. Future research also includes how to process the descriptor in both compressed domain and spatial domain more effectively.

Acknowledgement. This research is in part supported by the following grants/awards: NSF #EIA-9986057, NSF#EIA-0080134, Samsung Electronics, and ONR#N00014-01-1-0391. The first author would like to thank Dr. Yanglim Choi, Samsung electronics, for many fruitful discussions.

7. Reference

[1] Shih-Fu Chang, William Chen, Horace J.Meng, Hari Sundaram and Di Zhong, "A Fully Automated Content-Based Video Search Engine Supporting Spatiotemoral Queries", *IEEE Trans. on Circuits and Systems for Video Technology*, 8(5), pp.602-615, 1998.

[2] A. Divakaran and H. Sun, "A Descriptor for spatial distribution of motion activity", *Proc. SPIE Conf. on Storage and Retrieval from Image and Video Databases*, San Jose, CA 24-28 Jan. 2000.

[3] Y.Deng and B.S.Manjunath, "NeTra-V: toward an object-based video representation", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.8, (no.5), p.616-27, Sep 1998.

[4] A. Gersho and R.M. Gray , "Vector Quantization and Signal Compression, " *Kluwer Academis,* 1991

[5] B. G. Haskell, A. Puri and A. N. Netravali, "Digital Video: An Introduction to MPEG 2," *Chapman and Hall*, 1997.

[6] W. Y. Ma and B. S. Manjunath, "NETRA: A toolbox for navigating large image databases," *IEEE International Conference on Image Processing*, pp. 568-571,1997

[7] X. Sun, M. Kankanhalli, Y. Zhu and J. Wu, "Content-Based Representative Frame Extraction for Digital Video," *International Conference on Multimedia Computing and Systems*, pp. 190-194, 1998

[8] Hongjiang Zhang, Chien~Yong Low, and Stephen W. Smoliar, "Video parsing and browsing using compressed data," *Multimedia Tools and Applications*, 1(1): pp.89-111, 1995.

[9] URL: http://www.almaden.ibm.com/cs/cuevideo

[10] URL: http://www.cselt.it/mpeg/, official MPEG site.