Australian Journal of Basic and Applied Sciences, 6(2): 190-198, 2012
ISSN 1991-8178

Real Time Hand Tracking System Using Predictive Eigenhand Tracker

Mohd Shahrimie Mohd Asaari and Shahrel Azmin Suandi

Intelligent Biometric Group,School of Electrical and Electronics Engineering,
Universiti Sains Malaysia, Engineering Campus,14300, Nibong Tebal, Pulau Pinang, Malaysia.

Abstract: In this paper, we present a real time vision based hand tracking system by combining predictive framework and appearance model. Due to the nature of hand motion which is flexible, erratic and easily varies in its appearance, the hand tracking from a single camera remains a complex problem. Here, we present a simple and efficient method to overcome such difficulties using the integration of Adaptive Kalman Filter (AKF) and Eigenhand method. After the hand state is quickly estimated from the AKF prediction, appearance model is employed to improve the earlier estimation. The appearance model is constructed based on a low dimensional eigenspace representation; the so called Eigenhand. During the tracking, the eigenspace constantly learns and adapts to reflect the appearance changes of the hand image. The experimental results demonstrate the effectiveness of the proposed tracking algorithm in indoor and outdoor environments where the target objects undergo large pose changes, lighting variation and partial occlusion. We achieve an average detection rate above 97% at the speed of 35fps.

Key words: Hand tracking, Adaptive Kalman filter, Eigenspace, Eigenhand, Hand gesture

INTRODUCTION

Nowadays, hand gesture has gained a lot of attentions in computer vision research area due to its versatile applications in inter-human communication (Ishida*et al.*,2010; Suk *et al.*, 2010; Elmezain*et al.*, 2010; Hee *et al.*, 2008; Feng *et al.*, 2011). An efficient hand gesture recognition system would providenatural and intuitive ways of developing human-machine interaction (HMI) in a wide spectrum. One of the crucial and basic ingredients in the hand gesture system is hands tracking, where hands must be localized in every image sequence.The effectiveness of hand tracking greatly depends on its ability to function reliably in real time so that an instant interaction would be achievable. To guarantee more accessible, such tracking system should not require the user to wear special clothes or cumbersome devices, for instance, the colored markers. Moreover, in vision based, hand tracking is a challenging task due to the difficulty in dealing with variousappearance variations mainly caused by shape and pose change, rapid and erratic motion, illumination changes and occlusion.

In the last decade, there are several methods that attempt to develop robust tracking techniques for varying video conditions such as partial, clutter environment, variations in shape and appearance, changes in illumination, etc. In (Imagawa *et al.*, 1998; Binh *et al.*, 2005) the moving hand is tracked by computing hand blobs and hand location is predicted using Kalman Filter. In their work, they assumed that the process and measurement noises are Gaussian and hand is moving in constant velocity. However, this assumption restricts the hand gesture movement in natural ways. (Isard and Blake, 1998) proposed particle filtering framework, which also showed applications to hand tracking. In their work, they adopted parameterized B-spline curves to model hand contours, and tracked hands by tracking the deformed curves. The system can maintain multiple hypotheses of the current object state and provides impressively robust tracking results. But unfortunately, the computational complexity of particle filter increases exponentially with the number of dimensions of the state space. Moreover, the hand contours are view dependent and this usually make the contour-based trackers constrain the viewpoint, thus difficult to adapt with non-rigid appearance that varies greatly during natural hand motion (Shan *et al.*, 2007).

Appearance based technique is another popular approach in dismantling visual tracking problems (Comaniciu*et al.*, 2004; Porikli*et al.*, 2006; Ho *et al.*, 2004). (Black and Jepson, 1996) proposed a tracking algorithm using eigenspace approach known as EigenTracking using a pre-trained view-based eigenbasis representation and a robust error norm. Instead of relying on the popular brightness constancy working principle, they used subspace constancy assumption for motion estimation. Although their algorithm demonstrated excellent experimental results, the eigenbases construction requires a set ofoff-line training data before the tracking task starts. Furthermore, their method assumes that certain factors, such as illumination conditions, do not change significantly as the eigenbasis, once constructed, it is not updated (Ross *et al*., 2008).

