A Unified System for Segmentation and Tracking of Face and Hands in Sign Language Recognition

George Awad[1], Junwei Han[2], Alistair Sutherland[3]
School of Computing, Dublin City University, Ireland
[1]*gawad@computing.dcu.ie* [2]*jhan@computing.dcu.ie* [3]*alistair@computing.dcu.ie*

Abstract

This paper presents a unified system for segmentation and tracking of face and hands in a sign language recognition using a single camera. Unlike much related work that uses colour gloves, we detect skin by combining 3 useful features: colour, motion and position. These features together, represent the skin colour pixels that are more likely to be foreground pixels and are within a predicted position range. We extend the previous research in occlusion detection to handle occlusion between any of the skin objects using a Kalman filter based algorithm. The tracking improves the segmentation by reducing the search space and the segmentation enhances the overall tracking process. The algorithm is tested on several video sequences from a standard database and can provide a very low error rate.

1. Introduction

Sign language (SL) is the primary communication method that deaf people use in their daily life. SL recognition has gained a lot of attention recently by researchers in computer vision. Basically, most of the related works to SL recognition techniques have been categorized as: glove-based methods [1] and vision-based methods [2]. SL recognition systems in general require the knowledge of the hand's position, shape, motion, orientation and facial expressions. Consequently, the hands and face must be tracked across the video frames to differentiate between the left and right hands and to detect occlusions that usually happen in real world SL conversations. Locating the hands in the image sequence is generally implemented by using colour, motion, and/or edge information. Colour cue is used by means of colour gloves [3] because it is a very easy method to segment the hands using predefined colour. However, it is an unnatural way to ask users to wear markers or colour gloves. The more challenging but natural way to locate the hands is by using skin detection [4]. Motion cues were used in [5] with the assumption that the hand is the only moving object on a stationary background.

Tracking the face and hands is particularly challenging in the presence of occlusion. Some systems avoid the occurrence of occlusion entirely by their choice of camera angle, sign vocabulary, or by performing unnatural signs [5,6]. However, occlusions between face and hands or between the two hands occur frequently in many signs in the real world. Previously [3], we used a colour glove to segment the hands and proposed a method to detect occlusion between the two hands using Kalman filter prediction. In this paper, we are more interested in improving SL recognition in natural conversation. This is our motivation in using skin detection techniques and handling occlusion between skin objects in a robust way to keep track of the status of the occluded parts, which helps to reduce the search space in the recognition phase. Also dealing with the segmentation and tracking problems as one unit simplifies the process of locating the skin objects, unlike other works that separate the two tasks of segmentation and tracking [7]. In this paper, we introduce a method for combining colour, motion and position information to segment skin objects and we extend our previous work [3] in occlusion detection to use it in keeping track of the occlusion status of the face and the two hands during tracking in a robust manner.

The second section presents an overview of our system. In section 3, we explain in details the different components of our algorithm. The experimental results on SL speaker videos are shown in section 4, and we conclude in section 5.

2. System overview

A block diagram for the system architecture is shown in Fig. 1. In general we track three objects: the face and two hands. Two main components form the proposed algorithm. The first component, skin

Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE

segmentation is responsible for segmentation of skin objects. The second component, object tracking, is responsible for matching the resulted skin blobs of the segmentation component to the previous frame blobs and keeping track of the occlusion status of the three objects.

