Data Indexing for Video Shot Mining Based on Optical Flow

Ying Chen¹, WeiMing Hu, Ou Wu, XiangLin Zeng²

¹ Department of Basic Sciences
Beijing Electronic Science and Technology Institute
Beijing, P.R. China
ychen@besti.edu.cn

² National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
Beijing, P.R. China
{wmhu,wuou,xlzeng}@nlpr.ia.ac.cn

Keywords: Optical flow, Data mining, Motion-based queries, Video shot retrieval.

Abstract

Summarizing and understanding video shot based on their contents is an important research topic in multimedia data mining. This paper presents an efficient algorithm based on optical flow field to mine the motion data contained in a given video shot. Two features called *magnitude of motion pixel* and *direction of motion pixel* are constructed respectively and adopted to split the video shot into some categories automatically. The corresponding indexical structure is extracted from each category and directly applied to motion-based queries. A test system has been developed to prove the validity of our algorithm. The experimental results show that the algorithm performs well and will play an important role in content-based video shot retrieval.

1 Introduction

As the era of multimedia is coming to us, more multimedia contents are produced and distributed widely. With the rapid development of computer technologies, people benefit much from the web pages than before since the digital information gotten via internet become more popular and practicable. A typical application with content-based retrieval is useful for person or company to search their interested materials. Among these materials, since the video possesses rich visual information that is more recipient with human perceptual mechanism, it obviously outperforms the text, the voice and the image. While conventional technologies for stationary-based character retrieval is progressing at a rapid pace and becoming mature, the video still seems to be very difficult to deal with robustly. Therefore, an interesting challenge of retrieving meaningful contents in large scale video data is desired.

In order to represent the video's structure in a simpler style, we generally need to segment it into some logical related shots and assume that the segmentation is predefined by means of boundary identification [6, 10, 11]. Each segmented shot is

called group of frames (GOFs). Without loss of generality, we always assumed that all concerned operations are performed within a given GOFs. After segmentation, one or more key frames can be extracted from the shot depending on the complexity of it. For key frames extraction, a direct idea is to extract the first, middle, and/or last frame as the shot's key frame(s). Furthermore, the average alpha-trimmed approach [3, 4] presented a more robust color-histogram representation of a GOFs. The spatial and/or temporal criteria are also taken into consideration for key frame selection [7, 9]. Owing to the gap between human's apperceive, almost all these methods are based on low-level color features which sometimes do not make the retrieval outcomes satisfying. So researchers tried to find other high-level features to remedy this gap. Let's focus on two sketch shots in Fig.1 where a person is walking from left to right. Obviously, once the color distribution of the person and the background are changed, the two shots are likely to be not classified together based on color features but they have the same action. As a high-level feature, for video mining and retrieval, optical flow can easily overcome these shortages and offers a complementarity to low-level color features. Some researchers woke up to this point and applied optical flow to motion based retrieval [2, 8]. However, a drawback is that they do not availably mine the motion-based indexing to provide the most powerful video shot representation.

Figure 1: Two video shots with the same action and the different color distributions.

In this paper, we propose a new algorithm for video data mining. Our algorithm splits a given GOFs into some categories by computing the optical flow and then constructs a complete retrieval histogram for each partitioned category. Experimental results show that the proposed scheme is effective in terms of accuracy and efficiency.

The rest of the paper is arranged as follows. Section 2 introduces the proposed method. Sections 3 presents our

experimental results. Section 4 concludes the paper.

2 Proposed algorithm

For a given GOFs, we assume that the size of its frame is $X \times Y$ and the number is $N + 1$. The smoothed image $g_t(x, y)$ is regarded as the convolution of the initial image $f_t(x, y)$ and the filter $h(x, y)$:

$$g_t(x, y) = h(x, y) * f_t(x, y), \text{ for } 0 \leq t \leq N. \quad (1)$$

Then two features called magnitude of motion pixel (MOMP) and direction of motion pixel (DOMP) at (x, y) are defined by:

$$\begin{aligned} & \text{MOMP}_t(x, y) \\ &= \frac{1}{(2l+1)^2} \sum_{x'=x-l}^{x+l} \sum_{y'=y-l}^{y+l} \sqrt{\mu_t(x', y')^2 + \nu_t(x', y')^2}, \end{aligned} \quad (2)$$

$$\begin{aligned} & \text{DOMP}_t(x, y) \\ &= \frac{1}{(2l+1)^2} \sum_{x'=x-l}^{x+l} \sum_{y'=y-l}^{y+l} \arg(\mu_t(x', y'), \nu_t(x', y')), \end{aligned} \quad (3)$$

where the optical flow vectors $(\mu_t(x', y'), \nu_t(x', y'))$ are obtained by Horn-Schunck's estimation [5], $\sqrt{\cdot} \geq 0$ is the modulus indicating the motion magnitude, $\arg(\cdot, \cdot) \in [0, 2\pi)$ is the argument principal value indicating the motion direction, and l controls the size of window templet.