Corresponding Author: Shahrel Azmin Suandi, Intelligent Biometric Group, School of Electrical and Electronics Engineering, Universiti Sains Malaysia, Engineering Campus, 14300, Nibong Tebal, Pulau Pinang, Malaysia.
E-mail: shahrel@eng.usm.my.

In this paper, we proposed a real time hand tracking system which is robust in both indoor and outdoor environment with cluttered background. To ensure immersive interaction, the system does not require the user to wear any special clothes and no extra deviceis needed apart from the user's hand. In this work, the proposed hand tracking procedure is based on the interaction of a predictive framework and appearance model. Here, the main idea is motivated by the efficient computational means of Kalman filter to quickly estimate the hand state (Welch and Bishop, 2006)and the prowess of a linear subspace transformation (Turk and Pentland, 1991) to interpret the hand appearance model onto a low dimensional eigenspace representation. We advocate the use of adaptive parameters tuning into the original Kalman filtering to optimize its prediction when dealing with non-linear hand motion. Using the eigenspace representation as an appearance model, it provides a compact description of the object being tracked to facilitate object recognition and consequently, has an opportunity to rectify any inaccurate estimation from the earlier prediction. The eigenspace construction does not require any prior training phase but learn the eigenbases on the fly and constantly updated to account for appearance changes over times.

Hand Detection:

In this paper, we only consider on tracking one hand because our future goal is to extract the hand motion trajectory, which will be used as a trajectory based hand gesture recognition system. For the sake of simplicity,during the initialization, the actor makes significant movement using his hand andother part of the body is allowed to move only in small scale. During the detection phase, we assume the background image is static. The hand detection is then realized using a combination of skin color and motion features as describes in our previous work (Asaari and Suandi, 2010a).

Hand Tracking:

Observation Model:

We adopt Kalman filter technique to effectively predict the hand motion based on its location detected in the previous frame. It is necessary to derive a reliable observation model to ensure an efficient prediction. Naturally, hand motion varies greatly in terms of appearance, thus it is difficult to track the hand based on its geometry features such as contours. Here, we adopt non-geometric features using a fusion of skin color and motion cues to observe the state of hand position in thesuccessive frame by using the Region of Interest (ROI) based tracker as described in our previous works (Asaari and Suandi, 2010b) and assign the center position of the hand ROI to represent the actual observation state.

Dynamic State Model:

To develop a dynamic state, initially we measure the hand position and its velocity in a few frames, and then we define the state vector as in the following equation

$$x_t = (p_x(t), p_y(t), v_x(t), v_y(t))^T \tag{1}$$

where, $p_x(t)$, $p_y(t)$ are the hand position and $v_x(t)$, $v_y(t)$ are the velocity of hand in the t^{th} image frame. We define the observation vector, z_t to present the center position of hand ROI in the t^{th} frame from the observation model. The state vector,x_t and observation vector, z_t are related to each other by the linear stochastic difference equation vector as in the following equations

$$x_t = \mathbf{A}x_{t-1} + \mathbf{B}u_{t-1} + w_{t-1} \tag{2}$$

$$z_t = \mathbf{H}x_t + v_t \tag{3}$$

where, $\mathbf{A}$ is state transition matrix and $\mathbf{H}$ is observation matrix. $\mathbf{B}$ is the driving matrix that relates the optional control input, u_{t-1} to the state vector, x_t. The random variable, w_{t-1} is the process noise, which is assumed to be drawn from a zero mean. Since there is no additional signal used, (i.e., no extra sensor being used) we let the optional control input to be zero value. The random variable, v_t is the measurement noise where it is defined as the error between estimated location and actual measurement. The state transition matrix and observation matrix are defined as in the following equations

