Animal Tracking in Infrared Video based on Adaptive GMOF and Kalman Filter

Van Khien Pham*, Guee Sang Lee**

Abstract

The major problems of recent object tracking methods are related to the inefficient detection of moving objects due to occlusions, noisy background and inconsistent body motion. This paper presents a robust method for the detection and tracking of a moving in infrared animal videos. The tracking system is based on adaptive optical flow generation, Gaussian mixture and Kalman filtering. The adaptive Gaussian model of optical flow (GMOF) is used to extract foreground and noises are removed based on the object motion. Kalman filter enables the prediction of the object position in the presence of partial occlusions, and changes the size of the animal detected automatically along the image sequence. The presented method is evaluated in various environments of unstable background because of winds, and illuminations changes. The results show that our approach is more robust to background noises and performs better than previous methods.

Keywords : Gaussian mixture model | animal tracking | foreground detection | optical flow | Kalman filter

I. INTRODUCTION

Detection and tracking of moving object in digital videos are important information about the object locations and temporal of each frame. One of the main challenges in these applications is detected moving objects correctly. In addition, detection of wildlife in images and video has been an area of great interest. There are many recent methods for detection and tracking of moving objects, and all of them are difficult to solve noise, complex background, and changing animal size.

When animals are detected in infrared video, each frame in this video is the gray scale image that means object and background are nearly the same intensity. It is difficult to identify object accurately using background subtraction methods [1, 2, 9], they use the current frame minus the previous frame. Other methods use segmentation to detect animal but they take a lot of processing time, it is not suitable [13]. The mixture of Gaussians method is commonly used for the background modeling and it can identify any good shape of the density distribution, which is used in foreground detection in recent years [3, 8, 10, 17]. However, they have drawback that create noise detections with varying illumination environment. The role of the optical flow identifies the observation of animal movement [14]. Therefore, GMOF can be used to detect foreground pixels precisely.

In addition, object tracking also has many problems especially in wildlife field at night. When animals are occluded by something in some consecutive frames, recent tracking methods will not being predicted these positions.

Many recent object tracking algorithms create foreground region manually and sometimes track fail. The CAMSHIFT algorithm [13] usually uses color information for object tracking. To verify the center point of the probability distribution, CAMSHIFT applies the Mean Shift approach. This method is really good for single moving object, even multi moving objects, which contain some colors. However, this method has problem when moving object and background are similar color. Optical Flow tracking approach [14] finds the modification

This research was supported by the technological innovation R&D Program of SMBA(S2173771) funded by the Small and Medium Business Administration(SMBA), Korea.

*Student Member, **Member: Dept. of Electronics and Computer Engineering, Chonnam National University

Manuscript : 2016.03.08
Revised : 2016.03.22
Confirmation of Publication : 2016.03.23

Corresponding Author : Guee Sang Lee
e-mail : gslee@jnu.ac.kr

between moving objects and the background in the video. This method has good performance with fast velocity of single moving object. Therefore, when multi objects move with different velocity and very slowly, it is very difficult to identify object location accurately.

In the L1 tracker [15], they used trivial templates in the template dictionary such that its sparse linear grouping to solve the occlusion case and noising image in the target, nevertheless this method has problem with complex background. Sometime, they will update background part into template. Another method of Zhang et al. [16] used an appearance model based on extracted features from multi scale image with data independent, which compresses samples of background and foreground targets using the same sparse measurement matrix. When samples are compressed in all frames, the resolution is too small therefore they are difficult to identify the good position of target.

Kalman filter method is a good recursive filter [7], and offers an effective approach to compute the state estimation process, and create the minimum mean square error estimation. This method predicts the most possible object location in the current frame based on the results of targets tracking in the previous frame, and then search target location in the search area will continue to process the next frame. The main idea of Kalman filter is used to predict and update the object position.

To solve two issues of object tracking, we suggest a robust solution with an adaptive GMOF and Kalman model. This paper gives three main contributions:

- To remove noise detections using adaptive GMOF model in the complex environment.
- To change size of tracking object automatically and gradually.
- To predict the position of moving object using Kalman filter model in the effecting occlusion.