2.1 Classifying the GOFs

Since a single GOFs probably contain several distinct events, we believe that no algorithm can do a good job of extracting key frames without considering the difference between the different situations. Consequently, a hierarchical method is developed based on motion analysis in which we first classify GOFs into some categories and then apply the appropriate key frame extraction to each category. Our idea is illustrated in Fig.2 and Fig.3. In Fig.2, the camera is zoomed in the way “far–near–far–near”. When the camera is located in different position, the content of its frame also takes a transition. Obviously, 2.(a), 2.(b), 2.(e) and 2.(f) have the similar magnitude of scaling, while 2.(c), 2.(d), 2.(g) and 2.(h) have another magnitude. Similarly in Fig.3, the woman runs in the way “left–right–left–right”. It is natural to think that 3.(a), 3.(b), 3.(e), and 3.(f) (or 3.(c), 3.(d), 3.(g), and 3.(h)) belong to the same category in which they have the similar direction of motion. Therefore, it is desirable to classify them into two categories by the magnitude of scaling or the direction of motion to improve the retrieval performance. However, for these cases mentioned above,

Figure 2: A video shot sequences with camera zoom.

Figure 3: A video shot sequences with a running woman.

existing methods cannot capture all this important information of the frames.

In view of this, we compute an intuitionistic motion metric based on MOMP and DOMP and analyze the metric as a function of time to classify the frames into some categories. In the first step, the sum of MOMP and DOMP at each pixel are calculated as the metric $M(t)$ and $D(t)$ for $g_t(x, y)$:

$$M(t) = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \text{MOMP}_t(x, y), \quad (4)$$

$$D(t) = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \text{DOMP}_t(x, y). \quad (5)$$

The second step scans the $M(t)$ and $D(t)$ vs. t curve starting at $t = 1$. To facilitate the motion statistics calculation process, we quantize $M(t)$ and $D(t)$ into a smaller domain by the following equations:

$$M'(t) = \lfloor \frac{M(t)}{I_1} + 0.5 \rfloor, \quad (6)$$

$$D'(t) = \lfloor \frac{D(t)}{I_2} + 0.5 \rfloor, \quad (7)$$

where I_1 and I_2 control the degree of quantization and $\lfloor \cdot \rfloor$ called floor function represents the maximal integer no more than $\cdot$. Then based on MOMP or DOMP respectively, if two frames t_1, t_2 are in the same group, they satisfy equation $M'(t_1) = M'(t_2)$ or $D'(t_1) = D'(t_2)$. For example, Fig.4 and Fig.5 show the frames classified by the MOMP for a volleyball shot. It can be seen that the classification captures salient action events in this shot.

Figure 4: The initial volleyball video shot sequences.

Figure 5: Classification for the volleyball video shot sequences based on MOMP.

2.2 Motion histogram

Based on MOMP, the shot is divided into some categories and we assume the number is m . Let M_0 is the maximum of $MOMP_t(x, y)$ in the entire training set, then the modulus of optical flow at each pixel point can be quantized into B_M bins. For each category $C_i^M = \{g_{i1}(x, y), \dots, g_{if_i}(x, y)\}$ ($1 \leq i \leq m$), the magnitude frequency of occurrence in the k -th bin in $g_{ip}(x, y)$, labeled as h_{ipk}^M , is defined as:

$$h_{ipk}^M = \frac{\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \delta(\lfloor \frac{B_M MOMP_{ip}(x, y)}{M_0} \rfloor + 1 - k)}{XY}, \quad (8)$$

where $k \in \{1, \dots, B_M\}$, $p \in \{1, \dots, f_i\}$, and

$$\delta(\#) = \begin{cases} 1, & \text{if } \# = 0, \\ 0, & \text{if } \# \neq 0. \end{cases} \quad (9)$$

