은닉 마르코프 모델을 이용한 MPEG 압축 비디오에서의 점진적 변환의 검출

(Detection of Gradual Transitions in MPEG Compressed Video using Hidden Markov Model)

최 성 민 † 김 대 진 †† 방 승 양 †††
(Sungmin Choi) (Daijin Kim) (Sung-Yang Bang)

요 약 비디오 요약의 첫 걸음은 샷(shot) 변환의 검출이다. 이러한 샷 변환은 점진적인 변환과 급진적인 변환이 있다. 지금까지 급진적인 샷 변환은 이미 주어진 한계치나 연속된 두 프레임의 이미지에 기반을 둔 거리를 이용하여 검출하였고 점진적 변환 또한 일반적으로 한계치를 이용하여 검출하였다. 그러나 한계치에 따라 그 결과가 확연히 달라지고 또한 그 한계치를 정하는 것도 어려운 문제이다. 이 논문에서는 이런 문제의 해결과 MPEG 압축 비디오 상에서 점진적 변화의 검출뿐만 아니라 분류를 해결하는 방법을 제시하였다. 논문에서는 한계치를 사용하지 않은 은닉 마르코프 모델과 MPEG의 근사 DC 값을 이용하여 보다 빠르고 정확한 결과를 얻도록 하였다. 그리고 히스토그램의 차이뿐만 아니라 매크로 블록(macro block)의 차이라 불리는 새로운 척도를 도입하여 보다 정확한 값을 얻도록 하였다. 은닉 마르코프 모델은 샷, 페이드(fade), 디졸브(dissolve), 컷(cut) 등의 4개의 상태를 갖게 하고 학습은 Baum-Welch 알고리즘으로 필요한 변수들을 추정하였다. 그리고 특정 벡터에 Viterbi 알고리즘을 적용하여 원하는 상태를 얻을 수 있다. 대부분의 실험 결과를 보면 새로 제안한 척도를 사용한 방법이 히스토그램의 차만을 이용한 방법보다 더 좋은 결과를 나타내었으며 이산적 마르코프 모델보다 연속적 마르코프 모델이 좋은 결과를 보여준다.

키워드 : 점진적 변환, 은닉 마르코프 모델, MPEG, 매크로 블록 차이, 히스토그램 차이

Abstract Video segmentation is a fundamental task in video indexing and it includes two kinds of shot change detections such as the abrupt transition and the gradual transition. The abrupt shot boundaries are detected by computing the image-based distance between adjacent frames and comparing this distance with a pre-determined threshold value. However, the gradual shot boundaries are difficult to detect with this approach. To overcome this difficulty, we propose the method that detects gradual transition in the MPEG compressed video using the HMM (Hidden Markov Model). We take two different HMMs such as a discrete HMM and a continuous HMM with a Gaussian mixture model. As image features for HMM's observations, we use two distinct features such as the difference of histogram of DC images between two adjacent frames and the difference of each individual macroblock's deviations at the corresponding macroblocks between two adjacent frames, where deviation means an arithmetic difference of each macroblock's DC value from the mean of DC values in the given frame. Furthermore, we obtain the DC sequences of P and B frame by the first order approximation for a fast and effective computation. Experiment results show that we obtain the best detection and classification performance of gradual transitions when a continuous HMM with one Gaussian model is taken and two image features are used together.

Key words : gradual transition, Hidden Markov Model, MPEG, macro block difference, histogram difference

† 비 회 원 : (주)ITM 정보통신연구소 연구원
hopemini@postech.ac.kr
†† 비 회 원 : 포항공과대학교 컴퓨터공학과 교수
dkim@postech.ac.kr
††† 종신회원 : 포항공과대학교 컴퓨터공학과 교수
sybang@postech.ac.kr
논문접수 : 2003년 1월 16일
심사완료 : 2003년 12월 23일

1. Introduction

Due to rapid advances in computing and communication technologies, human beings are constantly being inundated with information in form of text,

image, audio, video and spatial data. There is an overwhelming need for an integrated multimedia system to reduce the work and information overload. The technologies for handling multimedia data are most important and most challenging. Much information is contained in video data but it is difficult to handle them. To alleviate this difficulty, we often need to segment the video data effectively.