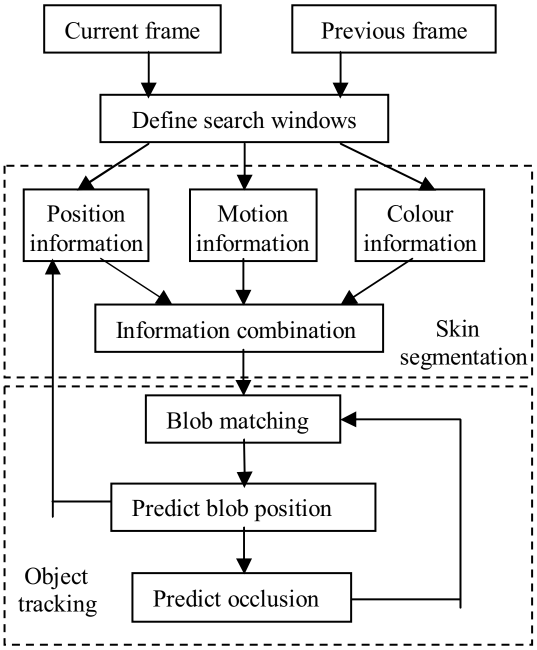

Figure 1. System architecture.

3. System components

3.1. Skin objects segmentation

In order to robustly detect the skin objects, we combine three useful features: colour, motion and position. Colour cue is useful because the skin has a distinct colour that helps to differentiate it from other colours. The motion cue is useful in discriminating foreground from background pixels. Finally, the predicted position of objects using Kalman filter helps to reduce the search space.

3.1.1. Colour information. Much related work in skin detection depends on simple skin models like colour ranges [7]. In order to detect different human skin colour variations, we trained a Support Vector Machine (SVM) [8] as mentioned in [9] and the output classifier is then applied on small search windows around the predicted positions of the face and hand objects and returns decision values representing how likely the pixels are skin. As the training of the SVM classifier is based on the first few frames, it is not optimum and it can miss some skin pixels. Therefore we propose another colour distance metric to take advantage of the prior knowledge of the last segmented object. This prior knowledge colour metric is denoted as dist(C_{skin}, X_{ij}), where C_{skin} is the median RGB colour vector of the previously segmented skin object, X_{ij} is the current pixel RGB colour vector in the search window in row *i* and column *j*, and *dist* is defined as the Euclidean distance between the two vectors. Finally, we normalize the values of the SVM classifier P_{svm}, and the prior knowledge colour metric P_{col}.

3.1.2. Motion information. Finding the movement information takes two steps. Firstly, motion detection, then next step, finding candidate foreground pixels. The first step examines the local gray-level changes between successive frames by frame differencing:

$$D_i(x,y) = |W_i(x,y) - W_{i-1}(x,y)| \qquad (1)$$

Where W_i is the *i*th search window and D_i is the absolute difference image. We then normalize D_i to convert it to probability values. The second step assigns a probability value $P_m(x,y)$ for each pixel in the search window to represent how likely this pixel belongs to a skin object. This is done by looking backward to the last segmented skin object binary image in the previous frame search window OBJ_{i-1} and applying the following model on the pixels in D_i:

$$P_m(x,y) = \begin{cases} 1 - D_i(x,y) & \text{if } OBJ_{i-1}(x,y) \equiv 1 \\ D_i(x,y) & \text{otherwise} \end{cases} \qquad (2)$$

In this way, small values (stationary pixels) in D_i that were previously segmented as object pixels will be assigned high probability values as they represent skin pixels that were not moved, and new background pixels with high D_i will be assigned small probability values. So simply, this model gives high probability values to candidate skin pixels and low values to candidate background values.

3.1.3. Position information. To capture the dynamics of the skin objects, we assume that the movement is sufficiently small between successive frames. Accordingly, a Kalman filter model can be used to describe the *x* and *y* coordinate of the centre of the skin objects with a state vector S_k that indicates the position and velocity. The model can be described as:

$$S_{k+1} = A_k S_k + G_k \quad (3)$$

$$Z_k = S_k + V_k \quad (4)$$

Where A_k is a constant velocity model, G_k, V_k represents the state and measurement noise respectively and Z_k is the observation. This model is used to keep track of the position of the skin objects and predict the new position in the next frame. Given that the search window surrounds the predicted centre, we translate a binary mask of the object from the previous frame to be centred on the new predicted centre. Then the distance transform is computed between all pixels in the search window and pixels of the mask. The inverse of this distance values assigns high values to pixels that are belonging or near the mask and low values to far pixels. The distance values are then converted to probabilities P_{pos} by normalization.

3.1.4. Information combination. After collecting the colour, motion and position features, we combine them logically using an abstract fusion formula to obtain a binary decision image $F_i(x,y)$:

$$F_i(x,y) = \begin{cases} 1 & \text{if } (P_{col}(x,y) > \tau)\ OR\ ((P_{svm}(x,y) > \gamma) \\ & \quad AND\ (P_m(x,y) > \upsilon)\ AND\ (P_{pos}(x,y) > \sigma)) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Where P_{col}, P_{svm}, P_m, and P_{pos} is the decision probability values of the simple colour metric, SVM classifier, motion, and position respectively, and τ, γ, υ, and σ are thresholds where σ is determined adaptively by the following formula:

$$\sigma = \frac{size((P_m(x,y) > \upsilon)\ AND\ (P_{pos}(x,y) \equiv 1))}{size(P_m(x,y) > \upsilon)} \quad (6)$$

The threshold σ determines the margin that we are searching into around the predicted object position. In Eq. (6) this is formulated by finding the overlapping between the predicted object position and the foreground pixels above certain threshold value. The other thresholds values are determined empirically.

3.2. Tracking and occlusion detection

3.2.1. Occlusion detection. In [3], we used the Kalman filter to detect the occlusion between only the two hands with colour gloves. In this paper, we extend this algorithm in two directions: first, to use it to detect occlusion between any of the face and the two hands. Second, we apply it to skin colour instead of colour gloves. In general, the algorithm uses the Kalman filter model to track the four corner points of the bounding box around the face and two hands. This model can predict in the next frame the positions of these four corner points. Accordingly, we check to see if there is any overlap between any of the bounding boxes in the next frame. If there is an overlap, we raise an occlusion alarm corresponding to the two bounding boxes that will overlap. If in the next frame, the number of detected skin objects is less than the current frame objects and an occlusion alarm was raised previously, we conclude that occlusion happened. On the other hand, if the number of detected skin objects decreases and no occlusion alarms were raised, then this means that one or more skin objects are hiding.

3.2.2. Tracking. As shown in Fig. 1, the tracking process starts by first constructing search windows around each of the tracked objects. When two or more objects are occluded, they are treated as one object and one search window is constructed around their position. Given that the search windows are constructed, we segment the skin objects as mentioned in section 3.1. Next, connected regions are labeled after removing noisy small regions.

Using the number of detected skin objects and the occlusion alarms as discussed in section 3.2.1, we maintain a high-level understanding of the status of the current frame with respect to the occlusion status. For example, if we detected 1 object and occlusion alarm between the face and left hand is raised, then we conclude that the face and left hand are occluded and the right hand is hiding. This technique can be extended to handle all 7 situations of occlusion status: separate face and two hands, face and 2 hands occluded, separate hand with face and hand occluded, face and hiding hands, face and hands all occluded.

The final step in the tracking part is the blob matching where the previous frame blobs are matched against the new frame blobs using the knowledge of the high-level occlusions status we described above. The matching is done using the distance between the previous objects centres and the new objects centres.

4. Experimental results

We tested the proposed system on different video sequences from the ECHO database (SL video database http://www.let.ru.nl/sign-lang/echo/) for different SL speakers under different lighting conditions and with different occlusion conditions. Fig.

2 illustrates 2 examples of the tracked images. To quantitatively evaluate the performance, we manually labeled 600 frames to construct the ground truth of the bounding boxes of the skin objects. Out of 600 frames, 237 frames included occlusions. As in [10], we measure the error in the position (x, y) of the centre of the bounding box. Table 1 shows the average error in x and y directions respectively and the average error of the tracking process, i.e. when skin object is incorrectly identified (ex. left hand identified as right hand). As shown in table 1, the algorithm accuracy is very high as the maximum error is about 6 pixels, and in terms of tracking errors, only 39 frames had objects identified incorrectly, where 37 frames of them, the error where due to occlusions, and only 2 frames had errors in absence of occlusion. From the results, we can conclude that the tracking is very robust to occlusions, as out of about 40% occluded frames, the error percentage was about 6.5%.

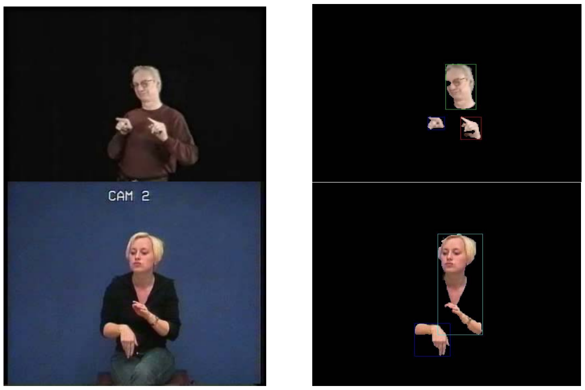

Figure 2. Left column: original frames, right column: after segmentation and tracking.

Table 1. Experimental results.

	Face	Right hand	Left hand
Error in X direction (pixel)	1.722	1.516	4.781
Error in Y direction (pixel)	2.796	2.268	6.236
Tracking error %	6.1%	6.5%	6.5%

5. Conclusion

In this paper, a complete unified system for segmentation and tracking of skin objects for gesture recognition systems has been proposed. The algorithm works on skin detection instead of using colour gloves. Occlusion detection is handled between any of the face and the two hands accurately, which is very important as most of the real-world SL video sequences include many occlusions (about 40% as demonstrated in our testing data). The tracking process uses the occlusion information to maintain a high-level understanding of the occlusion status of all the skin objects. More importantly, tracking and segmentation tasks have been approached as one unified problem where tracking helps to reduce the search space used in segmentation, and good segmentation helps to accurately enhances the tracking performance.

6. References

[1] D.J. Sturman, and D. Zeltzer, "A Survey of Glove-based Input", *IEEE Computer Graphics and Applications,* 14, pp. 30–39, 1994.

[2] C.W. Ong. Sylvie, and S. Ranganath, "Automatic Sign Language Analysis: A Survey and the Future beyond Lexical Meaning", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 27 (6), pp. 873–891, 2005.

[3] A. Shamaie and A. Sutherland, "Hand Tracking in Bimanual Movements", *Image and Vision Computing*, 23, pp. 1131–1149, 2005.

[4] M.-H. Yang, N. Ahuja, and M. Tabb, "Extraction of 2D Motion Trajectories and Its Application to Hand Gesture Recognition", *IEEE Trans. Pattern Analysis Machine Intelligence*, 24(8), pp. 1061-1074, 2002.

[5] C.-L. Huang and S.-H. Jeng, "A Model-Based Hand Gesture Recognition System", *Machine Vision and Application*, 12(5), pp. 243-258, 2001.

[6] J.-C. Terrillon, A. Piplr, Y. Niwa, and K. Yamamoto, "Robust Face Detection and Japanese Sign Language Hand Posture Recognition for Human-Computer Interaction in an Intelligent Room", In *Proc. Int'l Conf. Vision Interface*, pp. 369-376, 2002.

[7] K. Imagawa, S. Lu and S. Igi, "Color-Based Hands Tracking System for Sign Language Recognition", In *Proc. of IEEE FG'98*, pp.462-467, 1998.

[8] V. Vapnik, *The nature of statistical learning theory*. Springer, New York, 1995.

[9] J. Han, G. Awad, and A. Sutherland, "Automatic Skin Segmentation for Gesture Recognition combining Region and Support Vector Machine Active Learning", In *Proc. of IEEE FG'06*, pp.237-242, 2006

[10] J. Martin, V. Devin, and J.L. Crowley. "Active Hand Tracking", In *Proc. of IEEE FG'98*, pp. 573–578, 1998.