An Online Object-Based Key Frame Extraction Method for the Abstraction of Surveillance Videos

Yuan-Kai Wang
Department of Electronic Engineering
Fu Jen Catholic University
Email: ykwang@ieee.org

Li-Ya Wang
Department of Electronic Engineering
Fu Jen Catholic University
Email: liyawang@me.com

Yao-Ching Huang
Department of Electronic Engineering
Fu Jen Catholic University
Email: ffm1231@gmail.com

Ching-Tang Fan
Department of Electronic Engineering
Fu Jen Catholic University

***Abstract*—Key frame extraction is an important step for video surveillance. Key frames are able to inform users about the concept of an alarm event and guard environment more efficiently. Key frames can also be used for analysis of feature extraction, indexing and video retrieval. This paper proposes an object-based key frame extraction method for extracting representative frames of an alarm event. The method combines semantic features and weighted importance to extract key frames and devises an object features based formula to obtain better key frames that have clear object image. We also adopt Kalman filter to help predict objects' situations and extract key frames during the events.**

The proposed method has been verified by large amounts experiments that include testing 20 clips, implementing in a real-time mobile surveillance system. The experimental videos consist of single objects that are from MPEG-4 test video and so on. Our method proved by experiments not only can get clear and representative key frames but also reduce redundant key frame.

***Index Terms*—visual surveillance、video summarization、Kalman filter.**

I. Introduction

Intelligent video surveillance system is the current trend because it can be wildly used in guard field especially for public places such as train stations, banks and schools whose requirement for security is very high. It uses pattern recognition and computer vision techniques to extract, filter and index useful information from surveillance videos. It is efficient to prevent accidents and damages with reliable surveillance systems.

In the past decades, there are a lot of researches. Tremendous papers have concentrated on object detection, tracking and event detection [1, 2]. Many researches have concentrated on objects' activity, features, videos segmentation and so on.

Video surveillance generates and stores gigabytes video clips every day. The huge multimedia information is unstructured and hard to retrieve. There are main three processes of retrieving large amount video data: free browsing, text-based retrieval and content-based retrieval. In the user's perspective, content-based retrieval can supply friendly service for users such as interactive browsing, navigation with customized categories, and query by example by video abstraction and summarization. High-light sequences and key frames are the most broadly used in video abstraction and video summarization. High-light sequences aim to skim long videos to shorter sequences. Key frames are the still images extract from video clip that are relevant the content of a video. Key frame extraction has to produce minimized frames to represent a video and to economically abstract some of typical frames in a video. Users can survey the representative frames to understand events and save more time than to view the whole video. Good key frames also improve

high-level feature extraction and analysis, such as face recognition and behavior recognition. However, most key frames are generated with content-based methods by batch processing. These methods can't be adopted in real-time video surveillance because they can't produce key frames immediately [3].

Former methods have to collect all frames before the procedures of key frame extraction. These off-line procedures are not enough to provide the requirement for video surveillance because video surveillance is a lasting monitoring application and it has to continue detecting any possible events and send alarm messages day after day. For this critical requirement, key frames must be generated instantly from current input frame. Therefore online process is very important to video surveillance.

Classical approaches located key frames by calculating significant change among features of frames. The features of frames, such as color and motion, are obtained from all pixels of the frames, but are not from meaningful regions in the frames. These approaches do not consider semantic information in object level but low-level information in pixel level.

Recently, object-based key frame extraction methods have been proposed to improve the effectiveness of video abstraction [4]. It produces useful key frames for user because objects' information is the most meaningful in the surveillance videos. More accurate and optimized key frames can be chosen by calculating significant changes of shape, color and many others features.

This paper proposes an on-line object-based key frame extraction method for extracting representative frames of an alarm event. The method deals with not only single object but also multiple objects, and it takes Kalman filter to predict objects' behavior to extract key frames immediately.

Semantic features and weighted importance are adopted to extract expectative key frames in this paper. Semantic features are similar to human perception. This paper devises an object features based formula that combines semantic features and weighted importance to obtain better key frames which have clear object image.