Usually, one frame is a minimum unit of video and corresponds to one film of movie. A shot is a basic unit of video segmentation and a sequence of frames that are continuously captured from the same camera. Many video applications are based on the shot of video data. Identifying shot changes and indexing the video sequence will facilitate the fast browsing and retrieval of the content of interest to the user[1].

Shot changes can be divided into two categories: the abrupt and the gradual transition. An abrupt transition is often called a cut that occurs at an abrupt shot change in a single frame. Gradual transition includes a fade-in, a fade-out, a dissolving, etc. The fade is a slow change in brightness usually resulting in or starting with a solid black frame. The dissolve occurs when the images of the first shot get dimmer and the images of the second shot get brighter, with frames within the transition showing one image superimposed on the other. Many other types of gradual transition are possible. Gradual transition are more difficult to detect, since the frame content share the same semantic properties as camera motion[2].

Gamaz et. al.[2] proposed an algorithm using the DC coefficients, the number of forward and backward, and the interpolated macro blocks in the MPEG compressed video. But they just detected a cut, not gradual transition. Fernando et. al.[3] proposed some algorithms using mathematical models. Their algorithms detected gradual transitions. However, when the approximated DC-sequences were applied, their algorithms did not show a good result. Park et. al.[4] and Boreczky et. al.[5] used the HMM for detecting the gradual transition. But they used the color information, the edge information, and the speech information in the original uncompressed video.

In this paper, we propose a fast and robust algorithm for detecting the gradual transition in MPEG compressed video using HMM. We use the approximated DC-sequences of MPEG video, not all pixels, so we can detect gradual transitions rapidly. We also use only intensity information as image feature such as the difference of histogram of DC images between two adjacent frames and/or the difference of each individual macro block's deviations at the corresponding macro blocks between two adjacent frames. These feature are statistically modeled by HMM and gradual transitions will be detected by the trained HMM.

The outline of the paper is as follows. In section 2, we introduce methods that we use in proposed paper. In section 3, a new algorithm is proposed to detect gradual transition. Some experimental results are presented in section 4. Section 5 contains a summary and conclusions.

2. Background

2.1 DC images and DC sequences

DC images are spatially reduced versions of the original images. An image is divided in blocks of $N \times N$ pixels. The (i, j) pixel of the DC-image is the average value of the (i, j) block of the original image. Sequence formed in such manner will be called DC-sequence[6].

For JPEG and the I-frame of MPEG, the original image is grouped into 8×8 blocks and the DC term $c(0,0)$ of its 2-D discrete-cosine transform (DCT) is related to the pixel values $f(i,j)$

$$c(0,0) = \frac{1}{8} \sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \quad (1)$$

that is equal to 8 times the average intensity of the block.

While the DC-image is much smaller than the original image, it still retains significant amount of information. This means that a global computation originally performed on the original image could be performed on the DC-image.

Forming the DC-images from an uncompressed original image requires $O(N^2)$ operations per block. The DC image can be easily obtained by choosing the DC term in the case of the DCT-based

compressed images. From Eq. (1), we know that the average value of each DCT block is one eighth of the DC term of the DCT block in the case of N = 8 [7].

The advantages of using the DC-images and DC-sequences for shot analysis operations are twofold. First, the operations are performed directly on compressed data, thus eliminating the need for full-frame decompression. Second, we are working with a small fraction of the original data[6].

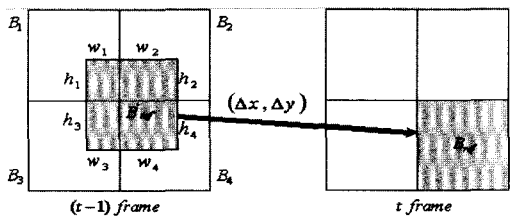


Figure 1 Reference block B_{ref}, motion vectors and original blocks

The extraction of DC image from the I-frame in the MPEG video is trivial as Eq. (1). For P-frame and B-frame, motion information must be employed to derive the DC images. Suppose in Fig.1 B_{ref} is the current block of interest in the current frame, which is corresponding to B'_{ref} in the previous frame by $B_{ref} = B'_{ref} + \Delta u$ and $\Delta u = (\Delta x, \Delta y)$. Next, we have the four original neighboring blocks $B_1, \ldots, B_4$ of the block B'_{ref} (see Fig. 1). Let h_i and w_i be the height and width of $B'_{ref} \cap B_i$. Then, the motion vector $\Delta u = (\Delta x, \Delta y)$ is equal to (w_1, h_1). Due to the linearity of DCT, the DC component of B_{ref} in the current frame can be approximated by

