Hidden Markov Model - based Gesture Recognition with Overlapping Hand-Head/Hand-Hand Estimated using Kalman Filter

Yona Falinie Abdul Gaus
School of Engineering and Information Technology,
Universiti Malaysia Sabah, Jalan UMS, 88400 Kota Kinabalu, Sabah, Malaysia
yonafalinie@gmail.com

Farrah Wong
School of Engineering and Information Technology,
Universiti Malaysia Sabah, Jalan UMS, 88400 Kota Kinabalu, Sabah, Malaysia
farrah@ums.edu.my

***Abstract*--- In this paper, we introduce a hand gesture recognition system to recognize isolated Malaysian Sign Language (MSL). The system consists of four modules: collection of input images, feature extraction, Hidden Markov Model (HMM) training, and gesture recognition. First, we apply skin segmentation procedure throughout the input frames in order to detect only skin region. Then, we proceed to feature extraction process consisting of centroids, hand distance and hand orientation collecting. Kalman Filter is used to identify the overlapping hand-head or hand-hand region. After having extracted the feature vector, the hand gesture trajectory is represented by gesture path in order to reduce system complexity. We apply Hidden Markov Model (HMM) to recognize the input gesture. The gesture to be recognized is separately scored against different states of HMMs. The model with the highest score indicates the corresponding gesture. In the experiments, we have tested our system to recognize 112 MSL, and the recognition rate is about 83%.**

Keywords-skin segmentation YCbCr; feature extraction; Kalman Filter, gesture trajectory, gesture path, states, Hidden Markov Model

I. INTRODUCTION

In Malaysia, Malaysian Sign Language (MSL) is the usual method of communication for deaf and hearing impaired people. The deaf often have to communicate with other people through a sign language interpreter, however, that cannot always be done since the number and availability of interpreters is limited. This situation has created the need for a system to automatically interpret MSL. The system consists of four modules: collection of input images, feature extraction, Hidden Markov Model (HMM) training, and gesture recognition.

II. LITERATURE REVIEW

Many methods have been developed in hand detection from images or videos. Skin color is a strong clue in a video to distinguish hands from other objects, so color-based hand detection is possible [1]. However, there may be some other skin-colored objects, such as the face, the arm, so, only using skin color is not enough. Hand shape is another way to perform hand detection [2]. Viola and Jones [3] applied integral images and AdaBoost to face detection, which brought a breakthrough to that field. Similarly, a lot of work has been done using Harr-like features and AdaBoost algorithm [4].

After skin detection, the main issue in sign language is how to make hand gesture to be understood by computer. Most present works can mainly be divided into Data Glove and Vision- based methods as discussed in [5] and [6]. The Data Glove method uses sensor devices for digitizing hand and finger motions for multi parametric data. For Vision-based method, it only required one or two cameras, but the challenges are, it needs to be background invariant, lighting insensitive, allows person and camera independent to achieve real time performance [7]. Also, vision-based extraction of the hand feature has studied hand shape by matching hand silhouette with 3D Computer Graphic (CG) in a simple background [8]. However, it is difficult in unconstrained background. Extracting skin regions from a range of colour has also been done. But the drawback is that it cannot be completely done because skin colour depends on each signer or each situation.

As for dynamic hand gesture recognition, HMM are introduced. HMMs can be successfully used in processing both speech and two-dimensional sign language data, because their state-based nature enables them to capture variations in the duration of signs, by remaining in the same state for several time frames. According to Vassilia et.al [9], there are whole-word or word-based systems, where separate HMMs are trained for each word, and phoneme-based systems, where separate HMMs are trained for each phoneme. In either case, HMMs are trained to yield the maximum probability for the signal representing their respective word or phoneme. The phoneme in Sign Language (SL) could be the posture among a specific predefined set of postures, such as those appearing in Lian-Ouhyoung model [10]; or movement or a hold as Vogeler and Mataxas did in [11] and [12]. The main advantage in the phoneme-based systems is that the number of phonemes is limited in any language, including SL, as opposed to the unlimited number of words that can be made from phonemes. Thus, for large-scale applications, the most effective and commonly used method is the phoneme-based recognition, while for small-scale application, whole-word training is more appropriate.