The full process of the proposed method is demonstrated in Figure 1. This paper is organized as follows: section 2 illustrates our proposed method. Section 3 shows the experiment method. Finally, section 4 gives our conclusion.

II. PROPOSED METHOD

1. Foreground extraction using GMM and Optical Flow

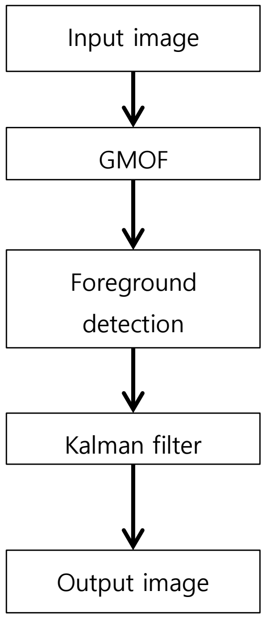


Fig. 1. The overview of our method

1.1 GMM background modeling

In this step, we implement the GMM approach to identify the background in a video sequence and study only the static background model. We consider the values of an individual pixel over time as a "pixel process", which is a time sequences of pixel values, for instance: scalars for gray scales or vectors for color images. In the video sequence, t is number of frames, what is acknowledged by a particular pixel, $\{x_0, y_0\}$, as shown in the following equation:

$$\{X_1, \dots, X_t\} = \{F(x_0, y_0, f): 1 \le f \le t\} \quad (1)$$

, where F is the sequence of consecutive frames.

The value of each pixel denotes a measurement of the radiance in the direction of the device of the first object, which intersects by the pixels visual ray. With a stationary background and static lighting, that value can be reasonably constant. When we accept that independent, Gaussian noise is suffered in the sampling process, its density can be defined by a single Gaussian density distribution centered at the mean pixel value.

The recent history of each pixel, $\{X_1, \dots, X_t\}$, is modeled by a mixture of K Gaussian density distribution. Given each new pixel value at the new frame, the probability of observing the current pixel value is

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * G(X_t, \mu_{i,t}, \Sigma_{i,t}) \qquad (2)$$

, where K is the number of Gaussian distribution, $w_{i,t}$ is the estimation of weight of the i^{th} Gaussian in the mixture model at time t. $\mu_{i,t}$ and $\Sigma_{i,t}$ are the mean and the covariance matrix of the i^{th} Gaussian. G is the Gaussian probability density function:

$$G(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t-\mu)^T \Sigma^{-1}(X_t-\mu)} \qquad (3)$$

K is represented by the available memory and computational power. At this time, K is used from 3 to 5. The distribution of recently observed values of each pixel in the frame is described by a mixture of Gaussians. A new pixel value used to update the model, which will be determined by one of the major components of the mixture model. The standard to select whether a pixel belongs to the foreground or background is the distance of pixel value to the mean of each Gaussian. A pixel is assigned to background when its value satisfies:

$$-2.5|\Sigma_{k,t}| \le (X_t - \mu_{k,t})^T (X_t - \mu_{k,t}) \le 2.5|\Sigma_{k,t}| \qquad (4)$$

Figure 2 shows a result of GMM background modeling in which the foreground label is extracted. As we can see, the detecting position of moving objects are accurately, however the labels still contain some noises and holes which will be improved in next step.

In the previous step, a foreground is extracted, but it includes a lot of noise in the object, therefore we are going to apply the morphology operation to improve object detection result. Firstly, we use the morphology erosion to eliminate the noise in the background:

$$L = \text{imerode}(L, E) \qquad (5)$$

, where L represents the foreground label, and E is the erosion mask.

After that the dilation is performed for binary image, and these areas of foreground pixels will grow in size:

$$L = \text{imdilate}(L, E) \qquad (6)$$

When morphology operators are implemented, foreground label reduces a noise and it covers all the regions of moving objects. The results are show in Figure 3.