$$DC(B_{ref}) = \sum_{i=1}^{4} \frac{h_i w_i}{64} DC(B_i) \qquad (2)$$

In Eq. (2), the cost for computing DC coefficient in P and B frame is reduced to at most 4 multiplications. In this approximation, we need the DC coefficients of the 4 neighboring block $B_1, \ldots, B_4$ and motion vector information to obtain the approximated DC coefficient of the block B_{ref}. Such information is easily obtained from the MPEG compressed stream.

2.2 Hidden Markov Model

The HMM models can be exploited to investigate the time varying sequences of observations[8]. The elements of an HMM is specified by the following:

- N : The number of states in the model. The individual states are denoted as $S = \{S_1, S_2, \ldots, S_N\}$ and the state of the model at time t is q_t, $1 \le q_t \le N$ and $1 \le t \le T$ where T is the length of the output observable symbol sequence.
- M : The number of distinct observable symbols. The individual symbols are denoted as $V = v_1, v_2, \ldots, v_m$.
- $A_{N \times N}$: An $N \times N$ matrix specifies the state-transition probability that the state will transit from state S_i to state S_j. $A_{N \times N} = [a_{ij}]_{1 \le i,j \le N}$ where $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$.
- $B_{N \times M}$: An $N \times M$ matrix specifies that the system will generate the observable symbol v_k at state S_j and at time t. $B_{N \times M} = [b_j(k)]_{1 \le j \le N, 1 \le k \le M}$, where $b_j(k) = P(v_k \text{ at } t | q_t = S_j)$.
- π_N : An N-element vector indicates the initial state probability. $\pi_N = [\pi_i]_{1 \le i \le N}$, where $\pi_i = P(q_1 = S_i)$.

The complete parameter set λ of the discrete HMM is represented by $\lambda = \pi, A, B$. The parameter selection process is called the HMM training process. This parameter set λ can be used to evaluate $P(O|\lambda)$, that is to measure the maximum likelihood performance of an output observable symbol sequences $O = o_1, \ldots, o_T$, where T is the number of observable sequence. For evaluating each $P(O|\lambda)$, we need symbols to select the number of states N, the number of observable M and then compute the result of probability density vector π and matrices A and B by training each HMM from a set of corresponding training data. In order to use a continuous observation density, it needs re-estimation of $b_j(O)$[8].

3. Proposed gradual transition detection

The proposed gradual transition detection system consists of DC module, training module and detection module as shown in Fig. 2.

DC module is extracting DC-sequences in MPEG

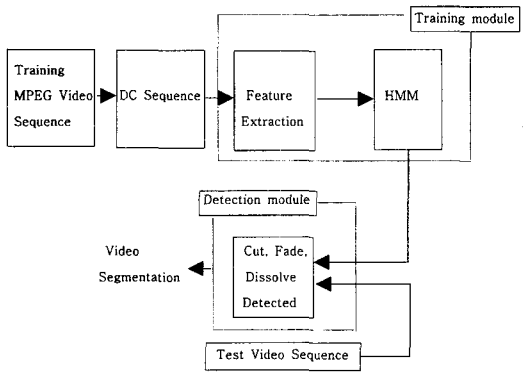


Figure 2 An architecture of the proposal gradual transition detection system

compressed video. In order to reduce time complexity, DC-sequences are extracted approximately as explained in section 2.1.