For a given k , sort these h_{ipk}^M 's values in an ascending order as follows:

$$h_{ip_1k}^M \leq h_{ip_2k}^M \leq \dots \leq h_{ip_{f_i}k}^M, \quad (10)$$

where $(p_1, \dots, p_{f_i})$ is a permutation of $(1, \dots, f_i)$. Taking the transform

$$\bar{h}_{ilk}^M = h_{ip_1k}^M, \quad (11)$$

A robust statistical histogram, which is constituted by averaging some $\bar{h}_{ilk}^M$'s values, can be treated as an motion representation based on the magnitude for the k -th bin:

$$H_i^M(k, \alpha) = \frac{1}{f_i - 2\lfloor \alpha f_i \rfloor} \sum_{l=\lfloor \alpha f_i \rfloor + 1}^{f_i - \lfloor \alpha f_i \rfloor} \bar{h}_{ilk}^M, \quad (12)$$

where the parameter $\alpha \in [0, 0.5]$ controls the selection of bin values. Incidentally, the computational process is equivalent to extracting the mean of all $\bar{h}_{ilk}^M$'s values when $\alpha = 0$ and the median when $\alpha = 0.5$. Based on DOMP, let the number of categories is d . For each category $C_j^D = \{g_{j1}(x, y), \dots, g_{jf_j}(x, y)\}$ ($1 \leq j \leq d$), a quantization for segmenting the DOMP into B_D bins from 0 to 2π is used. Obviously, the larger the value of MOMP, the more visible the direction. Therefore, the direction frequency of occurrence in the k -th bin in $g_{jp}(x, y)$, labeled as h_{jpk}^D , is defined as:

$$h_{jpk}^D = \frac{\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \lambda(x, y) \delta(\lfloor \frac{B_D DOMP_{jp}(x, y)}{2\pi} \rfloor + 1 - k)}{XY}, \quad (13)$$

where $k \in \{1, \dots, B_D\}$, $p \in \{1, \dots, f_j\}$, and

$$\lambda(x, y) = \frac{MOMP_{jp}(x, y)}{M_0} \quad (14)$$

is the weighted factor. Similarly, the histogram of the k -th bin based on DOMP, $H_j^D(k, \alpha)$, also can be computed. Fig.6 illustrates the corresponding values of $H_j^D(k, 0.2)$ for the different motion directions in Fig.3. Easy to see that our scheme offers an adaptive representation which can give prominence to the dominant motion directions of the shot.

Figure 6: The values of $H_j^D(k, 0.2)$ for the two different categories after classifying the shot in Fig.3.

2.3 Matching and query

In our algorithm, a given GOFs is represented as

$$\begin{pmatrix} H_1^M(1, \alpha) & H_1^M(2, \alpha) & \cdots & H_1^M(B_M, \alpha) \\ H_2^M(1, \alpha) & H_2^M(2, \alpha) & \cdots & H_2^M(B_M, \alpha) \\ \vdots & \vdots & \ddots & \vdots \\ H_m^M(1, \alpha) & H_m^M(2, \alpha) & \cdots & H_m^M(B_M, \alpha) \end{pmatrix} \quad (15)$$

and

$$\begin{pmatrix} H_1^D(1, \alpha) & H_1^D(2, \alpha) & \cdots & H_1^D(B_D, \alpha) \\ H_2^D(1, \alpha) & H_2^D(2, \alpha) & \cdots & H_2^D(B_D, \alpha) \\ \vdots & \vdots & \ddots & \vdots \\ H_d^D(1, \alpha) & H_d^D(2, \alpha) & \cdots & H_d^D(B_D, \alpha) \end{pmatrix}. \quad (16)$$

Let F and F' denote two shots, based on MOMP, their feature distance is measured by:

$$\begin{aligned} & Dist_{(F, F')}(H^M) \\ &= \sum_{i=1}^m \omega_i^M \frac{\sum_{k=1}^{B_M} |H_i^M(k, \alpha)(F) - H_i^M(k, \alpha)(F')|}{\sum_{k=1}^{B_M} (H_i^M(k, \alpha)(F) + H_i^M(k, \alpha)(F'))} \end{aligned} \quad (17)$$

where ω_i^M is user specified weight. Based on DOMP, their feature distance $Dist_{(F, F')}(H^D)$ is defined similarly. Then the overall feature distance can be measured by

$$\begin{aligned} & Dist(F, F') \\ &= \omega Dist_{(F, F')}(H^M) + (1 - \omega) Dist_{(F, F')}(H^D), \end{aligned} \quad (18)$$

where ω is user specified weight. The best match for the query shot is the one with the smallest overall feature distance.