978-0-7695-4668-1/12 $26.00 © 2012 IEEE
DOI 10.1109/ISMS.2012.67

III. COLLECTION OF INPUT FRAMES AND SKIN SEGMENTATION

For each video of MSL, the input frames is restricted from head to waist and in lab condition background. To improve processing time, the captured frame of 640 x 480 pixels are resized into 220 x 320 pixels standard Malaysian Sign Language [13]. As shown in Figure 1 and 2, in this step, the frames were selected by comparing the absolute difference between 2 consecutive frames and will stop after it represents one word. Then, only 30 frames (which we can assume to represent one word) that have the most absolute difference will be selected and further processed. From experimenting with gestures, it was found that 30 frames is the minimum number of frames that can accommodate all the meaningful data for all the gestures [13].

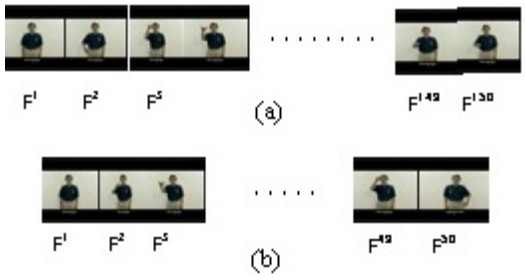

Figure 1: (a) Original gesture of MSL consists of 150 frames
(b) Gesture of MSL consists of 30 frames

After that, skin segmentation takes place. Generally hand detection comprises of subtracting the signers frame from the background frame. Finding hand regions using colour distribution is crucial because the nature of skin colour has its own unique value and can be processed. Skin colour model is used to detect pure hands images from complex background to be defined. Skin colour space such as HSL (Hue, Saturation, Luminance), RGB (Red, Green, Blue) and HSV (Hue, Saturation, Value) can be applied if there is minor variation in the luminance and tend to produce minimum overlap between skin colour and background colour distributions space [14].

The YCbCr colour space, however, behaves in such a way that the illumination component is concentrated in a single component (*Y*) while the blue chrominance component, *c(b)* and red chrominance component, *c(r)* are formed by substracing luminance from RGB red and blue component as shown in equation (1)

$$Y = 0.299R + 0.587G + 0.114R \quad (1)$$

where $c(b) = R - Y$; $c(r) = B - Y$

So we choose YCbCr as colour space for skin segmentation, which can be presented by Gaussian model, in a D-dimensional random variable *x* which is

$$N(x;\mu,\Sigma) = \frac{1}{(2\pi)\frac{D}{2}|\Sigma|\frac{1}{2}} \exp\left[-\frac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)\right] \quad (2)$$

where μ is the mean vector and Σ is the covariance matrix of the normally distributed random variable *x*. The model parameters are estimated from the training data, *n* using the following equations:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} c_i \quad (3)$$

$$\Sigma = \frac{1}{n-1}\sum_{i=1}^{n}(c_i - \mu)(c_i - \mu)^T \quad (4)$$

where $x = (r, b)^T$ and *c* is the skin pixel given in the color value of $r = c(r)$; $b = c(b)$

The original color image is transformed to a skin likelihood image before skin segmentation is carried out. This is done by transforming every RGB pixels in colour image to YCbCr color space and determine the likelihood value based on equation at (2).

After segmenting the frames using the properties of the YCbCr distribution for skin colour, two hands, face and other skin colored objects remain in the image. To simplify the algorithm, there are two assumptions that need to be addressed. We assume that only one or two hands are moving and needs to be tracked. Then, we assume that the scene for recognizing a sign language is relatively static and that the sign-language motion is acquired with a stationary camera.

IV. FEATURE EXTRACTION

A. Centroids

After skin segmentation has been undergone, we can assume that all that is left in the images are hand blobs, which is Right Hand (RH), Left Hand (LH) and head blobs, which is called Head (H). Figure 2 shows the example of skin segmented images.

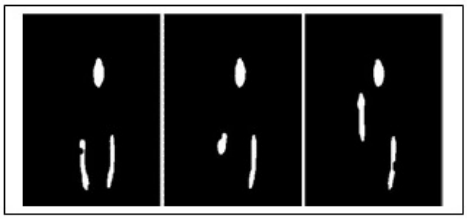

Figure 2: The skin segmented images indicated by white blobs

Then, the centroids of hand gesture are extracted while the signer is performing the sign language. The coordinate of the centroids, C_x and C_y can be expressed as in equations (5) and (6) and the area of blobs, *A* as in equation (7).

$$C_x = \frac{\int x \partial A}{A} \quad (5)$$

$$C_y = \frac{\int y \partial A}{A} \quad (6)$$

$$A = \int f(x) \partial x \quad (7)$$

where x is horizontal coordinate, y is vertical coordinate and $f(x)$ is the number of skin pixel in binary image. Figure 3 shows the centroids marked on the skin segmented blobs.