Training module consists of extracting features and training HMM parameters. Features used in this paper are shown in Eq. (3) and Eq. (4) where the index i is the frame number. In this paper, only the luminance component Y value among Y, Cb, Cr color space used to compute the image features.

$$HD_i = \sum_{k=0}^{255} |H_{i+1}[k] - H_i[k]| \quad (3)$$

where $H_i[k]$ is the kth histogram of the ith DC image.

$$MD_i = \sum_{k=1}^{N} |(DC_{i+1}[k] - \mu_{i+1}) - (DC_i[k] - \mu_i)| \quad (4)$$

where $DC_i[k]$ is the DC value of the kth macroblock in the ith frame, N is the number of macroblocks in the frame and u_i is the mean value of the ith DC image, i.e.,

$$\mu_i = \frac{1}{N} \sum_{k=1}^{N} DC_i[k] \quad (5)$$

In Eq. (3), the difference of histogram between two adjacent frames is calculated by not all pixels of the frame [4, 5], but DC values of the frame. Eq. (4) takes effect of difference of variance between two adjacent frames because of $DC_i(k) - \mu_i$ term and means the difference of each individual macro block's deviations at the corresponding macro blocks between two adjacent frames.

Fig. 3 shows the HMM which has been used for video segmentation. The states model the transition

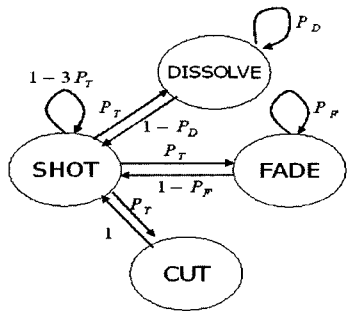


Figure 3 HMM for video segmentation

segments between shot, fade, dissolve and cut. The shot state models segments of the video within a single shot. The arcs between states model the allowable progressions of segments. Thus, from the shot state it is possible to go to any of the transition states, but from a transition state it is only possible to return to the shot state. This assures that only a single transition segment occurs between shots. The arcs from a state to itself model the length of time the video is in that particular state[4,5]. A probability is associated with each of the arcs in Fig. 3. The probability P_t is the probability that a transition occurs. We assume for simplicity that each of the transition type, cut, fade and dissolve are equally likely. The probability $1-2p_t$ is the probability of staying in a shot. The probability P_D is the probability of staying in the dissolve state. Similarly, we can define the probability P_F for the fade. The probability $1-P_D$ is the probability of returning from dissolve back to a shot. Since a transition is abrupt in the cut state, the transition occurs from shot to cut and returns from cut back to shot immediately.

The parameters of the HMM, namely the transition probabilities P_T, P_D, P_F are learned during the training phase. Data for training consists of features - Eq. (3) and/or Eq. (4) - that are computed for a collection of video, labeled according to whether there is a shot, a cut, a fade, a dissolve. Given this data, Baum-Welch re-estimation [8] is applied for training data.

Once the parameters are trained, segmenting the video into its shots and transitions is performed using the Viterbi algorithm [8] that is a standard

technique for segmentation and recognition with HMMs. Given a sequence of features, the Viterbi algorithm produces the sequence of states most likely to have generated these features. The state sequence is time-aligned with the feature sequence, so that the video is segmented into shots, cuts, fades or dissolves according to the time of feature sequence.

4. Experiment results and discussion

The proposed detection algorithm has been tested on video data including music video, news, documentary and so on [4]. Table 1 shows video data for experiment. Video data is MPEG-1 compressed video with size of 352×240.

Training data and test data consist of half part and remaining part of each video, respectively.

For detecting gradual transitions, it needs to modeling HMM and estimating parameters. An HMM is modeled as Fig. 3 of section 3 and parameters are estimated by using Baum-Welch algorithm [8]. Now, observations of HMM are features which obtain in Eq. (3) and Eq. (4). After parameters of HMM are trained, features that are obtained from the test video data are applied to the trained HMM as the observation sequence. By Viterbi algorithm [8], optimal states are estimated and gradual transitions and cuts are detected.

The experiment has been carried out by the following.

1. The two different HMM models, i.e., discrete HMM and continuous HMM, are used.
2. In continuous HMM, one Gaussian and Gaussian mixture are applied.
3. HD, MD and HD + MD are used as the observation of each HMM.