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{5}$$

The Kalman filter estimates the process state at the next time step and then obtains feedback in the form of measurements noise. As such, the equations fall into two groups: prediction equations and correction update equations (Welch and Bishop, 2006). The time update equations are presented in Eq. (6) until Eq. (7) and the correction updated equations are presented in Eq. (8) until Eq. (10)

$$\hat{x}_t^- = \mathbf{A}\hat{x}_{t-1} + \mathbf{B}u_{t-1} \tag{6}$$

$$\mathbf{P}_t^- = \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^T + \mathbf{Q} \tag{7}$$

$$\mathbf{K}_t = \mathbf{P}_t^-\mathbf{H}^T(\mathbf{H}\mathbf{P}_t^-\mathbf{H}^T + \mathbf{R})^{-1} \tag{8}$$

$$\hat{x}_t = \hat{x}_t^- + \mathbf{K}_t(z_t - \mathbf{H}\hat{x}_t^-) \tag{9}$$

$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t\mathbf{H})\mathbf{P}_t^- \tag{10}$$

where, $\hat{x}_t^-$ is the state estimates for the next time step, $\mathbf{P}_t^-$ is the covariance estimates *(priori)* and $\mathbf{Q}$ is the process noise covariance. In the correction update, $\mathbf{K}_t$ is Kalman gain, $\hat{x}_t$ is the improved state estimates, $\mathbf{P}_t$ is the improved covariance estimates *(posteriori)* and $\mathbf{R}$ is the measurement noise covariance. The process is repeated after each prediction and correction update pair to predict the new *priori*.

Adaptive Parameters Tuning:

From the Kalman gain update in Eq. (8), the measurement noise covariance, $\mathbf{R}$ and Kalman gain $\mathbf{K}_t$, are in inverse ratio. The smaller $\mathbf{R}$ is, the more heavily $\mathbf{K}_t$ weights the residual, thus the measurement is trusted more than the predicted result. On the other hand, as the *priori*, $\mathbf{P}_t^-$ is approaching zero, the Kalman gain, $\mathbf{K}_t$ weights the residual less heavily, and the predicted is trusted more than the measured result. Therefore, the system will get near optimal result if we can decide which one we will trust (Welch and Bishop, 2006). To achieve this goal, we need to appropriately determine the process noise and measurement noise covariance. The determination of the process noise covariance is generally more difficult as we typically do not have the ability to directly observe the process we are estimating. Therefore, in this procedure, we approximate the process noise to be described by $\mathbf{Q} = \sigma_w^2\mathbf{I}_{4\mathbf{x}4}$, where σ_w^2 is zero-mean Gaussian noise. In this case,we would hope that the process measurements are reliable. In the actual implementation of the filter, measuring the measurement error covariance is generally possible using some off-line sample measurements in order to determine the variance of the measurement noise. From our empirical results, we formulate measurement noise covariance to be $\mathbf{R} = \sigma_v^2\mathbf{I}_{2\mathbf{x}2}$, (where $\sigma_v^2 = 0.25$), in which σ_v^2 represents the variance or noise of hand position in both *x* and *y* directions. In both $\mathbf{Q}$and $\mathbf{R}$cases, $\mathbf{I}$is the identity matrix applied into the covariance calculation, so that the different state variables can be coded correctly into a single set of computation.

In real implementation, the $\mathbf{Q}$and $\mathbf{R}$covariance matrices might change in each time step during the prediction and measurement of the filter. In this implementation, the so called Adaptive Kalman Filter (AKF) is utilized to let the estimated parameters of the original Kalman filter change automatically in each time step. One such way to regulate the value of $\mathbf{Q}$ and $\mathbf{R}$ is by applying a weighting factor based on acceleration threshold value. In this case the hand acceleration is defined as in the following equation

$$\vec{a}_x = \frac{v_x(t) - v_x(t-1)}{\Delta t}, \; \vec{a}_y = \frac{v_y(t) - v_y(t-1)}{\Delta t} \tag{11}$$

where, $\vec{a}_x$ and $\vec{a}_y$ are the acceleration in *x* and *y* directions respectively. Then, we re-formulated theprocess noise covariance to be $\mathbf{Q} = \omega_q\sigma_w^2\mathbf{I}_{4\mathbf{x}4}$ and measurement noise covariance to be $\mathbf{R} = \omega_r\sigma_v^2\mathbf{I}_{2\mathbf{x}2}$. The scalar ω_q and ω_r are the weighting factor applied to process and measurement noise covariance, respectively, which is obtained from the following equations