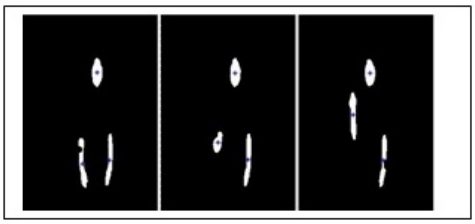

Figure 3: The blobs centroids indicated by blue marker.

The problem faced in marking the centroids is that the centroids will varies with posture, hand shape and orientation of the performer. This will cause problem to the feature vector of distance between hand and face as it varies among different person. To overcome this problem, the coordinate of centroid of area for each blob is shifted with respect to a reference point given for each of the face and both hands.

The shifting process is done by calculating the coordinate difference between the centroids of area of each blob in the first frame and the reference points given for each of the face and both hands separately. The reference points chosen are as shown in Table 1.

After obtaining the coordinate difference, the center of mass of each blob will be shifted according to the pixel difference computed.

TABLE 1: LIST OF REFERENCE COORDINATES FOR EACH REGION OF INTEREST (ROI)

	Coordinates	
ROI	x_{ref}	y_{ref}
H	110	110
RH	70	220
LH	139	270

The new coordinate for the center of mass is expressed in the equations (8) and (9).

$$x_{new} = x_{current} - (x_{initial} - x_{ref}) \quad (8)$$

$$y_{new} = y_{current} - (y_{initial} - y_{ref}) \quad (9)$$

So, in this section, we will have 3 sets of centroids, that is RH centroids, LH centroids and H centroids.

Some of the hand gesture trajectory involves overlapping hand with hand or hand with head. This will cause segmented blobs to fuse together (hand-hand/hand-head), resulting in only one or two centroids being detected, instead of three centroids, as shown in Figure 4.

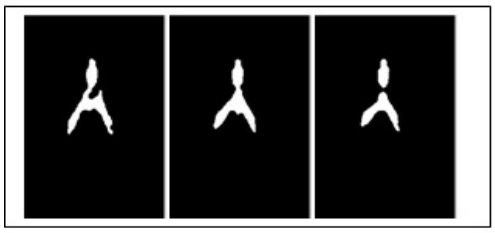

Figure 4: The segmented blobs fused together as caused by overlapping

Thus, in order to overcome this problem, the proposed method, Linear Kalman Filter has been used to estimate the actual centroids of the hands. Kalman Filter is an optimal estimation based on a system model and measurement model. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modeled system is unknown [15].

Theoretically, Kalman Filter is essential to estimate the actual centroids in overlapping cases, provided that the previous trajectory data before overlapping as an input is known to estimate the actual output. Treating the overlapping problem as a noise, Kalman Filter act to filter out the noise by estimating the actual trajectory using previous trajectory data, through equation (10).

$$\chi_k = \varphi_{k-1} + \xi_{k-1} \quad (10)$$

where ξ_{k-1} is a random vector modelling additive system noise, φ_{k-1} is the state transition matrix and χ_k is the state variable matrix.

Treating equation (10) as hand gesture trajectory, we will get (11) and (12) as gesture trajectory described by the position of hands (x_k, y_k) and the velocity of the hand, v sampled at discrete time, Δt.

$$x_k = x_{k-1} + v_{x_{k-1}} \Delta t \quad (11)$$

$$y_k = y_{k-1} + v_{y_{k-1}} \Delta t \quad (12)$$

Figure 5 shows the centroids of Kalman Filter estimation in overlapping problem.

Figure 5: The resulting centroids of Kalman Filter estimation in overlapping cases.

B. Distance between Hands and Face

The distance between the head and hands can be calculated using equation (13) known as the Pythagoras theorem. The distance calculated is scaled-down by 10 to make it favourable in the Hidden Markov Model.

$$distance = \sqrt{\Delta x^2 + \Delta y^2} \quad (13)$$

where Δx is the horizontal difference between face and hands and Δy is the vertical difference between face and hands. Figure 6 shows the illustration of distance measurement, which is measured in pixels.