To validate performance of detecting gradual transition, we calculate recall(Eq. (6)) and precision(Eq. (7)) used generally.

$$reacall = \frac{Detects}{Detects + MD's} \tag{6}$$

$$<ision = \frac{Detects}{Detects + FA's} \tag{7}$$

where *Detects* is the number of correct detections, *MD* is the number of missing detections and *FA* is the number of false detections.

Detection results of the trained HMM over three different HMM models, such as discrete HMM, continuous HMM with one Gaussian and continuous HMM with Gaussian Mixture are shown from Fig. 4 to Fig. 6, respectively, where the numbers 1, 2, 3 and 4 of y axis represent the shot, the fade, the dissolve and the cut state, respectively. And x axis

Table 1 Video data for experiment

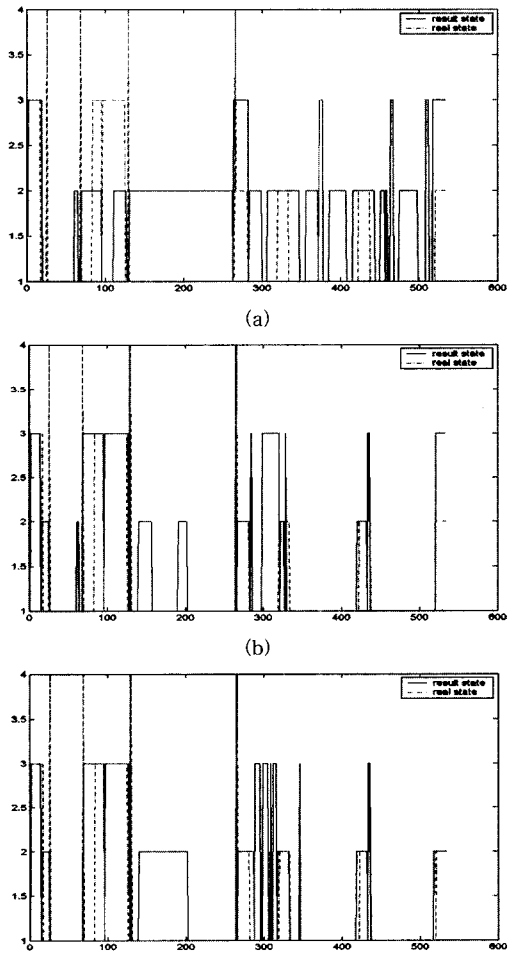
video data	# of frame	# of dissolves	# of fades	# of cuts
music video 1	875	3	4	7
music video 2	955	7	3	5
documentary 1	185	4	0	0
documentary 2	590	4	0	0
news	2661	2	2	13

Table 2 Result of detection about Fig. 4, Fig.5 and Fig. 6: (a) discrete HMM, (b) continuous HMM with one Gaussian and (c) continuous HMM with Gaussian Mixture.

	Real frames	Estimated frames								
		(a)			(b)			(c)		
		HD	MD	HD MD	HD	**MD**	**HD MD**	HD	MD	HD MD
Dissolve	58	29	30	44	46	**49**	**55**	12	52	52
	%	50.0	51.7	75.9	79.3	**84.5**	**94.8**	20.7	89.7	89.7
Fade	57	18	54	54	43	**51**	**49**	27	40	40
	%	31.6	94.7	94.7	75.4	**89.5**	**85.9**	47.4	70.2	70.2
Cut	4	0	2	2	3	**3**	**3**	0	3	3
	%	0	50.0	50.0	75.0	**75.0**	**75.0**	0	75.0	75.0