$$\omega_q = \begin{cases} 1, |\vec{a}_x| \; or \; |\vec{a}_y| > T_a \\ \dfrac{1}{T_a}, Otherwise \end{cases} \tag{12}$$

$$\omega_r = \begin{cases} \dfrac{1}{T_a}, |\vec{a}_x| \; or \; |\vec{a}_y| > T_a \\ 1, Otherwise \end{cases} \tag{13}$$

where T_a is acceleration threshold value. Throughout the experiment, we use T_a equals to 10 which is obtained empirically.

Eigenhand Tracker:

It is still insufficient if we barely utilize AKF prediction to achieve a robust visual hand tracking. In the prediction scheme, the internal hand appearance is ignored, thus we do not have a clear description about the target object. In the case of numerous unrelated moving objects appear in the background, AKF prediction arrives at its limitation and finally will drift to wrong position. To reflect this limitation, we put an attempt on adapting appearance-based object representation using eigenspace approach, the so called Eigenhand. With this representation, we have a compact description of the object being trackedto facilitate object recognition and directly enhance the tracking task. We develop our eigenspace representation by applying Principle Component Analysis (PCA) and follow almost the same approach inspired by Turk and Pentland, (1991). For the reader's convenience, we summarize our proposed Eigenhand tracking algorithm in Figure 1, and the overall tracking scheme is depicted in Figure 2 for a complete illustration.

1. From detection phase, locate the hand image in the first frame. Draw an ROI window around the detected hand image and assign it as a reference model, ω_{ref} .
2. From the AKF predicted location, use it as a starting point to draw *M* sample windows $\{\omega_1, \omega_2, \omega_3, ., \omega_M\}$ as candidate images. Treat these candidate images as a training set and project them as new vectors, Ω_k on *k* dimensional $\mathbf{L}^k$ eigenspace. In every step interval, reconstruct $\mathbf{L}^k$ corresponding to new *M* sample windows.
3. Treat the reference model as a new input image to be matched with the candidate images. Transform it on the existing $\mathbf{L}^k$ eigenspace to obtain a reference vector, Ω_{ref} .
4. The final hand state can be estimated to be the window such that the corresponding candidate vector, Ω_k minimize the Euclidean distance to the reference vector, Ω_{ref} on the $\mathbf{L}^k$ space.
5. Replace the previous reference model, ω_{ref} with the latest estimated window, ω_k and include the previous reference model, ω_{ref} in subsequent eigenspace construction. Replicate step 2 until 5 in every time step.

Fig. 1: Eigenhand Tracking algorithm.

RESULTS AND DISCUSSION

To demonstrate the experimental performance of the proposed algorithm, we recorded eight sequences of the real- life scene in indoor and outdoor environment using a low resolution web camera. Each video consists of 352 x 288 pixel color images captured at 25fps. Executed using MATLAB 7, our algorithm runs at average 35fps on standard Intel Core Duo processor at 2GB RAM, under Windows 7 operating systems. We consider that the estimated hand position is accurate if it falls within 9 x 9 neighborhood of manually labeled"ground truth" position. Some of the tracking samples are illustrated in Figure 3 until Figure 10, and the summary of the tracking results are depicted in Table 1. The tracking rate is defined to be the ratio of successful estimated frames over total frames, and the averaged position errors are measured trough mean absolute error (MAE) in both horizontal and vertical directions.