We are going to discuss related works on key frame extraction in the next section. In Section III, the object-based key frame extraction method will be presented. The semantic features extracted to measure the representative of a frame is explained in Section IV. The Section V discusses experimental results of single-object experiments.And the Section VI gives conclusions and future works.

II. Related Work

For giant amount of video data, key frames are important information to describe the concept of the media. Key frame extraction helps video systems to index videos by several representative frames. It also helps users efficiently understand the content of a video. Therefore, key frame extraction is widely used in image query [5], video indexing [6], video browsing [7,8], video retrieval [9] and video summarization [10-12].

Motion, color, shape and edge have been the most important features used in key frame extraction. Choudary et al. [8] adopted edge information to extract key frames for summarizing the instruction videos of chalkboard presentation. Key frames were extracted according to the changes of edge information. Kollias et al. [12] adopted color and depth to extract key frames of stereoscopic video sequences. They used Lloyd-Max algorithm to cluster similar shots, and extracted key frames by optimizing each shot. Tainming Liu et al. [13] adopted motion feature to extract key frames. They proposed a triangle model for modeling motion patterns in video and determine key frames based on this model. The frames in the peaks of the model are chosen as key frames. Kim and Huang [4] used motion and contour to extract key frames. They observe giant changes of theses features to find out key frames. Mukherjee et al. [14] presented a key frame extraction method for integrating the decisions of different low-level features , such as motion with a combiner model. The method uses the Dempster-Shafer (DS) theory to combine randomness measures of features into one confident

value to estimate key frames. Fan and Liu [15] and Fan and Song [16] proposed a method to frame-based extraction and object-based video segmentation. They adopted motion information to extract key frames. The method is a two-stage key frame extraction. At first, it selects a set of key frames roughly. The set of key frames is trained by Gaussian mixture model [17] for object segmentation. The final key frames are re-found by model-based key frames analysis.

The low-level features such as color, motion, edge and shape are directly extracted from digital data of videos, but they can not be well comprehended human perception well. The gap between low-level features and human perception can be resolved by extracting semantic features. Therefore, the features proposed by this thesis will be semantic features.

There are two major methods to extract key frames. One is frame-based methods [13,14,18], and another is object-based methods [4]. The frame-based methods determine key frames by the change of the whole frame, but the object-based methods concentrate on the features' variation of each object. Above key frame extraction methods are frame-based method except the proposed method of Kim et al [4].

Erol and Kossentini [19] proposed an object-based video objects plane selection method with shape information in the MPEG-4 compressed domain. They computed the distance of video objects in two successive frames by Hamming and Hausdorff measures to detect significant changes in the shape. Then they determined the frame with the significant change to be a key frame of the object.

Kim and Huang [4] proposed an object-based video abstraction for video surveillance. The system finds out objects by subtracting last frame and current frame. The paper adopted and compared two features: motion and contour. The motion feature is Hu-moments [20] and the contour feature is Fourier descriptor. The decision to extract key frames is by event and by object's action change. The event is to compare the object numbers with current frame and last key frame whether they are equal or not. If the numbers are different, an event occurs and the frame would be a key frame. The object's action change is to measure the dissimilarity between the object's shape of the current frame and the last key frame. The system will generate a key frame if the difference is over a threshold. Finally, the paper adopted motion-based system because the contour-based feature is sensitive to variations of object boundary.

The traditional procedures of key frame extraction are off-line methods [12-13,14-16,18-19]. They take batch processes to generate key frames. However for the real-time requirement of video surveillance, on-line methods[4] to extract key frames immediately is very critical.

We are interested in the moving objects in videos and we expect our method can generate expectative key frames during events. Therefore, the proposed method in this paper is quite different from previous key frame extraction methods. We propose an online object-based key frame extraction approach that adopts use high-level features to represent objects in video. The method extracts key frames immediately by analyzing object's semantic features to determine representative frames. In addition, we propose a weighted importance approach to find out key frames. Depending on our method, it can obtain key frames with meaningful and clear object images.