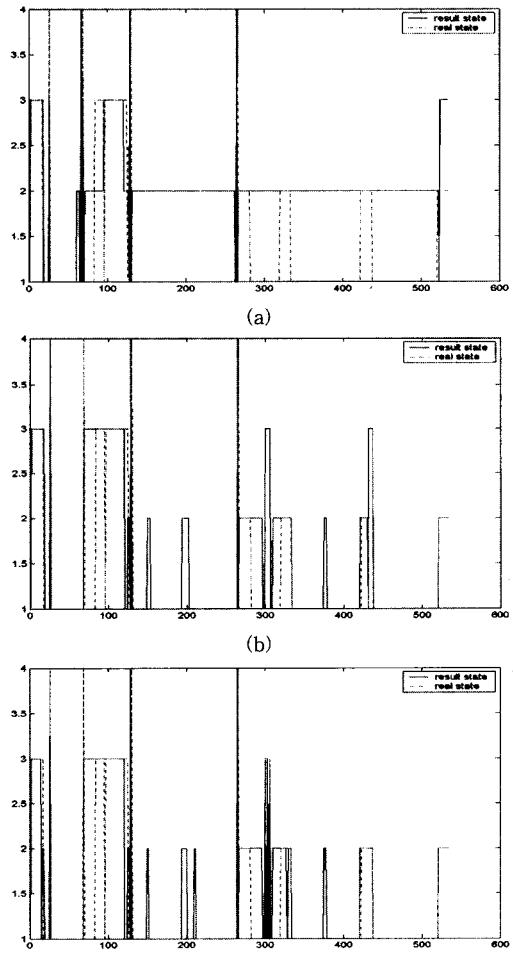
Table 3 Detection result using three different HMM and different image features

		Recall	Precision
Discrete HMM	HD	64.4 %	38.2 %
	MD	82.2 %	68.3 %
	HD + MD	86.8 %	73.2 %
Continuous HMM with one Gaussian	HD	86.5 %	40.2 %
	MD	**91.3 %**	**83.2 %**
	HD + MD	**91.0 %**	**84.7 %**
Continuous HMM with Gaussian Mixture	HD	70.3 %	50.6 %
	MD	88.6 %	73.4 %
	HD + MD	87.1 %	76.9 %



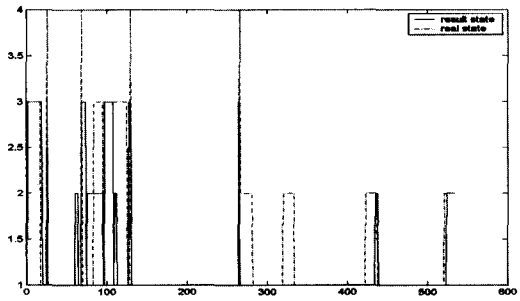
(a) (b) (c)

Figure 4 Detection results using (a) HD, (b) MD, and (c) HD + MD in discrete HMM. 1, 2, 3 and 4 of y axis represent the shot, the fade, the dissolve and the cut state in Fig. 3, respectively, and x axis shows the frame number

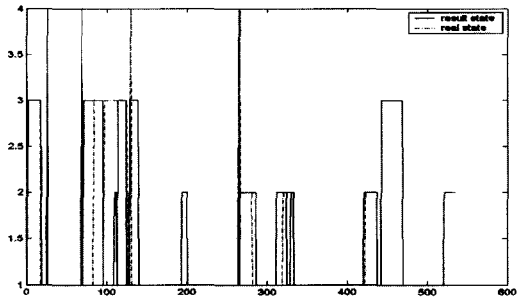


(a) (b) (c)

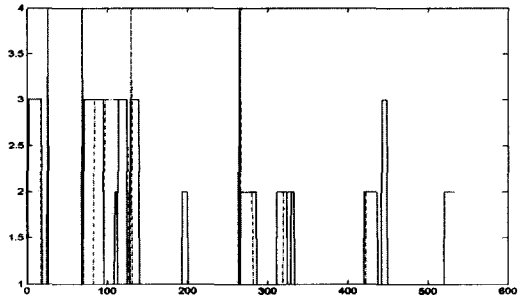
Figure 5 Detection results using (a) HD, (b) MD, and (c) HD + MD in continuous HMM with one Gaussian. 1, 2, 3 and 4 of y axis represent the shot, the fade, the dissolve and the cut state in Fig. 3, respectively, and x axis shows the frame number



(a)



(b)



(c)

Figure 6 Detection results using (a) HD, (b) MD, and (c) HD + MD in continuous HMM with Gaussian mixture. 1, 2, 3 and 4 of y axis represent the shot, the fade, the dissolve and the cut state in Fig. 3, respectively, and x axis shows the frame number

shows the frame number. Solid lines and dash lines represent the experimental resultant states and the real states, respectively. So we have a good experimental result if solid lines and dash lines are intersected as much as in many regions.