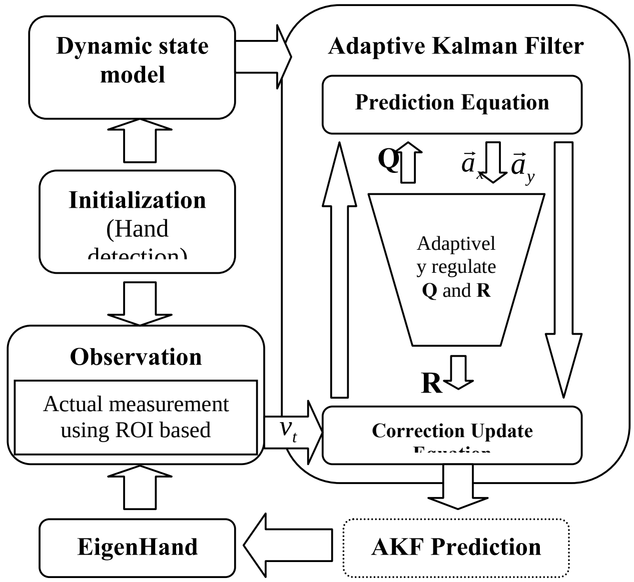


Fig. 2: Flow chart of the proposed hand tracking algorithm.

As a qualitative comparison, we ran three other tracking algorithms, Covariance based tracker (Porikli*et al.*, 2006), incremental PCA (IPCA) (Ross *et al.*, 2008), AKF based tracker (Asaari and Suandi, 2010b) on Indoor_1 sequence. The trajectory plot and some sample tracking results are depicted in Figure 11.Fromthe trajectory plot, it shows that the proposed procedure outperforms the Covariance and AKFtrackers. We observe that the Covariance tracker experiencing a slight drift off the target. This can be attributed to discriminating power of the covariance descriptor drops when the target object greatly resembles with background pixels due to occlusion with the identical color region. Next, we notice that AKF tracker performs poorly during a combination of drastic motion and severe occlusion as it undergoes a significant drift off the target. On the other hand, our proposed method provides a comparable tracking performance to the IPCA tracker. Even though both trackers manage to continuously track the target well, our proposed method gains an advantage on its low computational complexity thus realizing the real time implementation. Here, the computational complexity is dominated by the number of windows to generate candidate's region. With the benefit of AKF predictive, our tracker utilizes only 25 to 50 sample windows and manages to operate at the speed of 35fps.

The effectiveness of our algorithm can be attributed to several factors. One reason is that the fusion of skin and motion features in the Kalman filter observation model manages to simplify the background confusion, especially when the hand is tracked in the cluttered environment. On the other hand, the realization of adaptive parameter tuning in Kalman filter manages to minimize the difficulties in non-linear estimation process, especially when dealing with erratic hand motion. Furthermore, with the integration of Eigenhand, our tracker can perform object recognition and manages to quickly reflect any inaccurate estimation produced by the predictive framework. This fast reflection depends on the eigenspace update mechanism, which always blends the recently-acquired reference image to ensure the eigenspace representation is closed to the previous observation and at the same time minimizes the model from contamination.

Table1: Tracking results for eight real live tracking scenarios

Indoor Scene	TF	TR (%)	MAE_X	MAE_Y
Indoor_1	292	96.23	3.0992	3.7174
Indoor_2	328	96.65	2.0072	1.7137
Indoor_3	388	98.96	1.9027	1.5983
Indoor_4	299	94.98	2.9834	2.4782
Indoor_5	736	96.47	1.7496	2.6887
Outdoor Scene	TF	TR (%)	MAE_X	MAE_Y
Outdoor_1	402	99.50	1.5628	1.6957
Outdoor_2	352	97.16	1.4471	3.8462
Outdoor_3	394	97.46	1.9355	2.3272

MAE_X = Error in horizontal direction, MAE_Y = Error in vertical direction,
TF = Total frame, TR = Tracking rate

Fig. 3: Frame #211, #240, #263 and #278 from a scene of erratic hand motion and occlusion with face (Indoor_1).

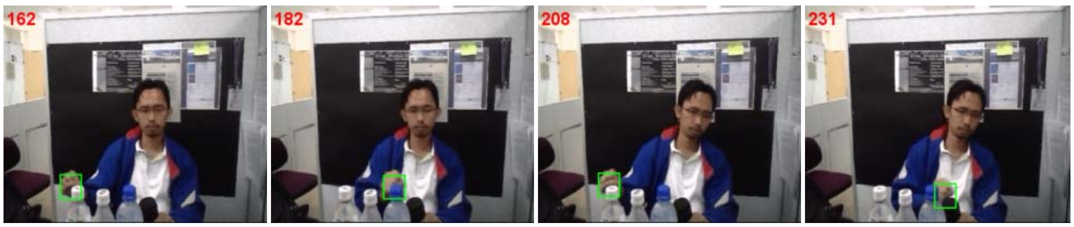


Fig. 4: Frame #162, #182, #208 and #231 from a scene of Partial occlusion behind several static objects (Indoor_2).