III. The Proposed Approach

In this section, we will present the proposed key frame extraction method. First, we will introduce the proposed weighted importance approach that combines multiple features. In order to get key frames during events, we adopt Kalman filter to adaptively smooth the result of the weighted importance and analyze its prediction to extract key frames. Finally, we show the path analyses in many situations to verify the proposed on-line key frame method is robust.

A. Problem Formulation

Suppose an object o in the frame of time step t has a feature vector $\mathbf{X}_t^o = (x_1, x_2, x_3, ..., x_n)^T$, $x_i \in [0,1]$ or $x_i \in \{0,1\}$. A feature can be a discrete variable to represent the existence of an object characteristic, or

a continuous variable to illustrate the magnitude of the object characteristic. We assign a weight vector $\mathbf{W}=(w_1, w_2, \ldots, w_n)^{\mathrm{T}}$ which gives an importance coefficient w_i to a feature x_i . A criterion function $F(\mathbf{X}_t^o, \mathbf{W})$, gives the representative score of the object. The extraction of key frame $\tilde{K}$ is formulated as an optimization process as follows,

$$\tilde{K}=\left\{\tilde{K}_i=\arg_t \max\left\{R(t)=\frac{1}{l}\sum_{o=1}^{l}F(\mathbf{X}_t^o,\mathbf{W}), t-\delta\le t\le t+\delta\right\}, 1\le i\le M\right\} \quad (1)$$

where l is the number of objects, δ is a time interval, and $F(\mathbf{X}_t^o, \mathbf{W})$ is a linear weighting scheme defined by $\mathbf{W}^T\mathbf{X}_t^o$.

The representative score *R(t)* of a frame comes from the average scores of all objects in the frame. The optimization processes find *M* key frames from a video assumed to be *N* frames. If the time duration δ equals *N/2*, Equation (1) becomes a global optimization process which selects key frames with batch processing. An online key frame extraction is defined as a local optimization process that only a short time period δ is considered in order to adaptively choose the most representative frames.

B. Online Key Frame Extraction

Let the state parameter vector of a score at time t be denoted as y_t and its scores as observations z_t. The history of observations from time 1 to t is denoted as $Z_{1:t}$={ z_1 ,…, z_t }. The posterior distribution over y_t given $Z_{1:t}$ is expressed as p(y_t |$Z_{1:t}$). The Bayesian formulation of the posterior distribution can be marginalized as follows:

$$p(y_t|Z_{1:t}) \propto p(z_t|y_t)p(y_t|Z_{1:t-1}) = p(z_t|y_t)\int_{x_{t-1}} p(y_t|x_{t-1})p(y_{t-1}|Z_{1:t-1}) \quad (2)$$

where a first-order Markov property is considered. The p(z_t| y_t) is called the observation model. The observation, z_i, is conditionally independent of the history of the observations from 1 to t-1, $Z_{1:t}$, given the state y_t . The first factor in the integration is the transition model and the second is the current state distribution. Hence, we derive a recursive formulation.

There are three continuous variables, state y, state's velocity $\hat{y}$ and measurement z. The state y is a linear Gaussian distribution and the next state y_{t+1} must be a linear function of the current state y_t with some Gaussian noise. Assign the interval of measurement is Δ and assume the velocity is constant. The state update is presented by

$$y_{t+\Delta} = y_t + \Delta\hat{y} \quad (3)$$

We get a linear Gaussian transition model if we plus Gaussian noise for variation of velocity.

$$P(Y_{t+\Delta}=y_{t+\Delta}\,|\,Y_t=y_t, \hat{Y}_t=\hat{y_t}) = N(y_t+\Delta y_t, \sigma)(x_{t+\Delta}) \quad (4)$$

Kalman filter [21] is a recursive estimator and is a special case under the standard Bayesian network operations. It is composed by the transition model and the sensor model that are shown in below.