Table 2 shows the detection and classification performance of Fig. 4, Fig. 5 and Fig. 6. In this table, we know that the performance is good in the case of MD or HD + MD as feature and continuous HMM with one Gaussian as the HMM model.

Table 3 shows the detection and classification performance of the trained HMM in terms of recall and precision, where three different HMM models such as the discrete HMM, the continuous HMM with one Gaussian and the continuous HMM with Gaussian Mixture of two Gaussians, has been used, and three different types of image features such as HD, MD and HD + MD, are taken.

From the Table 3, we know that (1) the detection performance is the worst when only HD is taken in all HMM models, (2) the detection performance of taking the MD is better than that of taking the HD in all HMM models, (3) the detection performance of taking both HD and MD together is the best in all HMM models, (4) the continuous HMM model shows better detection result than the discrete HMM model, and (5) the continuous HMM with one Gaussian show better detection than the continuous HMM with Gaussian mixture.

In Table 3, precision values are relatively lower than recall values. This is caused by the fact that some data for shot state are overlapped to other data for dissolve data and data for moving camera, i.e. zooming and panning, shows similar pattern to data for gradual transition.

5. Conclusion

In this paper, we propose a new detection method of gradual transition in the MPEG compressed video using HMM. There has been other researches that are using HMM in order to detect gradual transition [4, 5], but they do not carry out the detection work in the MPEG compressed domain and/or use other features as well as color information. This paper shows that we obtain a good detection performance of gradual transition even we take only simple image features based on the intensity information. Experiment results show that we obtain the best detection performance of gradual transitions when we take a continuous HMM with one Gaussian model and with two image features.

Generally, setting of thresholds for determining the gradual transition is very difficult task and the detection performance is very sensitive to the choice of thresholds. From a viewpoint of practical applications, using the HMM eliminates the burden of setting the thresholds for the frame-to-frame distances and shows better result than using thresholds for detecting cuts and gradual transitions.

References

[1] Ullas Gargi, Rangachar Kasturi and Susan H. Strayer, "Performance Characterization of Video-Shot-Change Detection Methods," IEEE Transaction on circuits and systems for video technology, Vol. 10, No. 1, pp. 1-13, Feb. 2000.

[2] N. Gamaz, X. Huang and S. Panchanathan, "Scene change detection in MPEG domain," IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 12-17, 1998.

[3] W. A .C. Fernando, C. N. Canagarajah, D. R. Bull, "Fade, dissolve and wipe production in MPEG-2 compressed video," IEEE transaction on Consumer Electronics, Vol. 46, No. 3, pp. 717-727, Aug. 2000.

[4] Jong-Hyun Park, Soon-Young Park, Wan-Hyun Cho, "Video Scene Change Detection Using Hierarchical Hidden Markov Model," IPIU, pp. 196-201, 2001.

[5] John S. Boreczky, Lynn D.Wilcox, "A Hidden Markov Model framework for video segmentation using audio and image features," Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal processing, Vol. 6, pp. 3741-3744.

[6] Boon-Lock Yeo, Bede Liu, "Rapid scene analysis on compressed video," IEEE transaction on circuits and systems for video technology, Vol. 5, No. 6, pp. 533-544, Dec. 1995.

[7] Boon-Lock Yeo, Bade Liu, "On the extraction of DC sequence from MPEG compressed video," ICIP, Vol. 2, pp. 260-263, 1995.

[8] Lawrence R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceeding of IEEE, Vol. 77, Issue. 2, pp. 257-286, Feb. 1989.

최 성 민

1996년 3월~2000년 2월 포항공과대학교 컴퓨터공학과 학사. 2000년 3월~2002년 2월 포항공과대학교 컴퓨터공학과 석사. 2002년 1월~현재 (주)ITM 정보통신연구소 연구원

김 대 진

정보과학회논문지 : 소프트웨어 및 응용 제 31 권 제 2 호 참조

방 승 양

정보과학회논문지 : 소프트웨어 및 응용 제 31 권 제 1 호 참조