Fig. 5: Frame #199, #182, #212 and #214 from a scene of occlusion with skin colored region caused by left hand movement (Indoor_3).

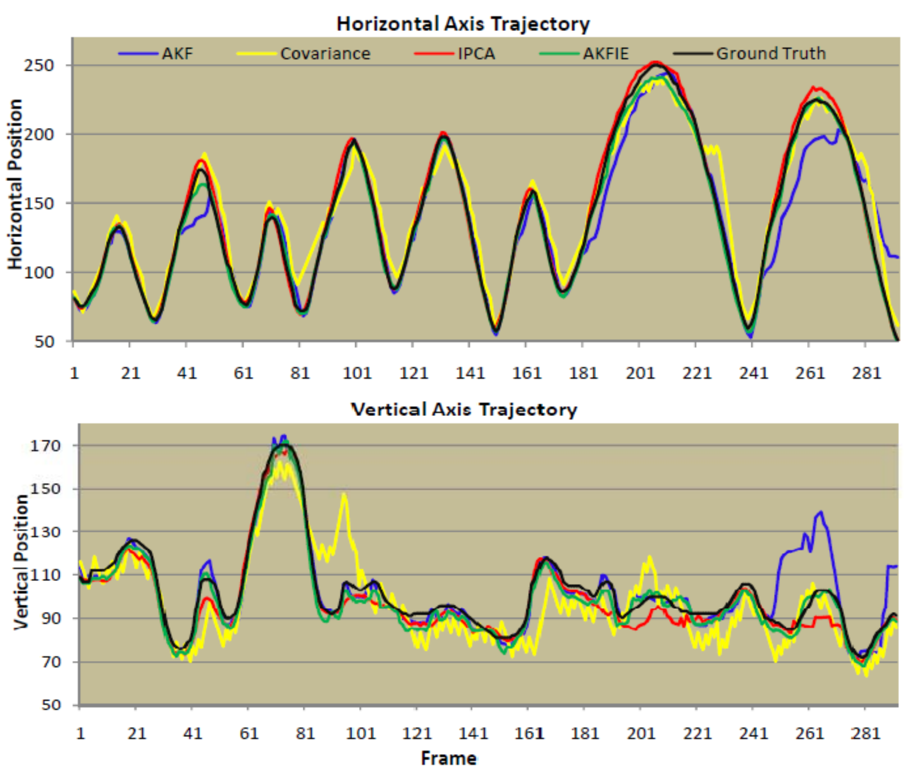
Fig. 6: Frame #104, #114, #128 and #144 from a scene of cluttered environment with dim lighting condition (Indoor_4).

Fig. 7: Frame #152, #303, #427 and #735 from a scene of cluttered environment with long term video sequence (Indoor_5).

Fig. 8: Frame #88, #92, #102 and #155 from a scene containing interference in the background caused by a moving person (Outdoor_1).

Fig. 9: Frame #106, #149, #169 and #235 from a scene of dim lighting caused by underexposed phenomenon (Outdoor_2).

Fig. 10: Frame #200, #223, #297 and #393 from a scene lighting variation caused by overexposed phenomenon(Outdoor_3).



(a)

(b)

Fig. 11: (a) Trajectory plot for horizontal and vertical axis, and (b) sample tracking results of proposed tracker (green box), Covariance tracker (yellow ellipse), IPCA tracker (red window), and AKF based tracker (blue dash box) for Indoor_1 sequence

Conclusion:

We have presented a simple and efficient integration of probabilistic estimation and appearance-based trackers for robust real time hand tracking system. Adaptive Kalman filter has been utilized to quickly estimate the hand state in video sequence. In the proposed algorithm, process and measurement noises covariance are adjusted adaptively using weighting factor based on acceleration threshold value to account for theprediction of non-linear hand motion.An appearance model based on eigenspace representation is employed and constantly updated to quickly response for the appearance changes, thereby facilitatestheimprovement on the earlier AKF prediction for an efficient tracking result. Our empirical results show the effectiveness of the proposed tracking algorithm for several real-life scenarios in indoor and outdoor environments with average detection rate above 97% at the speed of 35fps on average. In the future, we plan to integrate the current system with a gesture recognitionengine to produce a meaningful HCI system.