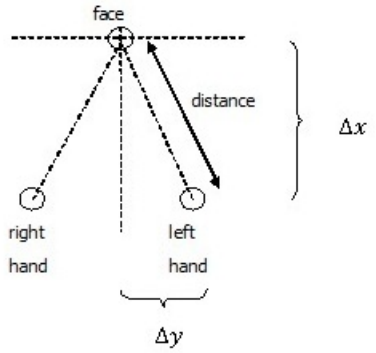

Figure 6: Distance between Hands and Face

C. Hand Orientation

The angle of the hand orientation is measured in the range of -90° to 90° from the horizontal axis. The angle is determined by enclosing the blob in an ellipse. The angle between the major axis of the ellipse and the horizontal axis gives the hand tilting angle, representing the orientation of the hand. The orientation value range from 1 to 19, representing the angle -90° to 90° at the division of 10°. The angle of hand orientation is shown in Figure 7.

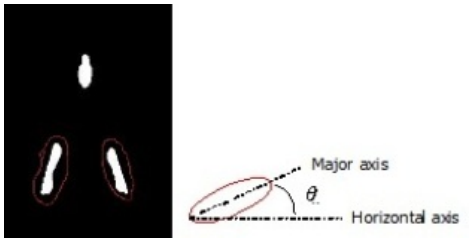

Figure 7: The angle of hand orientations

After feature extraction is determined, the centroids for hands are translated to specific gesture path, by using chain code. A gesture path has a spatio-temporal pattern which is consists of centroid points (x_{hand}, y_{hand}). So, the orientation as determined between two consecutive points form hand gesture path by using equation (14):

$$\theta_t = \tan^{-1}\left(\frac{y_{t+1} - y_t}{x_{t+1} - x_t}\right) \; ; t = 1, 2, ..., T - 1 \quad (14)$$

where T represents the length of gesture path. The orientation is quantized by dividing it by 20 in order to generate the codebook from 1 to 18 (Figure 8). The directional angle is divided by 20 in order to quantize the angle of 0^o to 360^o to the form of 1 to 18 accordingly as shown in Figure 8 (b). The hand motions are represented by quantizing the changes of the centroids of each hand to 1 of the 18 chain codes. A sequence consisting of centroids (explained in Section IV.A) the distance calculated between hand and face (explained in Section IV.B) as well as the hand orientation (explained in Section IV.C) is determined and then used as input to HMM. Table 2 shows the gesture chain code for the word '*mereka'* (in English, '*they'*)

TABLE 2: CODE OF GESTURE "*mereka*" BASED ON ITS DIRECTIONAL ANGLE

Directional Angle	Code
261.9937	13
168.9357	8
331.8982	15
147.2598	7
63.3762	3
73.0991	4

Hidden Markov Models is a mathematical tool widely used in statistical pattern recognition. HMM can be viewed as a combination of two random variables representing the states of a discrete stochastic process: the first one defines the hidden part whereas the second one defines how a given state produces a given symbol (the visible part). There are three main steps in HMM: Evaluation, Decoding and Training that can be solved by using Forward-Backward algorithm, Viterbi algorithm and Baum-Welch algorithm respectively.

Also, HMM has three topologies: Fully Connected (Ergodic Model) where any state in it can be reached from any other state, Left-Right model such that each state can go back to itself or to the following states and Left-Right Banded (LRB) model such that each state can also go back to itself or the following state only.

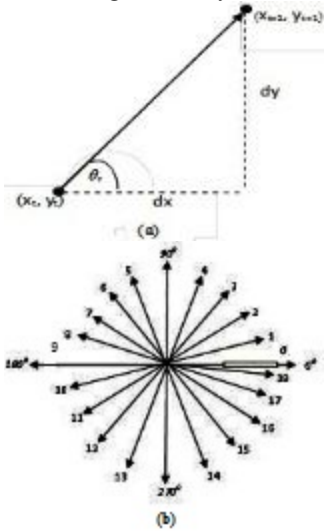

Figure 8: The orientation and its codebooks (a) orientation between 2 consecutive points (b) directional codebooks ranging from 1 to 18 including also zero codebook.