$$P(y_{t+1}|y_t) = N(Ay_t, \Sigma_y)(y_{t+1}) \quad (5)$$

$$P(z_{t+1}|y_{t+1}) = N(Hy_{t+1}, \Sigma_z)(z_{t+1}) \quad (6)$$

Where A and Σ_y are linear Gaussian transition model and Gaussian transition noise covariance, H is linear Gaussian measurement model and Σ_z is Gaussian measurement noise covariance, μ_t^- is the *a priori* state estimate and Σ_t^- is the a priori state error covariance for the next time step. The following two equations are prediction processing to predict next mean state μ_t^- and error covariance Σ_t^-.

$$\mu_t^- = A\mu_{t-1} \quad (7)$$

$$\Sigma_t^- = A\Sigma_{t-1}A + \Sigma_y \quad (8)$$

There are three steps in the correction process. The first step is to compute Kalman gain G_t which minimizes the *a posteriori* error covariance in the time step *t*. Equation (9) shows the Kalman gain equation, where G_t is the Kalman gain. Kalman gain means how much confident to the new measurement relative to the prediction. The second step shown in the Equation (10) gives an *a posteriori* state estimate μ_t from the z_t . The final step is to calculate the *a posteriori* error covariance estimate Σ_t in the time step *t*, which is shown in the Equation (11).

$$G_t = \Sigma_t^- H^T(H\Sigma_t^- H^T + \Sigma_z)^{-1} \quad (9)$$

$$\mu_t = \mu_t^- + G_t(z_t - H\mu_t^-) \quad (10)$$

$$\Sigma_t = (1 - G_t H)\Sigma_t^- \quad (11)$$

We use the prediction result, *a priori* state estimated value, μ_t^- to determine the key frames. We adopt secondary derivative to point out the peaks of the prediction trend. We select key frames from the peaks corresponding to the frames. And we use a threshold, T_s, to filter out the frames whose variation is small or object image is not clear. T_s is the threshold of the prediction scale. A peak set P of image sequence and p_i is the prediction value of the peak, $P=(p_1, p_2, \ldots, p_m)$, $p_i \in [0,1]$. Key frames are selected from the peak set P shown in Equation (12),

$$\tilde{K}_i = P_i \quad if \quad P_i > T_s,\ i \in [1,m] \quad (12)$$

C. Semantic Feature Extraction

We adopt three semantic features: Object region, skin color region and face. Face is important information of the moving object for surveillance system. Face information can help system to extract an object key frame with clear face. In this paper, we use Adaboost to detect whether a face exists or not. The face detection is achieved by the Adaboost algorithm [22]. The skin color is a useful feature to extract the key frames when the system can't detect the object's face. For example, the moving object goes backward the camera or the moving object is too small to detect face. The skin color region of the moving object is determined by Gaussian Mixture Model [23].

Moving object detection and tracking segments useful moving regions from the background and extract foreground objects (blobs) from these regions. The aim of moving object detection and tracking is to filter out non-important frames that have no detected objects. The frames which contain moving objects are less than 1% frames of surveillance videos. Therefore, moving object detection and tracking are the very effective filtering method for video surveillance system to skim surveillance videos.

In this paper, we adopt a Gaussian Mixture background subtraction approach to get foreground images. The background subtraction approach [24] builds the background images and subtracts the current frame to separate the regions of moving objects from background. The advantage of background subtraction approach is that remains the regions segmented from background completely. The connected component labeling assigns labels to the foreground regions. The output of moving detection is a binary image composed of foreground regions. The moving objects are modeled by color features and tracked by particle filter [25].