ACKNOWLEDGEMENTS

This work was supported by the USM Research University - Postgraduate Research Grant Scheme(RU-PRGS), (1001/PELECT/8034030) and the USM Postgraduate Incentive Research Grant (1001/PELECT/8021023).

REFERENCES

Asaari, M.S., S.A. Mand Suandi, 2010a, “Fusion of motion and color cue for hand detection and tracking”, World Engineering Congress, pp: 240-245.

Asaari, M.S.M and S.A. Suandi, 2010b, “Hand gesture tracking system using adaptive Kalman filter”, International Conference on Intelligent Systems Design and Applications, pp: 166-171.

Binh, N.D., E. Shuichi and T. Ejima, 2005. “Real time hand tracking and gesture recognition system”, International Conference on Graphic,Vision and Image Processing, pp: 111-115.

Black, M.J. and A.D. Jepson, 1996. “Eigentracking: Robust matching and tracking of articulated objects using a view based representation”, International Journal of Computer Vision, 26(1): 63-84.

Comaniciu, D., V. Ramesh and P. Meer, 2004. “Real-time tracking of non-rigid objects using Mean Shift”, IEEE Conference on Computer Vision and Pattern Recognition, pp: 142-149.

Elmezain, M., A. Al-Hamadi, R. Niese and B. Michaelis, 2010, “A robust method for hand tracking using Meanshift algorithm and Kalman filter in stereo color image sequences”, International Journal of Information Technology, 6(1): 24–28.

Feng, Z., B. Yang, Y. Chen, Y. Zheng, T. Xu, Y. Li, T. Xu and D. Zhu, 2011. “Features extraction from hand images based on new detection operators”, Pattern Recognition, 44(5): 1089-1105.

Hee, S.K., K. Gregorij and B. Ruzena, 2008. “Hand tracking and motion detection from the sequence of stereo color image frames”, International Conference on Industrial Technology, pp: 1-6.

Ho, J., K.C. Lee, M.H. Yang and D. Kriegman, 2004. "Visual tracking using learned linear subspaces", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp: 775-782.

Imagawa, K., S. Lu and S. Igi, 1998. "Color based hands tracking system for sign language recognition", International Conference on Face and Gesture Recognition, pp: 462–467.

Isard, M and A. Blake, 1998. "Conditional density propagation for visual tracking", International Journal of Computer Vision, 29(1): 5-28.

Ishida, H., H. Takahashi, I. Ide and H. Murase, 2010, "A hilbert warping method for handwriting gesture recognition", Pattern Recognition, 43(8): 2799-2806.

Porikli, F., O. Tuzel and P. Meer, 2006. "Covariance tracking using model update based on means of Riemannian manifolds", IEEE Conference on Computer Vision and Pattern Recognition, pp: 728-735.

Ross, D., J. Lim, M.H. Yang and R.S. Lin, 2008. "Incremental learning for robust visual tracking", International Journal of Computer Vision, 77(1): 125-141.

Shan, C., T. Tan and Y. Wei, 2007. "Real-time hand tracking using a mean shift embedded particle filter", Pattern Recognition, 40(7): 1958-1970.

Suk, H.I., B. Sin, S.W. Kand Lee, 2010. "Hand gesture recognition based on dynamic bayesian network framework", Pattern Recognition, 43(9): 3059-3072.

Turk, M and A. Pentland, 1991. "Eigenfaces for recognition", Journal of cognitive neuroscience, 3(1): 71-86.

Welch, G and G. Bishop, 2006. "An introduction to the Kalman filter", Technical Report, 95-041.