Fig. 2. Results of GMM background modeling

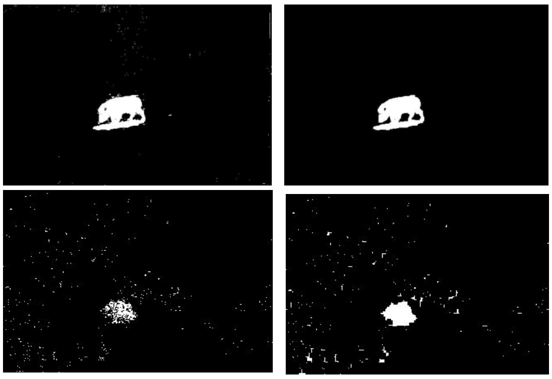


Fig. 3. Foreground extraction. The left column is the background modeling. The right column is the results after using morphology.

In this step, foreground extraction regions is drawn these red rectangles in each frame. We used to remove remain noise regions using Optical Flow model. In the initial frame, each pixel starts as being stationary. When motion is detected at a pixel, its nearby region is examined to define the optical flow for that pixel. Optical flow systems try to approximate the velocity of each pixel given a set of nearby frames. We withstand that the region matching to the animal will contain the largest amount of motion. These white lines denote velocity vector in Figure 4.

We focus the proposed approach on the Lucas-Kanade optical flow determination. This gradient based method is going to use the constraint of constant pixel intensities:

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (7)$$

The optical flow constraint has an equation as below:

$$I_x v_x + I_y v_y + I_t = 0 \quad (8)$$

, where I_x , I_y and I_t indicate the derivatives of the image function I(x, y) with x, y and t respectively. The vector V = (v_x , v_y) is the velocity vector in x and y direction.

Fig. 4. Remove noise. The left column is the foreground detection. The right column is the foreground detection using Optical flow.

2. Animal tracking using Kalman filter

Kalman filter method is applied to predict the state of a linear system where the state is expected to be Gaussian density distribution. On tracking objects of consecutive frames in image sequences, the continuity of the motion of the detected scene permits the prediction based on their previous directions of the image. As the moving state in the consecutive frames changes slight, the state parameters of KF are determined by the object location, its velocity, and its size of the object correspondingly. For each time step *t*, KF will give a prediction of the state at this time step:

$$X_t = A . X_{t-1} + W_t \quad (9)$$

, where X_t defines a vector demonstrating process state at time *t* and A defines a process transition matrix.

In our system of moving objects on digital camera, state defines a 4 dimensional vector [*x, y, dx, dy*], where *x* and *y* is the coordinates of the center object, and *dx* and *dy* denote its velocity. The transition matrix is simply

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Observation equation is the same as below:

$$Z_t = H.X_t + V_t \quad (10)$$

, where W_t and V_t represent movement and measurement noise vectors which follow Gaussian distribution p(w) ~ N (0, Q), p(v) ~ N (0, R) and H represents measurement matrix.

The Kalman filter gives the time update steps by projecting approximate error covariance forward one time stage:

$$X_t^{'} = A.X_{t-1} \quad (11)$$

$$P_t^{'} = A.P_{t-1}.A^T + Q \quad (12)$$

, where P_{t-1} defines a matrix indicating error covariance in the state prediction at time *t*, and *Q* represents the process noise covariance.

Then Kalman filter update steps are as follows:

$$K_t = P_t^{'}.H^T.(H.P_t^{'}.H^T + R)^{-1} \quad (13)$$

$$X_t = X_t^{'} + K_t.(Z_t - H.X_t^{'}) \quad (14)$$

$$P_t = P_t^{'} - K_t.H.P_t^{'} \quad (15)$$

The values of measurement matrix H, measurement noise covariance matrix R and process noise covariance matrix Q are shown in the following:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} , \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

KF algorithm begins with initial settings with K_0

and $P_0^{'}$. K_t represents the Kalman gain, which expresses the updating weights between the predictions and the new measurements as of the dynamic model.

When the animal is identified, the Kalman filter first forecasts its state at the current frame, and then it uses the newly identified animal position to correct its state. This process removes an error in the measurement of the object position. When the animal is not detected, this method depends on its previous state to forecast the animal current position.

