

Real-Time Hand Gesture Recognition for Uncontrolled Environments Using Adaptive SURF Tracking and Hidden Conditional Random Fields

Yi Yao and Chang-Tsun Li

Department of Computer Science, University of Warwick, Coventry CV4 7AL, UK

Abstract. Challenges from the uncontrolled environments are the main difficulties in making hand gesture recognition methods robust in real-world scenarios. In this paper, we propose a real-time and purely vision-based method for hand gesture recognition in uncontrolled environments. A novel tracking method is introduced to track multiple hand candidates from the first frame. The movement directions of all hand candidates are extracted as trajectory features. A modified HCRF model is used to classify gestures. The proposed method can survive challenges including: *gesturing hand out of the scene, pause during gestures, complex background, skin-coloured regions moving in background, performers wearing short sleeve and face overlapping with hand*. The method has been tested on Palm Graffiti Digits database and Warwick Hand Gesture database. Experimental results show that the proposed method can perform well in uncontrolled environments.

1 Introduction

Hand gesture recognition is an intuitive way for facilitating Human Computer Interaction (HCI). However, its robustness against uncontrolled environments is widely questioned. Many challenges exist in real-world scenarios which can largely affect the performance of appearance based methods, including presence of cluttered background, moving objects in background, gesturing hand out of the scene, pause during the gesture, and presence of other people or skin-coloured regions, etc. This is the reason why the majority of works in hand gesture recognition are only applicable in controlled environments.

There have been few attempts for recognising hand gestures in different uncontrolled environments. Bao et al. [1] proposed an approach using SURF [2] as features to describe hand gestures. The matched SURF point pairs between adjacent frames are used to produce the hand movement direction. This method only works under the assumption that the gesture performer occupies a large proportion of the scene. If there are any other moving objects at the same scale of the gesture performer in the background, the method will fail. Elmezain et al. [3] proposed a method which segments hands from the complex background using 3D depth map and colour information. The gesturing hand is tracked by using Mean-Shift and Kalman filter. Fingertip detection is used for locating the target hand. However, this method can only deal with the cluttered background and is unable to cope with other challenges mentioned

earlier. Alon et al. [4] proposed a framework for spatiotemporal gesture segmentation. Their method is tested in uncontrolled environments with other people moving in the background. This method tracks a certain number of candidate hand regions. The number of candidate regions can largely affect the performance of the method, which must be specified beforehand, making it unrealistic in real-world scenarios. Two other works ([5], [6]) also tested their methods on the database of [4]. But none have outperformed [4] on their database.

In this paper, we propose a method for hand gesture recognition in uncontrolled environments. A novel tracking method called Adaptive SURF Tracking is introduced to extract hand trajectories. A model based on Hidden Conditional Random Fields (HCRF) [7] is trained to classify hand trajectories into 10 digits gesture classes.

2 Adaptive SURF Tracking

One of the key differentiating features of our proposed method from other existing methods is that the exact location of the gesturing hand is not required. Similar to [4], our method also keeps tracks of multiple candidates of hand regions. We call this tracking method Adaptive SURF Tracking. In the first frame of the video sequence, skin colour cues are used to detect possible skin colour regions as the initial regions of interests (ROI). After the first frame, key SURF points are extracted from every ROI and matched against their counterparts in the next frame. The dominant movement orientation of every ROI is then extracted from each frame to form the candidate hand trajectory vector, which is used as the input of the HCRF model.

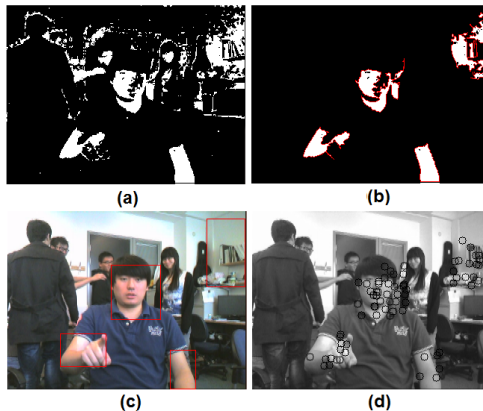


Fig. 1. Processing of the first frame, (a): skin colour binary image; (b) : result of denoising; (c) : initial ROIs; (d): SURF key points within initial ROIs

Figure 1 illustrates the mechanism for processing the first frame. The method detects faces in the frame, by using the Viola-Jones face detector [8]. Then thresholds of skin colour in the HSV colour space are estimated using pixels in the detected facial regions. Those thresholds are later used to produce a skin-colour binary image (Figure 1(a)) in the processing of every frame in the video. Hence the method can adapt to

different illumination conditions. If no faces are detected in the first frame, a Gaussian Mixture Model (GMM) in the RGB colour space, which is inferred from a large database of skin and non-skin pixels [9], will be used to calculate the skin-colour binary image, until at least one valid face region is detected in later frames. All closed contours are then detected in this binary image. A denoising process is performed on the skin-colour binary image by deleting all the interior contours and the contours of the areas smaller than a threshold T_{dsr} (Figure 1(b)), where

$$T_{dsr} = \bar{A}_f \times 0.25 \quad (1)$$

\bar{A}_f is the average area of all detected facial regions in the first frame. The minimum bounding rectangles from this binary image are taken as the initial ROIs of the first frame, as shown in Figure 1(c). Subsequently, SURF points are extracted from the first frame and those key points within the ROIs are kept (circles in Figure 1(d)).

Starting from the second frame, SURF features are extracted from the whole image of the current frame and matched against their counterparts in the previous frame, as shown in Figure 2(a). Once the matched pairs are calculated, a pruning process is performed on all matched pairs. Only those pairs with a displacement within a certain range between the matched key points in the current frame and the matching points in the previous frame are preserved. All the matched pairs which are located in stationary regions (e.g. in the face region) or regions that do not move beyond the lower bound of this displacement range are dropped. On the other hand, if a matched key point has displaced beyond the upper bound of the displacement range in the next frame, it most likely is a mismatch. This is a reasonable assumption because if an object moves too much within such a short period of time, it is unlikely to be the target hand. Various displacement ranges have been tested and we found that the range between 3 and 40 pixels is empirically feasible. An example of pruning is shown in Figure 2(b).

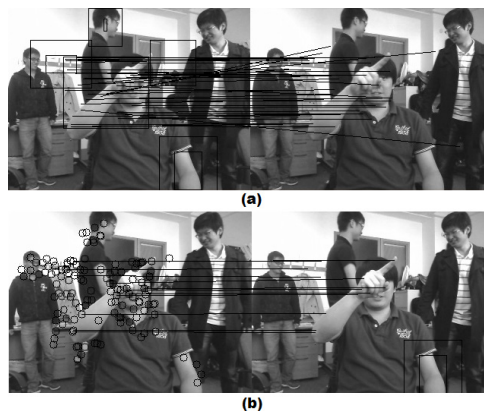


Fig. 2. Pruning process. (a): matched key point pairs from one of the ROIs, between previous frame (left) and current frame (right). (b): the remaining matched key point pairs after pruning.

After the pruning process, the ROIs in the current frame are drawn. For the SURF key points in one ROI in the previous frame, there are key SURF points in the current frame that matched to them. The corresponding ROI in the current frame is defined as the minimum bounding rectangle of these matched key points in the current frame. During every frame, according to the number of the remaining matched key points P after pruning and the area of the new ROIs A , the boundaries of the new ROIs may be extended by e pixels. The value of e is set as:

$$e = \begin{cases} 0, A \geq 3000 \\ 20, 500 \leq A < 3000 \\ 20, A < 500 \wedge P > 5 \\ 30, A < 500 \wedge 0 < P \leq 5 \\ 40, P = 0 \end{cases} \quad (2)$$

Instead of only keeping the matched key points in each of the new ROIs of the current frame, all key SURF points within these new ROIs are preserved for matching with those in the next frame. The reason for enlarging the ROIs is that they may not cover the entire area of the target hand candidates. Hence, in order to get as many tracking features as possible from the current ROIs of target hand candidates, the ROIs need to be enlarged to make sure that the new ROIs cover the corresponding hand candidates.

For every frame, the dominant movement direction of each ROI will be calculated as the hand trajectory feature of this hand candidate. Assume we have P matched SURF pairs between frames $t-1$ and t after pruning in a ROI, denoted by $M_t = \{ \langle S_{t-1}^1, S_t^1 \rangle, \langle S_{t-1}^2, S_t^2 \rangle, \dots, \langle S_{t-1}^P, S_t^P \rangle \}$, where $\langle S_{t-1}^i, S_t^i \rangle$ is the i^{th} pair. The dominant movement direction of the r^{th} ROI in frame t is defined as:

$$drt(t, r) = \arg \max_d \{q_d\}_{d=1}^D \quad (3)$$

where, $\{q_d\}_{d=1}^D$ is the histogram of the movement direction of all matched SURF key point pairs in this ROI, and d indicates the index of directions. q_d is the d^{th} bin of the histogram. Each bin has an angle interval with range α , and $D = 360^\circ / \alpha$. We have tested various values for α and found that 20° produces best results. Definition of q_d is:

$$q_d = C \sum_{p=1}^P k \left(\|S_t^p\|^2 \right) \delta(S_t^p, d) \quad (4)$$

where, $k(x)$ is an monotonic kernel function which assigns smaller weights to those key SURF points farther away from the centre of this ROI; $\delta(S_t^p, d)$ is the Kronecker delta function which has value 1 if the movement direction of $\langle S_{t-1}^p, S_t^p \rangle$ falls into the d^{th} bin; and the constant C is a normalisation coefficient defined as:

$$C = 1 / \sum_{p=1}^P k \left(\|S_t^p\|^2 \right) \quad (5)$$

Another desirable feature of our proposed method is that we only use hand movement direction as hand trajectory feature. Since speed and location of gestures are used as features in [4], to make their method less sensitive to the location and size of the gestures, face detection is used to estimate the location and scale of the gesture performers. Unlike [4], the location and speed of hand candidates are not used to describe hand gestures, hence our method does not need to estimate the location and scale of the gestures, the modified HCRF model is also not sensitive to the length of the gestures, which makes the proposed method invariant against the location, speed and size of the hand gestures.

3 Gesture Classification

After the tracking stage, once the movement direction vectors, namely the input sequences for HCRF model, of every hand candidates in the videos are extracted, they are put into a multi-class chain HCRF model as feature vectors, as shown in Figure 3. The videos are naturally segmented as one single frame is a single node in HCRF model. HCRF has been proven to be one of the strongest discriminative models with hidden states [7]. In this paper, since the task is recognising a set of hand-signed digits $Y = [y_0, y_1, \dots, y_9]$ (as shown in Figure 4), we define the hidden states to be the strokes of gestures. There are in total 13 states in the HCRF model for our own database, and 15 states in the Palm Graffiti Digits database [4]. Figure 3 shows 4 of the 13 states in our Warwick Hand Gesture Database, which form the gesture of digit 4. The optimisation scheme used in our HCRF model is Limited Memory Broyden–Fletcher–Goldfarb–Shanno method [10]. In our experiments, the weight vector θ is initialised with the mean value, and the regularisation factors are set to zero.

As one sequence of the movement direction represents the trajectory direction vector of one hand candidate, a video clip X with R ROIs can have multiple sequences: $X = [x_1, x_2, \dots, x_R]$. Hence we modified the original HCRF model to suit our special case of multiple sequences for one video. In the original HCRF model, the probability of gesture y , given the video clip X , hidden states h and weight vector θ , is calculated by,

$$P(y | X, \theta) = \sum_h P(y, h | X, \theta) = \frac{\sum_h \exp\{\Psi(y, h, X; \theta)\}}{\sum_{y', h} \exp\{\Psi(y', h, X; \theta)\}} \quad (6)$$

where $\Psi(y, h, X; \theta)$ is the potential function. Follow [7], we define the partition function :

$$Z(y | X, \theta) = \sum_h \exp\{\Psi(y, h, X; \theta)\} \quad (7)$$

For multiple sequences video $X = [x_1, x_2, \dots, x_R]$, the new partition function is defined:

$$Z'(y | X, \theta) = \arg \max_{x_r} \sum_h \exp \{ \Psi(y, h, x_r; \theta) \} \tag{8}$$

Hence the probability of gesture y , given the video clip X is:

$$P(y | X, \theta) = \frac{Z'(y | X, \theta)}{\sum_{y'} Z'(y' | X, \theta)} \tag{9}$$

and we take the final gesture to be $\arg \max_{y \in Y} P(y | X, \theta)$. Namely, the final gesture

label assigned to this video clip is the one with the highest partition value among all the partitions of all sequences during this gesture.

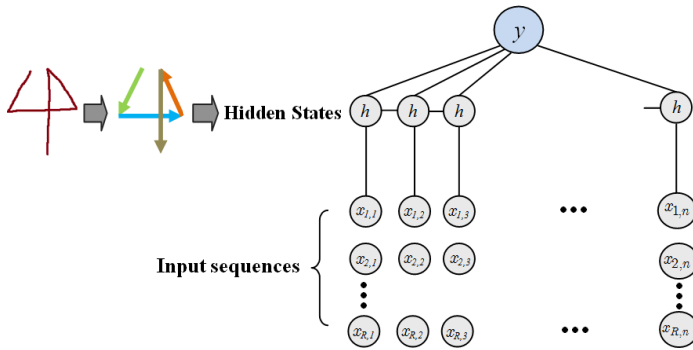


Fig. 3. HCRF model, the hidden states are defined as strokes of gestures, y is the gesture label, node $x_{R,n}$ means the movement direction of the R^{th} hand candidate in the n^{th} frame of the video

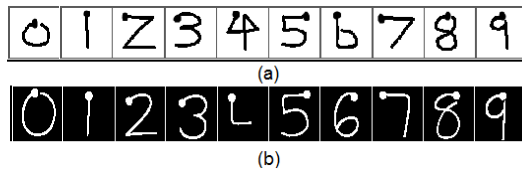


Fig. 4. The hand gesture sets, (a) is defined in our own database (Warwick Hand Gesture Database), (b) is from Palm Graffiti Digits Database[4]

4 Experiments


We conducted two experiments on two databases. First experiment is on the Palm Graffiti Digits database used in [4]. This database contains 30 video samples for training, 3 samples from each of 10 performers that wear gloves. Each sample captures the performer signing digits 0-9 each for once. There are two test sets, the "hard" and "easy" sets. There are 30 videos in the easy set, 3 from each of 10 performers, and 14 videos in the hard set, 2 from each of 7 performers. The contents of both test sets are the same as the training set, except that performers do not wear gloves, and there are 1

Table 1. Results of experiment on the easy set of the Palm Graffiti Digits database [4]

Gesture class	Easy Set				
	Training samples	Testing samples	Recognition Results		
			True	Detected	Accuracy (%)
0	30	30	30	34	100.00
1	30	30	30	31	100.00
2	30	30	28	28	93.33
3	30	30	28	28	93.33
4	30	30	30	30	100.00
5	30	30	30	32	100.00
6	30	30	25	26	83.33
7	30	30	29	30	96.67
8	30	30	30	30	100.00
9	30	30	27	31	90.00
Overall	300	300	287	300	95.67

Table 2. Results and sample (Gesture 4) of experiment on the hard set of the Palm Graffiti Digits database [4]

Gesture class	Hard Set				
	Training samples	Testing samples	Recognition Results		
			True	Detected	Accuracy (%)
0	30	14	11	15	78.57
1	30	14	13	13	92.86
2	30	14	13	15	92.86
3	30	14	13	14	92.86
4	30	14	13	14	92.86
5	30	14	14	16	100.00
6	30	14	6	6	42.86
7	30	14	12	13	85.71
8	30	14	13	16	92.86
9	30	14	13	18	92.86
Overall	300	140	121	140	86.43



to 3 people moving back and forth in the background (Table 2) in hard set. The specifications of the videos are: 30Hz, and resolution of 240×320 pixels. The results of the proposed method are shown in Table 1 and 2. Follow [4], the proposed method also does not aware the starting and ending frames of each gesture. A simple gesture spotting rule is applied. From last ending point up to the current frame, if the partition from all gesture classes are lower than a threshold, this part of the video will be treated as nonsign gesture. When at least one gesture class produces partition higher than the threshold, the proposed method will treat this frame as the starting frame of the gesture, until partitions from all gesture classes are lower than the threshold again. Compared with [4],[5],[6] and [1], the proposed method produced better accuracy on both easy and hard set, as shown in Table 3 and Figure 6.

The reasons the proposed method outperformed [4] are that: Firstly, in [4], the number of hand candidates must be specified beforehand, which is often unrealistic in real-world scenarios. The proposed method does not require the prior knowledge on the contents of the background. When processing the first frame, the eligible skin-coloured regions are taken as initial ROIs. Hence our method can adaptively detect

Table 3. All reported experimental results ([4],[5] and [6]) on Palm Graffiti Digits database, and results produced by this paper (the proposed method and method of [1])

10 Palm Graffiti Digits database [4]		
	Easy set	Hard set
Correa et al. RoboCup 2009 [5]	75.00%	N/A
Malgireddy et al. CIA 2011 [6]	93.33%	N/A
Alon et al. PAMI 2009 [4]	94.60%	85.00%
Bao et al. ICEICE 2011 [1]	52.00%	28.57%
The proposed method	95.67%	86.43%

number of hand candidates. Secondly, the people or other moving objects entering the scene after the first frame have no impact on the proposed method. Those objects will not be matched to the SURF features of the objects (including gesturing hand) that exist since the first frame. In [4], all the hand candidates in all frames have to be tracked, which makes the method inapplicable to the real-world scenarios.

We collected a more challenging database called Warwick Hand Gesture Database (see Figure 2 for example) to demonstrate the performance of the proposed method under new challenges. 10 gesture classes as in Figure 4(a) are defined for our database. This database consists of two testing sets, namely "easy" and "hard" sets. There are 300 video samples for training, 3 samples were captured from each of 10 performers for each gesture. There are 1000 video samples in total for testing. For each gesture, 10 samples were collected from each of 10 performers. The specifications of videos are the same as Palm Graffiti Digits database. Similar to the Palm Graffiti Digits database, the hard set of our database captures performers wearing short-sleeve tops with cluttered backgrounds. The differences are: No gloves in training set. Instead of 1-3 people, we had 2-4 people moving in the background, and there are new challenges in the clips, including: gesturing hand out of scene and pause during gesture. Since the work of [1] is the one most similar to the proposed method, we compared the performance between these two methods (Table 4 and Figure 6).

Table 4. The performances of method of [1] and the proposed method on Warwick Hand Gesture Database

Warwick hand gesture database		
	Easy set	Hard set
Bao et al. ICEICE 2011 [1]	71.00%	18.20%
The proposed method	93.80%	85.40%

As shown in the graph of movement direction vectors (Figure 5), the intra-class variance in our database is larger than the database of [4]. Our method still produced similar accuracy on both Warwick Hand Gesture Database and Palm Graffiti Digits Database. The reason our method can handle the new challenge of gesturing hand out of scene is that the ROI covers the arm section when the hand is out of the scene. The arm section is tracked until the frame in which the hand is back in the scene. Since, when the ROI is being redefined and enlarged in this frame, the hand section will be covered again. Therefore, the SURF features will be extracted in the new ROI

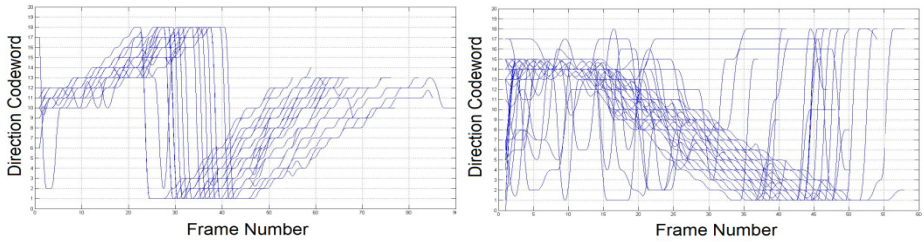


Fig. 5. Movement direction vectors for the gesture of digit 6 of the training set of: Palm Graffiti Digits database (left) and Warwick Hand Gesture database (right). The horizontal axis is the frame number while the vertical axis is the direction codeword (1-18).

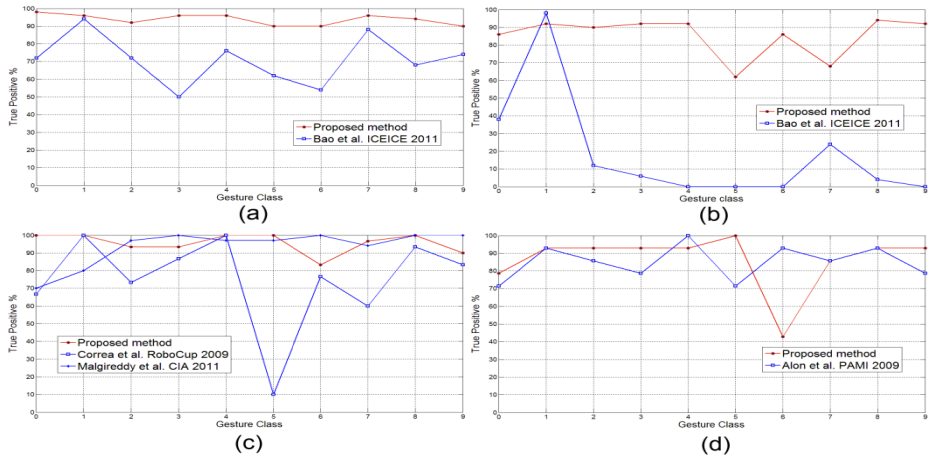


Fig. 6. The comparison of performances on (a): easy set of the Warwick hand gesture database; (b) hard set of the Warwick hand gesture database; (c): easy set of the Palm Graffiti Digits database; (d): hard set of the Palm Graffiti Digits database. The horizontal axis is the gesture label while the vertical axis is the recognition rate.

which covers the returning hand section. As for the pause during gesture challenge, the proposed method preserves the ROI when the number of moving matched SURF pairs in this ROI is 0, which means either the target is stationary or the method has lost track on this ROI. The ROI are enlarged in every frame until the number of matched SURF pairs is not 0. Hence the method can regain tracks on targets in most of the situations within several frames. As for speed, the proposed method performs on average in both experiments at: 53.00 ms/frame for easy sets, 53.75ms/frame for hard sets. That is 18.9 frames/sec and 18.6 frames/sec, respectively. Hence our method is able to perform comfortably in real time.

5 Conclusions

In this paper, we propose a real-time and purely vision-based method for hand gesture recognition in uncontrolled environments. The method can recognise hand gestures

against the complex background with 2 to 4 people moving in it. The method can handle challenges such as complex background, skin-coloured regions moving in background, performers wearing short-sleeve and face overlapping with hand. The method was tested on Palm Graffiti Digits Database [4], and achieved 95.67% on easy set, 86.43% on hard set. We also tested the proposed method on our own database with additional challenges of gesturing hand out of scene and pause during gesture. The method achieved 93.80% on easy set and 85.40% on hard set.

Acknowledgment. This work is included in the pending patent: UK patent application GB1305812.8, 28 March 2013, University of Warwick.

References

1. Bao, J., Song, A., Guo, Y., Tang, H.: Dynamic Hand Gesture Recognition Based on SURF Tracking. In: International Conference on Electric Information and Control Engineering, ICEICE (2011)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded-Up Robust Features. *Computer Vision and Image Understanding (CVIU)* 110(3), 346–359 (2008)
3. Elmezain, M., Al-Hamadi, A., Michaelis, B.: A Robust Method for Hand Gesture Segmentation and Recognition Using Forward Spotting Scheme in Conditional Random Fields. In: International Conference on Pattern Recognition, ICPR, pp. 3850–3853 (2010)
4. Alon, J., Athitsos, V., Yuan, Q., Sclaroff, S.: A Unified Framework for Gesture Recognition and Spatiotemporal Gesture Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 1685–1699 (September 2009)
5. Correa, M., Ruiz-del-Solar, J., Verschae, R., Lee-Ferng, J., Castillo, N.: Real-Time Hand Gesture Recognition for Human Robot Interaction. In: Baltes, J., Lagoudakis, M.G., Naruse, T., Ghidary, S.S. (eds.) *RoboCup 2009*. LNCS, vol. 5949, pp. 46–57. Springer, Heidelberg (2010)
6. Malgireddy, M.R., Nwogu, I., Ghosh, S., Govindaraju, V.: A Shared Parameter Model for Gesture and Sub-gesture Analysis. In: Aggarwal, J.K., Barneva, R.P., Brimkov, V.E., Korutchev, K.N., Korutcheva, E.R. (eds.) *IWCIA 2011*. LNCS, vol. 6636, pp. 483–493. Springer, Heidelberg (2011)
7. Quattoni, A., Wang, S., Morency, L.P., Collins, M., Darrell, T.: Hidden-state Conditional Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 1848–1852 (October 2007)
8. Viola, P., Jones, M.J.: Robust Real-Time Face Detection. *International Journal of Computer Vision* 57, 137–154 (2004)
9. Jones, M.J., Rehg, J.M.: Statistical Color Models with Application to Skin Detection. *International Journal of Computer Vision* 46(1), 81–96 (2002)
10. Liu, D.C., Nocedal, J.: On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical Programming* 45(1-3), 503–528 (1989)