In order to detect skin color, we transfer RGB color space to YCbCr color space because skin color clusters well and not sensitive to light in the YCbCr color space. We use Gaussian mixture model to construct the skin color region [23]. The main reason we choose Gaussian mixture model is it represents the skin color probability distribution with only weights, means and covariance. It speeds up the computation of detecting skin color.

$$N(x_{pixel}) = \sum_{i=1}^{M} k_i \frac{1}{\sqrt{(2\pi)^N |\Sigma_i|}} \exp(-\tfrac{1}{2}(x_{pixel} - \mu_i)) \quad (13)$$

$$\sum_{i=1}^{M} k_i = 1 \quad (14)$$

The skin color is modeled M Gaussian distributions $N(x_{pixel})$. The Gaussian component i is from 1 to M, μ_i is the mean of the *i-th* component, and Σ_i is the covariance matrix of the *i-th* component. In equation (13), it shows the equation to calculate the probability of an object pixel x_{pixel} belonging to skin color. k_i is the weight of each Gaussian component. It means the component's importance of the skin color Gaussian Mixture Model. And the sum of all k_i equals 1. If the probability is more than a threshold t, the pixel belongs to the skin color.

In this paper, we adopt the cascaded Adaboost approach to detect face [22] in object regions. Adaboost is short for Adaptive Boosting. It is formulate by Yoav Freund and Robert Schapire. Adaboost approach takes a train of week features to identify the pattern which is face pattern in this paper.

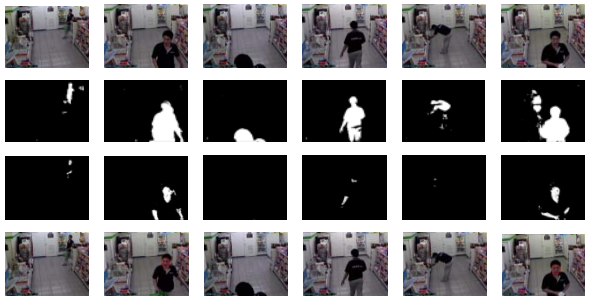
The cascade Adaboost approach is the first

real-time face detection approach presented by Viola and Jones. It uses a cascade of simple classifiers each trained to achieve high detection rates to improve detection rates and decrease the false positive rates. The first classifier will filter out amount of negative inputs. Therefore, the further processes can efficiently detect face pattern. The system is successful in detecting front view faces. It is efficient and effective to detect face in images.

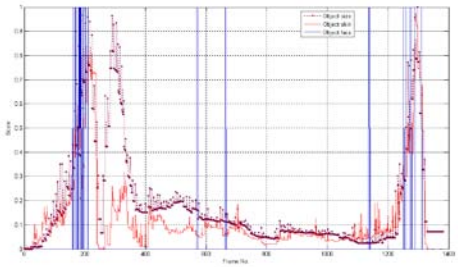
D. An Example

We illustrate this on-line key frame selection method by taking real-life video captured at a monitored convenient store. A man walked forth and back several times along the passage. The main idea is the key frame extracted by our method has large size of object region, large size of skin region and clear appearance of face. There, the feature vector $\boldsymbol{X}$ in this paper includes the three semantic features: the size of the skin color region belonging to the object, the size of the object region, and is the appearance of human face.

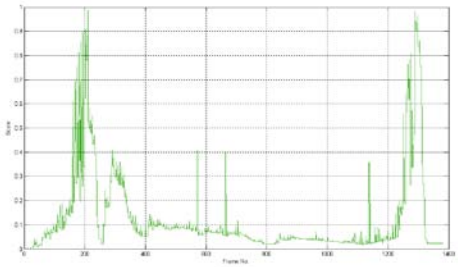
The first row of Fig. 1(a) gives a part of the image sequence. Three semantic features that are the regions of objects, the regions of skin color and face of objects of the corresponding frame are shown in the second, the third and forth rows of Fig. 1(a). Fig. 1(b) illustrates the normalized features. The original measurements of features are unstable and hard to be utilized for the decision of key frames because every feature's scale is different. Therefore, all features should be normalized. Fig. 1(c) is the criterion values of frames that are obtained by combining these features with weight vector $\boldsymbol{W}$=(1/3, 1/3, 1/3).



(a)



(b)



(c)

Fig. 1 Key frame selection by the weighted importance semantic features. (a) Original images and detected semantic features. (b) Three semantic features: region of object, skin region of object and object face. (c) Criterion function *F(W,X)* of the image sequence.

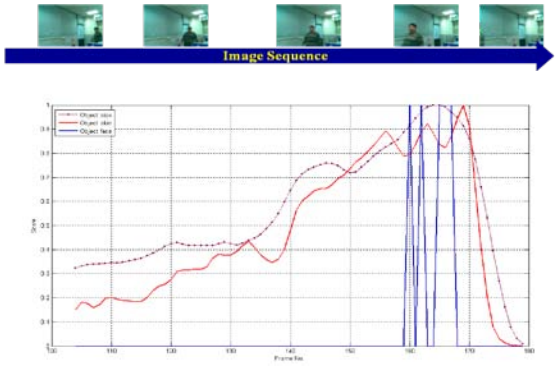
IV. Experimental Results

There are 20 video clips in the experiments to test our algorithm. Half of test videos consist of one moving object and the remaining videos consist of multiple objects. The testing video clips are collected from IBM S3 video clips, gait video clips and real-world convenient store surveillance video clips. The illuminate of test clips are very unstable, some videos are captured by analog CCD cameras, the quality of videos are low because high compressive rates. The resolution of all video clips is 320*240. We evaluate the proposed method by observing the different weight importance and different noise covariances of key frame extraction

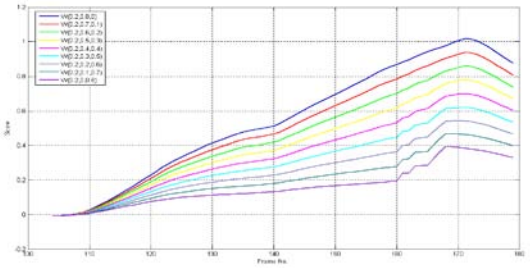
A. Weighted Importance Experiments

In this section, we observe different weight importance whether affect the extraction results of key frames. The weighted vector is $\boldsymbol{W}$*=(size, skin, face)*. We take two experimental videos for comparison. One is simple behavior and the other is complex behavior. We take 66 different combinations of weight vectors for each video

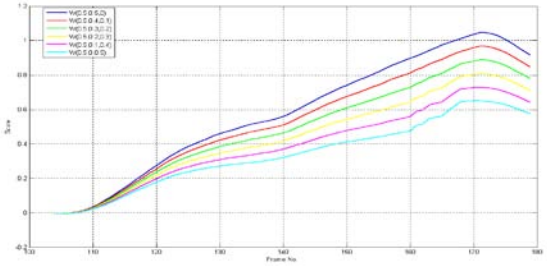
sequence. The simple case illustrated in Fig. 2, the object moves toward camera therefore the measurements of object size and skin increase. And the face can be detected when object is closed to camera. The measurement is shown in Fig. 2(a). We set the weight of size is 0.2 illustrated in Fig. 2(b) and 0.5 demonstrated in Fig. 2(c) individually, change the remaining weights of skin and face and the sum of three weights is 1. We find out the peaks of every prediction corresponding to different weight vectors are similar. So the key frames generated by different weight vectors are similar, too.



(a)



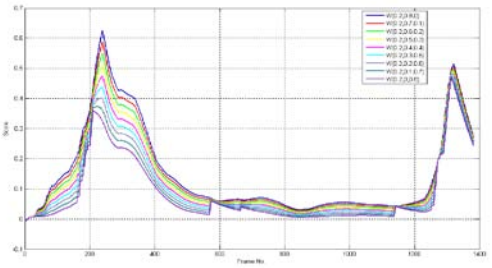
(b)



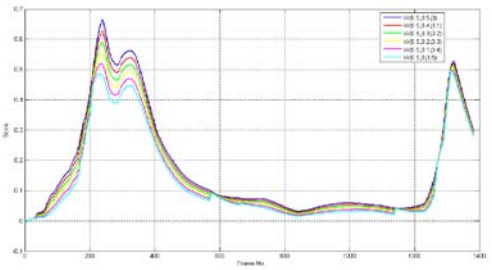
(c)

Fig. 2 Different weighted importance to simple case. (a) The measurement of features. (b) Fixing weight of size to be 0.2 and changing remaining weights. (c) Fixing weight of size to be 0.5 and changing remaining weights.

In the complex case, we also adopt the experiments of fixing weights of size 0.2 shown in Fig 3 (a) and 0.5 illustrated in Fig 3 (b). The trends of predictions are slightly different and they affect the number of key frames. It is caused by the moving object that walks forth and back twice and face only can be detected when the object move toward camera. Therefore different weight vectors doesn't affect when single object's behavior is simple. The weight vectors cause slightly key frames changing when the object's behavior is complex.



(a)



(b)

Fig. 3 Different weighted importance to complex case. (a) Fixing weight of size to be 0.2 and changing remaining weights. (b) Fixing weight of size to be 0.5 and changing remaining weights.

B. Noise Covariance Experiments

In this section, we change the scale of the measurement noise covariance *R* and the scale of the process noise covariance *Q*. Fig. 4 is the experiment to change the scales of measurement noise covariance. We only change measurement noise covariance: e^{-1}, 1 and 5 and set processing noise covariance is e^{-5}. We find although the curve of e^{-1} has more noise than others but it can predict the peak immediately than others. Fig. 5 is the experiment to change the process noise covariance

from e^{-5}, e^{-6} , e^{-7} and e^{-8} and set the measurement noise value is e^{-1} .

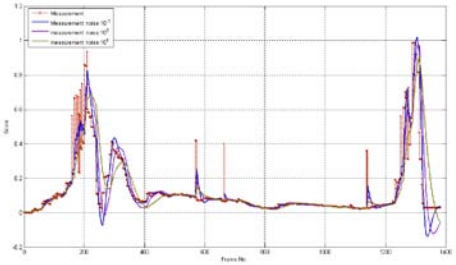


Fig. 4 Filtering results with different measure-ement noise covariance.

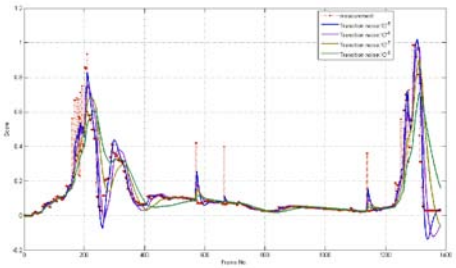


Fig. 5 Filtering results with different process noise covariance.

V. Conclusions

In this paper, we proposed a novel object-based key frame extraction method to abstract surveillance videos. The key frame extraction method not only adopts semantic features but also uses weighted importance. The optimized key frame is determined by the greatest importance, and the representative frames are chose by the peaks of the trend which is predicted with Kalman filter.

This method has been implemented in a real-time surveillance system. The system integrated object-based surveillance technique over 3G mobile communication network. The object-based surveilance technique generates object information, object-based key frame and object-based skimming video clip.

The furthermore object-based surveillance techniques can be developed for different monitor objects, such as vehicles. Object recognition can also be applied to summarize surveillance videos.

References

[1] M. Shah, O. Javed, and K. Shafique, “Automated visual surveillance in realistic scenarios,” *IEEE Multimedia*, vol. 14, no. 1, January-March 2007, pp. 30-39.

[2] Hao Jiang, S. Fels and J. J. Little, “Optimizing multiple object tracking and best view video synthesis,” *IEEE Transactions on Multimedia*, vol. 10, pp. 997-1012, October 2008.

[3] Yuan-Kai Wang, Li-Ya Wang, Yung-Hsiang Hu, “A mobile surveillance system with intelligent analysis” *Proceedings of SPIE on Electronic Imaging* , vol. 6821, San Jose, January 27-31, January 2008, pp. 68210I-01-08

[4] C. Kim and J.-N. Hwang, "Object-based video abstraction for video surveillance systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 1128-1138, 2002.

[5] Y.-H. Ho, C.-W. Lin, J.-F. Chen, and H.-Y. M. Liao, "Fast coarse-to-fine video retrieval using shot-level spatio-temporal statistics," *IEEE Transactions on Citcuits and System for Video Technology*, vol. 16, pp. 642-648, May 2006.

[6] P. M. Fonseca and F. Pereira, "Automatic video summarization based on MPEG-7 description," *Signal Processing: Image Communication*, pp. 685-699, 2004.

[7] C.-Y. Chen, J.-C. Wang, and J.-F. Wang, "Efficient news video querying and browsing based on distributed news video servers," *IEEE Transactions on Multimedia*, vol. 8, pp. 257-269, April 2006.

[8] C. Choudary and T. Liu, "Summarization of visual content in instructional videos," *IEEE Transactions on Multimedia*, vol. 9, pp. 1443-1455, November 2007.

[9] X. Zhu, A. K. Elmagarmid, X. Xue, L. Wu, and A. C. Catlin, "Insight video: toward hierarchical video content organization for efficient browsing, summarization and retrieval," *IEEE Transactions on Multimedia*, vol. 7, pp. 648-666, August 2005.

[10] A. M. Ferman and A. M. Tekalp, "Two-stage hierarchical video summary extraction to match low-level user browsering preference," *IEEE Transactions on Multimedia*, vol. 5, pp. 244-256, June 2003.

[11] G. Ciocca and R. Schettini, "Supervised and unsupervised classification post-prepr- ocessing for visual video summaries," *IEEE Transactions on Consumer Electronics*, vol. 52, pp. 630-638, May

2006.

[12] N. D. Doulamis, A. D. Doulamis, Y. S. Avrithis, K. S. Ntalianis, and S. D. Kollias, "Efficient summarization of stereoscopic video sequence," *IEEE Transactions on Citcuits and System for Video Technology*, vol. 10, pp. 501-517, June 2000.

[13] T. Liu, H.-J. Zhang and Feihu Qi, "A novel video key-frame-extraction algorithm based on perceived motion energy model," *IEEE Transactions on Citcuits and System for Video Technology*, vol. 13, pp. 1006-1013, October 2003.

[14] D. P. Mukherjee, S. K. Das, and S. Saha, "Key frame estimation in video using randomness measure of feature point pattern," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 612-620, 2007.

[15] L. Liu and G. Fan, "Combined key frame extraction and object-based video segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 869-884, 2005.

[16] X. Song and G. Fan, "Joint key-frame extraction and object segmentation for content-based video analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 905-914, 2006.

[17] R. Hammoud and R. Mohr. (2000, March) Gaussian mixture densities for indexing of localized objects in a video sequence. INRIA. [Online]Tech. Rep. RR-3905 [Online] Available: http://www.inria.fr/RRRT/RR-3905.html.

[18] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *IEEE International Confrence on Image Processing*, Chicago, IL,USA: IEEE, October 1998, pp. 886-870.

[19] B. Erol and F. Kossentini, "Automatic key video object plane selection using the shapeinformation in the MPEF-4 compressed domain," *IEEE Transactions on Multimedia*, vol. 2, pp. 129-138, June 2000.

[20] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Transaction on Information Theory*, pp. 179-187, February 1962.

[21] R. E. Kalman, "A new approach to linear filtering and prediction problem," *Transaction of the ASME-Journal of Basic Engineering*, pp. 35-45, March 1960 .

[22] P. Viola and M. J. Jonse, “Robust real-time face detection,” *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137-154, 2004.

[23] L. M. Bergasa, M. Mazo, A. Gardel, M. A. Sotelo, L. Boquete, “Unsupervised and adaptive Gaussian skin color model,” *Image and Vision Computing*, vol. 18, pp. 987-1003 2000.

[24] Y. K. Wang and C. H. Su, “Illuminant-invariant Bayesian Detection of Moving Video Objects,” *International Conference on Signal and Image Processing*, Hawaii, August 15-16, 2006, pp. 57-62.

[25] K. Nummiaro, E. Koller-Meier, L. V. Gool, “An adaptive color-based particle filter,” *Image and Vision Computing*, 2002, pp. 1-12.