By this time, the referred retrieval is based on an example query, that is to say, the computer matches a given query shot from the database without any user interaction. However, the user usually holds the balance in the retrieval performance. Therefore, if a prior knowledge about the query shot is

introduced with human supervision, the efficiency of retrieval may be improved. On the other hand, we notice that the direction is one of the characters to describe many actions and easy to be summarized quantificationally. For example, if the query shot is expressed in Fig.3, the user easily knows that there are two domain motion directions in this shot by quick browsing. To retrieve the similar action quickly, the query can be firstly represented as the union of some disjunct directional intervals. For the example in Fig.3, the directional intervals are $[0^\circ, 5^\circ] \cup [355^\circ, 360^\circ]$ and $[175^\circ, 185^\circ]$, where $[0^\circ, 5^\circ] \cup [355^\circ, 360^\circ]$ corresponds to the domain motion direction “Right” and $[175^\circ, 185^\circ]$ corresponds to “Left” respectively. Generally, we assume that the directional intervals of query are

$$Q = [\theta_{i1}, \theta_{i2}] \cup [\theta_{i3}, \theta_{i4}], \cup \cdots \cup [\theta_{in-1}, \theta_{in}]. \quad (19)$$

For all S_i in the database, we compute its $H_j^D(k, \alpha)$. Let

$$H^D(k, \alpha) = \frac{1}{d} \sum_{j=1}^d H_j^D(k, \alpha), \quad (20)$$

and

$$H = \sum_{k=1}^{B_D} H^D(k, \alpha). \quad (21)$$

The direction intervals similarity between S_i and Q is defined as:

$$\begin{aligned} & Dist(S_i, Q) \\ &= \sum_{[\theta_{s-1}, \theta_s] \subseteq Q} \sum_{k=1}^{B_D} \frac{B_D \cdot I \cdot H^D(k, \alpha)}{2\pi H} f(\theta_{s-1}, \theta_s, k), \end{aligned} \quad (22)$$

where

$$I = \min\{\theta_s, k \frac{2\pi}{B_D}\} - \max\{\theta_{s-1}, (k-1) \frac{2\pi}{B_D}\} \quad (23)$$

denotes the intersection of $[\theta_{s-1}, \theta_s]$ and $[(k-1) \frac{2\pi}{B_D}, k \frac{2\pi}{B_D}]$, while the function

$$\begin{aligned} & f(\theta_{s-1}, \theta_s, k) \\ &= \begin{cases} 1, & \text{if } [\theta_{s-1}, \theta_s] \cap [(k-1) \frac{2\pi}{B_D}, k \frac{2\pi}{B_D}] \neq \emptyset \\ 0, & \text{if } [\theta_{s-1}, \theta_s] \cap [(k-1) \frac{2\pi}{B_D}, k \frac{2\pi}{B_D}] = \emptyset. \end{cases} \end{aligned} \quad (24)$$

is used to judge whether they intersect. Go over all $Dist(S_i, Q)$, select those S_i subjected to

$$Dist(S_i, Q) \geq Dist_0 \quad (25)$$

as the elements of the possible result set, where $Dist_0$ is a threshold. Now, the needed shot can be retrieved from a slightly smaller set rather than the initial database. Such strategies discussed above are called dominant direction priority scheme (DDPS).

3 Experimental results

We will focus on the analysis of final algorithmic retrieval performance. 690 pre-cut shots, extracted from the video sequences of volleyball, basketball, football and so on, were used to test. Considering the complexity of computation, we set m to 5, d to 8 in our all experiments. To evaluate the performance of the proposed algorithm quantificationally, the *average normalized modified retrieval rank* (ANMRR) and the *average recall* (AR), which were developed in [1], are chosen as the benchmark indicators. The value of ANMRR determines the rank of the correct shots unretrieved and the value of AR determines the rate of the correct shots retrieved. The lower the value of the ANMRR, the better the performance. In contrast, the higher the value of the AR, the better the performance.

3.1 Experiment I

The first experiment is to compare the performance of our algorithm with different values of α . Fig. 7 shows the ANMRR and the AR with $\alpha = \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$. It indicates that our algorithm achieves better performance for the selected test sets when $\alpha \in [0.2, 0.3]$. So in the next experiments, we let α be 0.2.

3.2 Experiment II

In the second experiment, we use Table 3.2 to list the ANMRR and the AR values with different algorithms. It can be seen that our algorithm result in a lower ANMRR and a higher AR, and that means the performance is better than the others.

	ANMRR	AR
Algorithm in [2]	0.4775	0.6407
Algorithm in [8]	0.4536	0.6616
Algorithm in [9]	0.4633	0.6341
Our algorithm	0.3983*	0.7216*

Table 1: Comparison of the ANMRR values and the AR values with different algorithms, where the number of queries is 690 and the mark * indicates the better performance.

3.3 Experiment III

The third experiment considers the performance of dominant direction priority scheme (DDPS). We selected 107 shots whose dominant direction can be predefined by user from the database containing 690 shots. Fig.8 shows the ANMRR and the AR of the operation to retrieve the selected 107

Figure 7: Performance of our algorithm with different α .

shots with DDPS and without DDPS. In point of experimental results, these two schemes have similar performance. But after filtering out the unbecoming dominant direction, the former only retrieves from a set whose average number of elements is 61 while the latter is 609 throughout. That is to say, it consumedly proves the serviceability of the DDPS.

4 Conclusion

In this paper, we proposed an algorithm based on optical flow for video shot retrieval. Our algorithm can mine the most significant motion contents within a video shot. The retrieval performance is enhanced by considering the classification of GOFs and the statistical information of each category. Experiments have demonstrated that it is indeed powerful. An example of query results for a given video shot is shown in Fig.9. Several questions remain to be addressed by future works. Firstly, during constructing the motion histogram only the statistical information of motion is adopted, the spatial information of motion is ignored. One possible way to solve this problem is to spatially split the optical flow field in four equal regions or even more, for each region we can build the motion index. And that, such operation maybe

Figure 8: Overall ANMRR and AR without DDPS and with DDPS respectively when the number of queries varied in {10, 30, 50, 70, 90, 107}.

ultimately realize the retrieval for motion trajectories of the object. Secondly, an excellent retrieval system based on a single feature is not practical. It is necessary to merge motion feature with other features, such as color, shape or audio cue, which leads to the detection of more semantic contents in the video.

Figure 9: Top 9 retrieval results are listed; only their first frames are shown.

References

- [1] Mpeg-7 visual part of experimentation model (xm) version 2.0. *MPEG-7 Output Document ISO/MPEG*, 1999.
- [2] E. Ardizzone and M. L. Cascia. Video indexing using optical flow field. *IEEE International Conference on Image Processing*, pages 831–834, 1996.
- [3] A. M. Ferman, S. Krishnamachari, A. M. Tekalp, M. A. Mottaleb, and R. Mehrotra. Group-of-frames/pictures color histogram descriptors for multimedia applications. *IEEE International Conference on Image Processing*, pages 65–68, 2000.
- [4] A. M. Ferman, A. M. Tekalp, and R. Mehrotra. Robust color histogram descriptors for video segment retrieval and identification. *IEEE Transaction on Image Processing*, 11:497–508, 2002.
- [5] B. K. P. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, pages 185–203, 1981.
- [6] R. A. Joyce and B. D. Liu. Temporal segmentation of video using frame and histogram space. *IEEE Transactions on Multimedia*, 8:130–140, 2006.
- [7] H. C. Lee and S. D. Kim. Rate-driven key frame selection using temporal variation of visual content. *Electronics Letters*, 38:217–218, 2002.
- [8] A. G. Nguyen and J. N. Hwang. Scene context dependent key frame selection in streaming. *International Conference on Distributed Computing Systems Workshops*, pages 208–213, 2002.
- [9] K. W. Sze, K. M. Lam, and G. P. Qiu. A new key frame representation for video segment retrieval. *IEEE Transaction on Circuits and Systems for Video Techonology*, 15:1148–1155, 2005.
- [10] J. Yu and M. D. Srinath. An efficient method for scene cut detection. *Pattern Recognition Lettters*, 22:1379–1391, 2001.
- [11] R. Zhao and W. I. Grosky. A novel video shot detection technique using color anglogram and latent semantic indexing. *International Conference on Distributed Computing Systems Workshops*, pages 550–555, 2003.