In this experiment, since the discrete vector contains a bunch of sequences of codebook from 1 to 18, we chose LRB model because it is restricted and simple for data training. This will enable match easier matching of the data to the model.

Each hidden state of the model has a likelihood of producing an output observation, *O*. In applications of this, kind the number of states and the topology used for the HMMs is important. We define the elements of an HMM as follows: *N* is the number of states in the model (which is in sign-language motion), and a gesture is defined as an order of states. For example, gesture "*mereka*" is mapped into a three states HMM as shown in Figure 9.

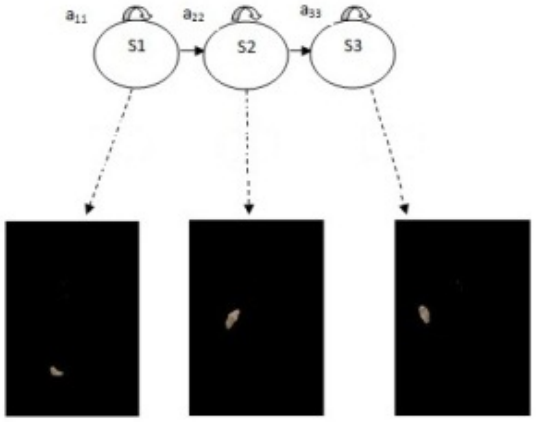

Figure 9: The Three States of HMM of gesture "*mereka"*

The HMM is parameterized as equation (15),

$$\lambda=(A,B,\Pi) \tag{15}$$

The first parameter, matrix *A* is the transition matrix. The system generated the transition matrix by random process followed by stochastic process to let the sum of each row to be equal to 1. Matrix A depends on the duration time *d* of states for each gesture such that *d* is defined as;

$$d=\frac{T}{N} \tag{16}$$

where *T* is the length of pure path and *N* represents the number of states that has a value of 3 in this example.

$$A=\begin{pmatrix} a_{ii} & 1-a_{ii} & 0 \\ 0 & a_{ii} & 1-a_{ii} \\ 0 & 0 & 1 \end{pmatrix} \tag{17}$$

Matrix B is the second parameter which is referred as the observation matrix. Matrix B is generated by random process followed by stochastic process in the system. Since HMM states are discrete, all elements of matrix B can be initialized with the same value for all different states.

$$B=\{b_{im}\};\quad b_{im}=\frac{1}{M} \tag{18}$$

where *i* run over the number of states and *m* is the number of discrete symbols respectively. The third parameter in the HMM is the initial vector Π which takes the following value;

$$\Pi = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T \tag{19}$$

This value is used because we use 3 states as the maximum numbers of the segmented parts of the gesture and in order to guarantee that it begins from the first state as shown in Figure 9.

V. EXPERIMENTAL RESULTS

In the experiments, the subjects, who uses a single hand or 2 hands to make hand gesture, is standing in front of any stationary background with normal lighting. We have selected 112 sign language of MSL. Each sign language has been performed 6 times by different signer, capturing a single hand or both hands moving in different directions with constant or time-varying hand shape. So, in total, we have 672 video sequence of MSL, of which, 560 video sequence were used for training and the remaining 112 videos were used for testing. The average performance rate using Kalman Filter estimation for overlapping hand-head/hand-hand is shown in Table 3. Then, gesture path, hand distance and hand orientation will become an input to HMM. We designed LRB topology with different states ranging continuously from 3 to 57 states. From a summary as stated in Table 4, the highest recognition of LRB topology was 83%. In general LRB topology with 8 states gave the best performance. The following criterion has been used to evaluate our results: The total testing data is considered as *test=112*, include gesture *hits* and *miss* data such that;

$$test = hits_j + miss_j;\ j = 3,4, \ldots, 57 \tag{20}$$

where *j* represents the number of states. The valid ratio for each specific HMM topology is calculated by using equation (21):

$$\eta_j = \frac{hits_j}{test}\cdot 100 \tag{21}$$

TABLE 3: AVERAGE PERFORMANCE RATE FOR OVELAPPING GESTURE USING KALMAN FILTER ESTIMATION

Hand Head Detection	Hand-Hand Detection
94.38%	84.91%

TABLE 4: GESTURE RECOGNITION RATE FOR HMM TOPOLOGIES WITH STATES RANGING FROM 3 to 57.