II. EXPERIMENTAL RESULTS

In our application, we first show the classical GMOF algorithm. The tracking results are implemented based on Kalman Filter tracking of moving object. This method is implemented in Matlab 2014 on Intel Core i7−3770 CPU at 3.40 GHz, 8GB RAM and Windows 7 system.

The tracking approach is denoted as the following: firstly we use the GMOF approach to identify these animals and to remove noises in each video frame, and then the next position candidates will be predicted in the next frame by KF algorithm. In sequence, the experimental results of animal tracking are extracted the moving object in each video frame. Lastly, Kalman filter update function is used to estimate position of these animals.

We evaluated the performance of our object tracking algorithm with the collected animal video sequence on the internet (wild boar, water deer, tiger and wild life videos). A lot of previous papers used initial rectangle tracking manually; however, initial rectangle tracking of the proposed method is used to detect automatically based on GMM. We can see in Figure 5 and Figure 6, our tracking result can provide a good animal position. We also compare results with some good papers [15, 16] in Figure 7 and Table I, II, III, IV that object tracking results of our method are better than others.

Table 1. Tracking results of wild boar video

Method	Our method	L1-tracker[15]	RTCT[16]
Total frame	490	490	490
Good tracking frame	490	465	400
Precision (%)	100	94.9	81.6

Table 2. Tracking results of water deer video

Method	Our method	L1-tracker[15]	RTCT[16]
Total frame	814	814	814
Good tracking frame	811	781	671
Precision (%)	99.6	95.9	82.4

Table 3. Tracking results of wild boar video

Method	Our method	L1-tracker[15]	RTCT[16]
Total frame	710	710	710
Good tracking frame	704	663	593
Precision (%)	99.1	93.4	83.6

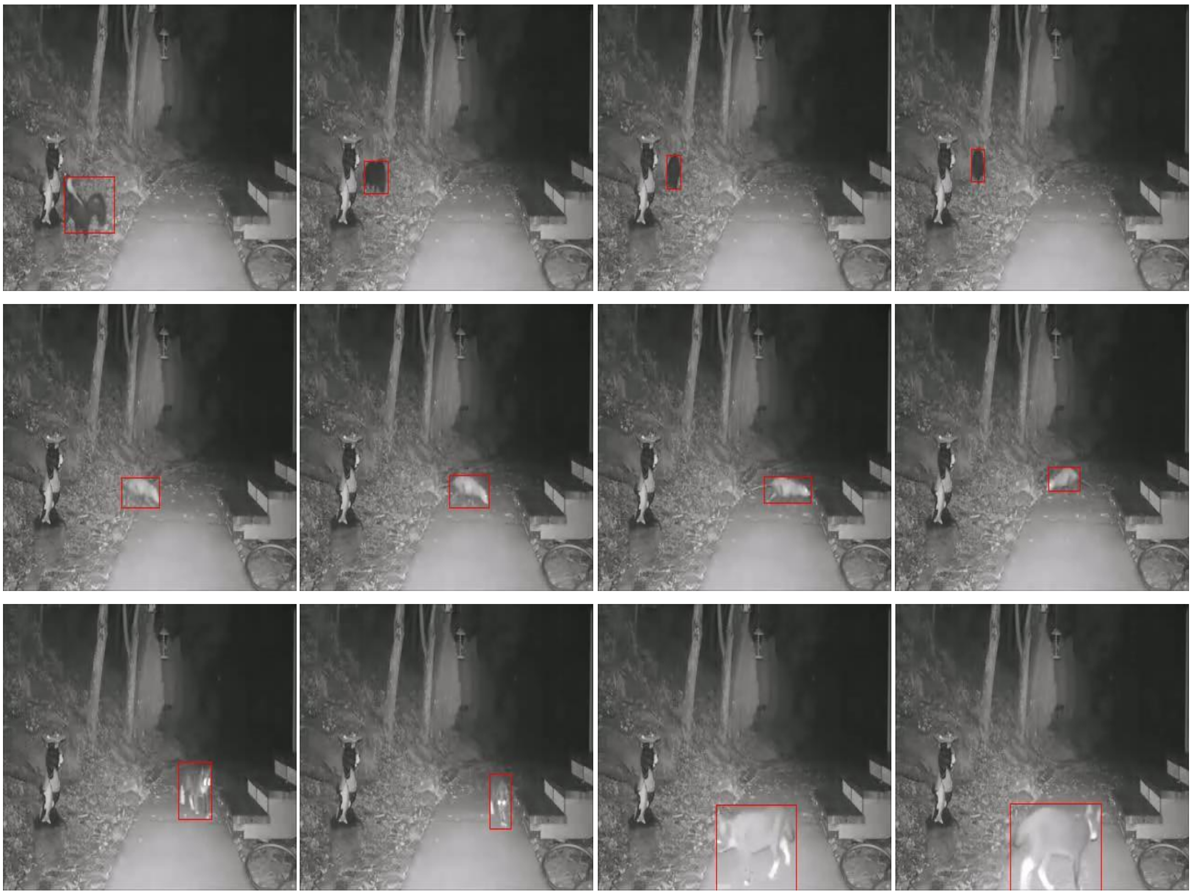
Table 4. Tracking results of water deer video

Method	Our method	L1-tracker[15]	RTCT[16]
Total frame	999	999	999
Good tracking frame	984	925	831
Precision (%)	98.5	92.6	83.2

III. CONCLUSION

In our paper, we present an object tracking method under static backgrounds for animal video sequences. Using GMOF and KF, we could get the good tracking result with infrared animal video. This method is deal with noise in the changing illumination scenes and animal occlusion. The tracking results are demonstrated by the first considering a moving animal where a complete tracking was successfully captured. The experimental results showed that the accuracy of our method is better than these previous methods. In the future works, we will focus on the issues in dynamic backgrounds and other animals.

(a) Wild life



(b) Water deer

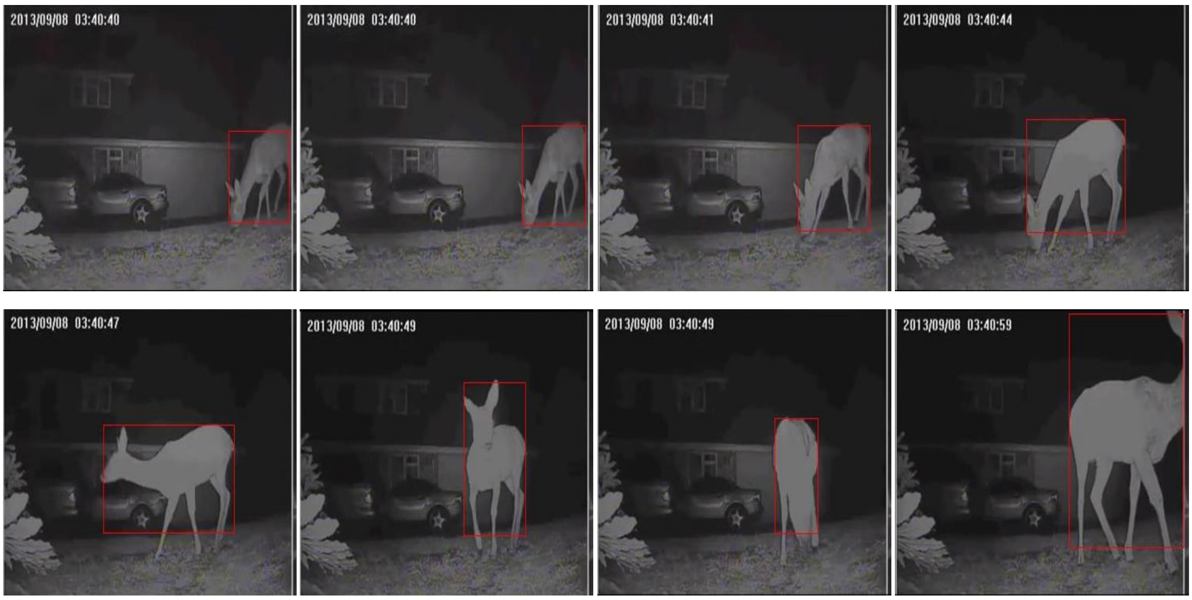
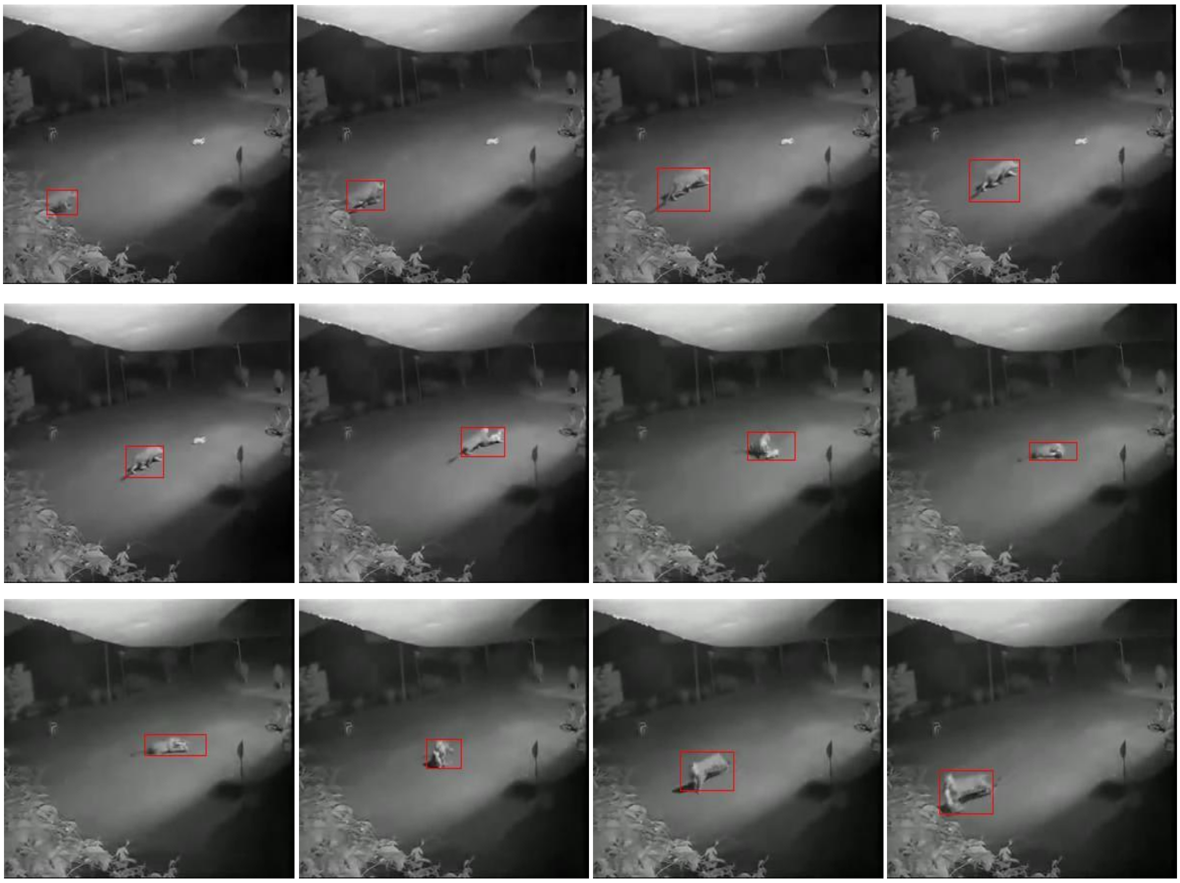


Fig.5. The tracking result of our method for animal sequence

(c) Tiger



(d) Wild boar

Fig.6. The tracking result of our method for animal sequence

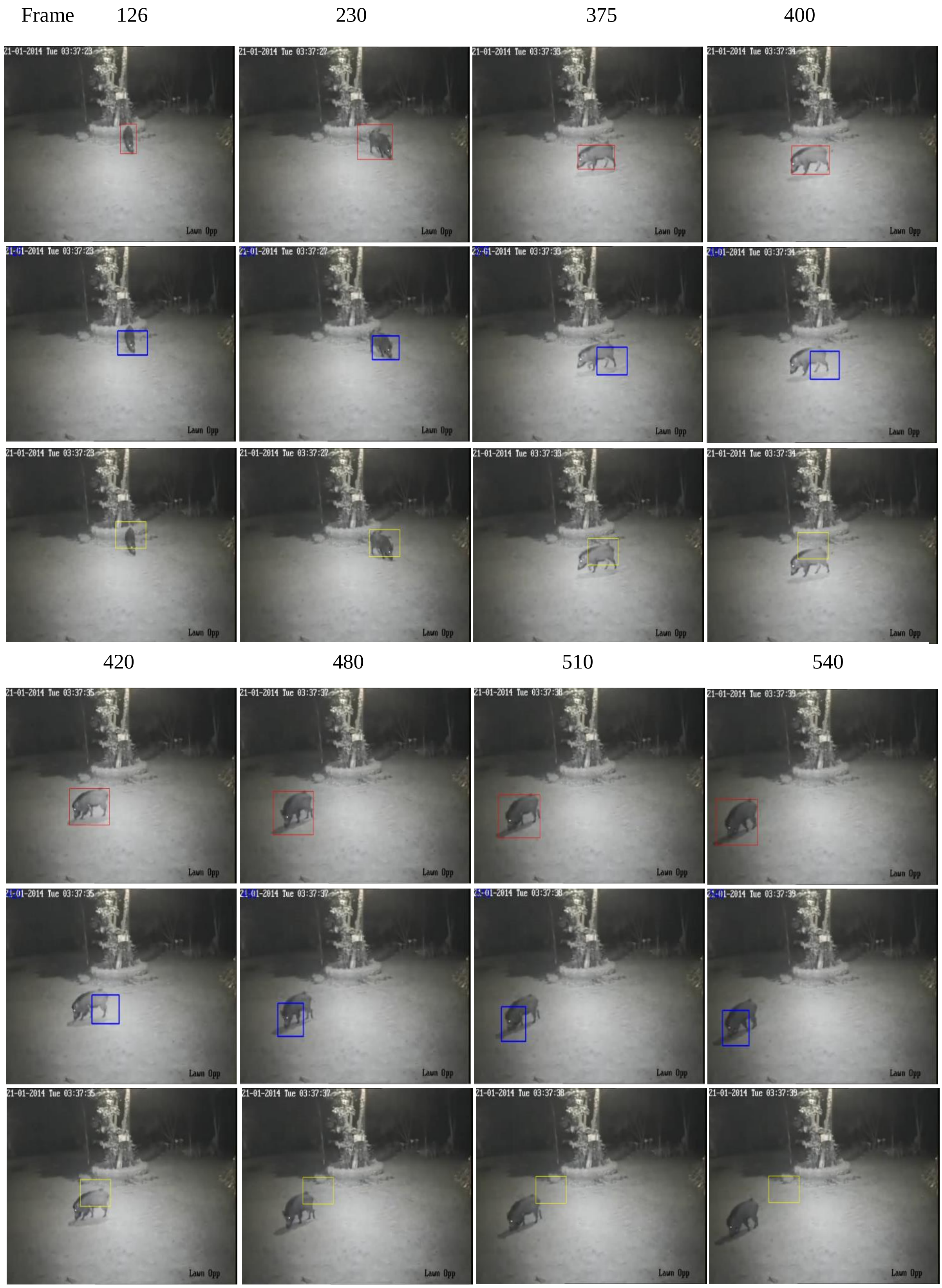


Fig.7. The tracking result for animal sequence; Red box: Our method; Blue box: L1−tracker [15]; Yellow box: Real−time Compressive Tracking [16]

References

[1] E. J. Fernandez-Sanchez, L. Rubio, J. Diaz, E. Ros , "Background subtraction model based on color and depth cues," Machine Vision and Applications, October 2013.

[2] D. Zamalieva, A. Yilmaz, J. W. Davis, "A Multi-transformational Model for Background Subtraction with Moving Cameras," ECCV, Vol. 8689, pp 803-817, 2014.

[3] S. Chris, and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," CVPR, vol. 2, pp. 529-551, pp. 246-252, 1999.

[4] Matthias Zeppelzauer, "Automated detection of elephants in wildlife video", EURASIP, July.2013.

[5] W.E.L. Grimson, Chris Stauffer, Raquel Romano, and Lily Lee. "Using adaptive tracking to classify and monitor activities in a site," In Computer Vision and Pattern Recognition 1998(CVPR98), Santa Barbara, CA. June 1998.

[6] Wren, Christopher R., Ali Azarbayejani, Trevor Darrell, and Alex Pentland. "Pfinder: Real-Time Tracking of the Human Body," In IEEE Transactions on Pattern Analysis and Machine Intelligence, July 1997, vol 19, no 7, pp.780-785.

[7] Christof Ridder, Olaf Munkelt, and Harald Kirchner. "Adaptive Background Estimation and Foreground Detection using Kalman-Filtering," Proceedings of International Conference on recent Advances in Mechatronics,ICRAM' 95, UNESCO Chair on Mechatronics, 193-199,1995.

[8] T. Bouwmans, F. El Baf and B. Vachon, "Background Modeling using Mixture of Gaussians for Foreground Detection", Recent Patents on Computer Science, vol. 1, no. 3, (2008), pp. 219-237.

[9] T. Ko, S. Soatto and D. Estrin, "Background Subtraction on Distributions", 10th European Conference on Computer Vision, Marseille, France, (2008).

[10] Y. Zhu and K. Fujimura, "Driver Face Tracking Using Gaussian Mixture model (GMM)", Proc. of IEEE on Intelligent Vehicles Symposium, (2003), pp. 587 - 592.

[11] W. Zhang, J. Sun and X. Tang," From Tiger to Panda: Animal Head Detection", IEEE Transactions on Image Processing, Vol. 20, pp. 1696-1708, 2011.

[12] T. Burghardt and J. Calic, "Real-time face detection and tracking of animals", 8th Seminar on Neural Network Applications in Electrical Engineering, pp. 27-32, 2006.

[13] Exner D. , Bruns E. , Kurz D. , Grundhofer A. , Bimber O, " Fast and robust CAMShift tracking", ICPR, 2010

[14] M. Lucena, J. M. Fuertes, N. Perezde la Blanca, "Evaluation of Three Optical Flow-Based Observation Models for Tracking", ICPR, Proceedings of the 17th International Conference on Pattern Recognition, 2004.

[15] Chenglong Bao, Yi Wu, Haibin Ling, Hui Ji, "Real time robust L1 tracker using accelerated proximal gradient approach", CVPR, Computer Vision and Pattern Recognition, 2012.

[16] Kaihua Zhang, Lei Zhang, Ming-Hsuan Yang, "Real-time compressive tracking", ECCV, Proceedings of the 12th European conference on Computer Vision, 2012.

[17] Jung Yeon Yeo, Guee Sang Lee, Background Subtraction based on GMM for Night-time Video Surveillance, Smart Media Journal, vol.4 no.3, pp. 50-55, 2015.

Authors

Van Khien Pham

He received B.S degree from Hanoi University of Sciences and Technology, Vietnam in 2012. Since 2014, he has been taking the M.S. course in Electronics & Computer Engineering at Chonnam National University, Korea. His research interests are mainly in the field of Image Processing, Computer Vision.

GueeSang Lee

He received the B.S. degree in Electrical Engineering and the M.S. degree in Computer Engineering from Seoul National University, Korea in 1980 and 1982, respectively. He received the Ph.D. degree in Computer Science from Pennsylvania State University

in 1991. He is currently a professor of the Department of Electronics and Computer Engineering in Chonnam National University, Korea. His research interests are mainly in the field of image processing, computer vision and video technology.