No. of states	*Data*		*Performance Rate (%)*
	Train	*Test*	*Topology (LRB)*
3	672	112	78.5214
5	672	112	70.5387
8	672	112	83.0537
15	672	112	74.1071
32	672	112	78.5714
57	672	112	82.1429

VI. CONCLUSIONS

This paper presents a system to recognize Malaysian Sign Language (MSL) from video sequence by the motion trajectory of a single hand or two hands moving using HMM which is suitable for real time application. Our database contains 672 video sequences of which 560 video sequences were used for training and 112 video sequences were used for testing. We choose Left Right Banded (LRB) topology and the number of stages ranging from 3 to 57 are applied and tested. The LRB topology with 8 states in conjunction with untrained testing data presents the best performance. Our results show that, the highest recognition rate was 83%, when tested with untrained testing data. In future, our research focuses on recognizing the MSL sentences instead of single MSL words to make it more useful for real time applications

ACKNOWLEDGEMENT

The author's wishes to thank MOHE Fundamentals Research Grant (FRGS0026-TK-1/2006) obtained through UMS for supporting this research.

REFERENCES

[1] Jones, M.J., Rehg, J.M, "Statistical Color Models with Application to Skin Detection". Int. Journal of Computer Vision 46(1), 81–96 (2002).

[2] Cootes, T.F., Taylor, C.J, "Active Shape Models: Smart Snakes". In: Proceedings of the British Machine Vision Conference, pp. 9–18. Springer, Heidelberg (1992).

[3] Jones, M., Viola, P., "Fast Multi-view Face Detection". Technical Report TR2003-96, MERL (2003)

[4] Kolsch, M., Turk, M., "Robust Hand Detection" In: Proc. IEEE Intl. Conference on Automatic Face and Geature Recognition, pp 614-619, May 2005

[5] K.Imagawa, "Colour-Based Hands Tracking System for Sign Language Recognition", Face and Gesture (FG1998), pp 462-467, 2000.

[6] K.Imagawa, H.Matsuo, R.Taniguch, and D.Arita, "Recognition of Local Features for Camera-based Sign Language Recognition System", Face and Gesture (FG2000), pp 849-853, 2000.

[7] Pragati Garg, Naveen Aggarwal and Sanjeev Sofat,"Vision Based Hand Gesture Recognition", World Academy of Science, Engineering and Technology ,49 ,pp 972-977, 2009.

[8] N. Shimada, K.Kimura and Y.Shirai, "Real-time 3D Hand Posture Estimation based on 2-D Apperance Retrieval Using Monocular Camera", Proc.Int.WS.on RATFG-RTS (satelliteWS od ICCV2001),pp 23-30,2001.

[9] Vassilia Pashaloudi and Konstantinos Margaritis, "Feature Extraction and Sign Recognition for Greek Sign Language" Proceedings of the 7th IASTED International Conference Artificial Intelligence and Soft Computer, pp.93-98, 2003

[10] R.-H. Liang and M. Ouhyoung, "A Sign Language Recognition System Using Hidden Markov Model and Context Sensitive Search", Proceedings of the ACM Symposium on Virtual Reality Software and Technology, pp. 59-66, HongKong, 1996.

[11] C. Vogler and D. Metaxas, "Towards Scalability in ASL Recognition: Breaking Down Signs into Phonemes", Gesture Workshop, Gif sur Yvette, France, 1999.

[12] C. Vogler and D. Metaxas, "Parallel Hidden Markov Models for American Sign Language Recognition", Gesture Workshop, Corfu, Greece, 1999.

[13] Yona Falinie bte Abdul Gaus, Farrah Wong, Kenneth Teo "Feature Extraction from 2D Gesture Trajectory in Malaysian Sign Language Recognition", International Conference 2011 4th International Conference on Mechatronics (ICOM), 17-19 May 2011, Kuala Lumpur, Malaysia.

[14] Hee-Sung Kim, Gregorij Kurillo, Ruzena Bajcsy, "Hand Tracking and Motion Detection from the Sequence of Stereo Color Image Frames," Industrial Technology 2008, ICIT 2008, IEEE International Conference, August 2008.

[15] Gary Bishop, "An Introduction to the Kalman Filter" University of North Carolina at Chapel Hill Department of Computer Science Chapel Hill, NC 27599-3175
www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf