A Framework for Real-Time Hand Gesture Recognition in Uncontrolled Environments with Partition Matrix Model based on Hidden Conditional Random Fields

Yi Yao and Chang-Tsun Li
Department of Computer Science,
University of Warwick,
Coventry CV4 7AL
Y.Yao@warwick.ac.uk, c-t.li@warwick.ac.uk

***Abstract*—The main obstructions of making hand gesture recognition methods robust in real-world applications are the challenges from the uncontrolled environments, 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. Therefore, a framework for real-time hand gesture recognition in uncontrolled environments is proposed in this paper. A novel tracking scheme is proposed to track multiple hand candidates in unconstrained background, and a weighting model for gesture classification based on Hidden Conditional Random Fields which takes trajectories of multiple hand candidates under different frame rates into consideration is also introduced. The framework achieved invariance under change of scale, speed and location of the hand gestures. The Experimental results of the proposed framework on Palm Graffiti Digits database and Warwick Hand Gesture database show that it can perform well in uncontrolled environments.**

Keywords-hand gesture recognition; uncontrolled environments; Hidden Conditional Random Fields; SURF tracking.

I. INTRODUCTION

Within the context of hand gesture recognition, the challenges from the uncontrolled environments, including the presence of cluttered backgrounds, moving objects in the background, gesturing hand out of the scene during gesture, pause during the gesture, and presence of other people or skin-coloured regions, are the main difficulties that keep this intuitive way of Human Computer Interaction (HCI) from widely utilised in real-world scenarios. Moreover, the position, scale and length variance of the hand gestures can be large even for the same gesture from the same gesture performer under the same environment. In this paper, a framework which is dedicated for tackling those aforementioned challenges is proposed.

One of the key differentiating features of the proposed framework is that our method does not require any constraint on the environment, namely no assumptions are made about the content of the background or the scale, speed or location of the gesture performer. Many existing methods (e.g., [1][2][3]) only work under certain assumptions. Bao et al. [1] proposed an approach using Speeded Up Robust Features (SURF) [4] to track hands. The method can only handle moving background distractions with areas smaller than the gesture performing arm. Elmezain et al. [2] proposed a method which tracks hands from the complex background using 3D camera and fingertip detection. The method requires certain hand posture for fingertip detection to work. Alon et al. [3] proposed a framework for spatiotemporal gesture segmentation. The amount of hand candidates must be specified beforehand, which is a strong assumption on the content of the background. There are two other works ([5][6]) also reported experimental results on the database of [3]. Our experiments show that the proposed framework outperformed aforementioned methods ([1][3][5][6]).

II. ADAPTIVE SURF TRACKING

Another key differentiating feature of the proposed framework is that it does not need hand segmentation process, namely the exact position of the target hand is not required in the tracking scheme. The framework can locate all eligible hand candidates, namely Regions of Interests (ROIs) from the first frame and keeps track on all of them. Hence the framework can adapt to arbitrary content in the background (e.g. other people moving closely with the performer in the background). A novel tracking scheme is proposed in this paper, we call it *Adaptive SURF Tracking*.

To locate all hand candidates, skin-colour cues are used. The processing of the first frame is illustrated in Fig.1. For adapting to different illumination conditions, the proposed tracking scheme detects faces in the first frame using the Viola-Jones detector [8]. Then the thresholds in HSV colour space for producing the skin-colour binary image (Fig. 1a), are estimated using the pixels in the detected facial regions. If no faces are detected in the first frame, a Gaussian Mixture Model (GMM) in the RGB colour space which trained out of a large skin-colour database [9], will be used to produce the skin-colour binary image, until eligible facial regions are detected in later

978-1-4799-0652-9/13 $31.00 © 2013 IEEE
DOI 10.1109/SMC.2013.209

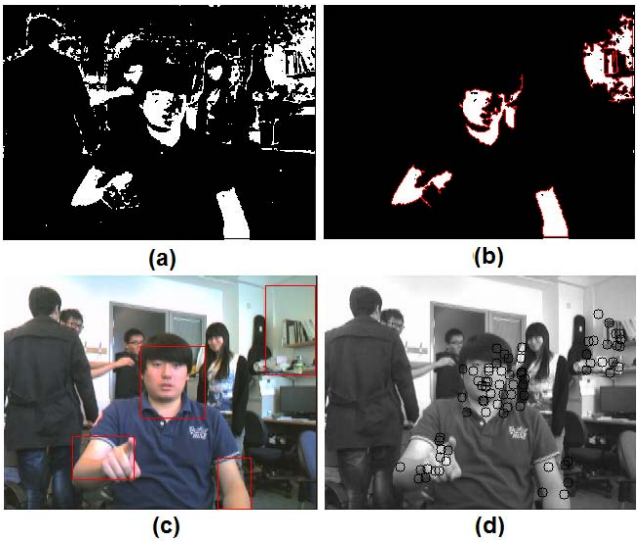

Figure 1: Processing of the first frame, (a): skin color binary image; (b) : result of denoising; (c) : initial ROIs; (d): SURF key points within initial ROIs.

frames. Then a denoising process is performed on all the closed contours in the skin-colour binary image. All the interior contours and contours with areas smaller than a threshold T_{dsr} are deleted (Fig.1b).

$$T_{dsr} = \bar{A}_f \times 0.25 \tag{1}$$

where $\bar{A}_f$ is the average area of all detected facial regions in the first frame. Eligible hand candidates, namely ROIs are defined as the minimum bounding rectangles of the remaining contours (Fig.1c). Then the SURF key points are extracted from the ROIs in the first frame.

From the second frame, SURF key points are extracted from the whole image of the current frame *t*, and matched with SURF key points from ROIs in the last frame *t-1*. The r^{th} ROI which corresponds with the target hand region is taken as example in the rest of this section (Fig. 2a). Assuming the set of SURF key points $S_{t-1} = \left[S_{t-1}^1, S_{t-1}^2, ..., S_{t-1}^{P_{t-1}} \right]$ of the r^{th} ROI in frame *t-1* contains P_{t-1} key points, and every SURF key point $S_{t-1}^P = \left(CX_{t-1}^P, CY_{t-1}^P, DS_{t-1}^P \right)$ contains coordinates $\left(CX_{t-1}^P, CY_{t-1}^P \right)$ and the SURF descriptor DS_{t-1}^P. We define a matched SURF key point S_t^P given a key point S_{t-1}^P to be:

$$\left\{ S_t^P \mid \left\| DS_t^P - DS_{t-1}^P \right\| / \left\| DS_t^{P'} - DS_{t-1}^P \right\| \geq T_{match} \right\} \tag{2}$$

where ,

$$\begin{aligned} S_t^P &= \arg\min_{S_t^P} \left\| DS_t^P - DS_{t-1}^P \right\|, S_t^P \in S_t, \\ S_t^{P'} &= \arg\min_{S_t^{P'}} \left\| DS_t^{P'} - DS_{t-1}^P \right\|, S_t^{P'} \in S_t - S_t^P \end{aligned} \tag{3}$$

and T_{match} is set to 0.9 empirically. Once the matched pairs $M_t = \left\{ \left\langle S_{t-1}^1, S_t^1 \right\rangle, \left\langle S_{t-1}^2, S_t^2 \right\rangle, ..., \left\langle S_{t-1}^{P_{t-1}}, S_t^{P_{t-1}} \right\rangle \right\}$ are found, a pruning process is performed on all matched pairs in the ROI. Only

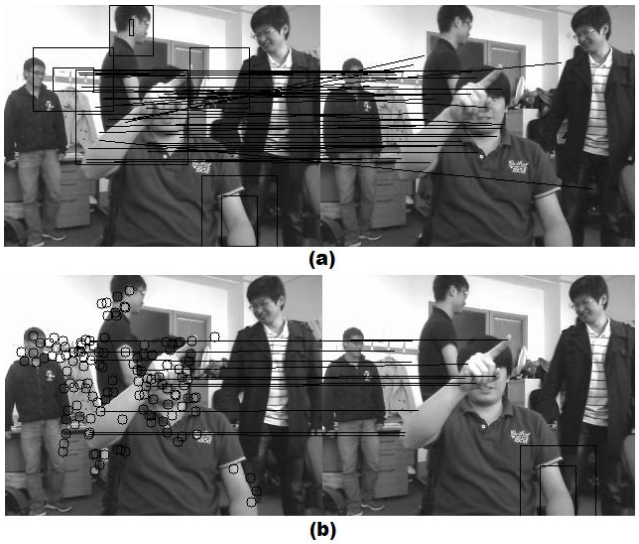

Figure 2: Pruning process. (a): matched key point pairs from one of the ROIs, between previous frame (left) and current frame (right). (b): the remaining matched key point pairs after pruning.

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 with a displacement smaller than the lower bound $T_{\min,t}$ of the ROI's displacement range are dropped. On the other hand, if a matched key point has displaced more than the upper bound $T_{\max,t}$ of the ROI's displacement range in the next frame, it is most likely 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. The lower and upper displacement bounds of the ROI in frame *t* are calculated based on the average displacement of all matched key point pairs in the ROI between frame *t-2* and *t-1,* namely $M'_{t-1} = \left\{ \left\langle S_{t-2}^1, S_{t-1}^1 \right\rangle, \left\langle S_{t-2}^2, S_{t-1}^2 \right\rangle, ..., \left\langle S_{t-2}^{P'_{t-2}}, S_{t-1}^{P'_{t-2}} \right\rangle \right\}$, where the prime symbols indicate *after pruning*. The definition of the lower and upper bounds are:

$$T_{\min,t} = \frac{\sum_{p=1}^{P'_{t-2}} \left\| \left(CX_{t-1}^p - CX_{t-2}^p, CY_{t-1}^p - CY_{t-2}^p \right) \right\|}{P'_{t-2}} \times F_{mov,\min} \tag{4}$$

$$T_{\max,t} = \frac{\sum_{p=1}^{P'_{t-2}} \left\| \left(CX_{t-1}^p - CX_{t-2}^p, CY_{t-1}^p - CY_{t-2}^p \right) \right\|}{P'_{t-2}} \times F_{mov,\max} \tag{5}$$

where $F_{mov,\min} = 0.25$ and $F_{mov,\max} = 3$ are factors of minimum and maximum displacement, the values are chosen through experiments. For the first frame, various default displacement ranges have been tested and we found that the default values of lower and upper bounds $T_{\min,0} = 3$ and $T_{\max,0} = 40$ pixels were empirically feasible. Also if $T_{\min,t}$ is less than the default value, it would be set to the default value. Hence for the stationary regions (e.g. in the face region), where no large movements would be found, the majority of the matched key point pairs can be dropped, so the challenge of face/hand overlapping is naturally resoloved. Moreover, since the displacement range of

ROI is recalculated in every frame adaptively, based on the movement distance of the main object in the ROI, the proposed tracking scheme can adapt to speed changes of the target. If the target, namely the main object in the ROI is accelerating, the displacement range will move up according to the actual acceleration, which can be represented by the average displacement of all matched key point pairs. An example of pruning is shown in Fig. 2b. After the pruning process, the new ROI in the current frame are drawn. The corresponding ROI in the current frame is defined as the minimum bounding rectangle of these matched key points $S'_t = \left[S_t^1, S_t^2, ..., S_t^{P'_{t-1}} \right]$ after pruning. Instead of only keeping the matched key points in the new ROI of the current frame, all key SURF points within the new ROI $S_t = \left[S_t^1, S_t^2, ..., S_t^{P_t} \right]$ are preserved for matching with SURF key points in the next frame. Hence, in order to get as many tracking features as possible from the current ROI, the ROI need to be enlarged to make sure that the new ROI covers the corresponding hand candidate.

Assuming the number of the remaining matched key points after pruning is P'_{t-1} and the area of the new r^{th} ROI is A_t, the boundaries of the new r^{th} ROI are then extended by e_t pixels. The value of e_t is set as:

$$e_t = \begin{cases} 0, A_{r,t} > A_{\max ROI} \\ \exp\left(-\dfrac{A_{r,t}}{A_{HA}}\right) \cdot E_r, A_{HA} < A_{r,t} < A_{\max ROI} \\ \left[\exp\left(-\dfrac{P'_{t-1}}{P_{\min}}\right) + E_{boost}\right] \cdot E_r, A_{r,t} < A_{HA} \wedge P'_{t-1} \le 3 \\ \exp\left(-\dfrac{P'_{t-1}}{P_{\min}}\right) \cdot E_r, A_{r,t} < A_{HA} \wedge P'_{t-1} > 3 \end{cases} \quad (6)$$

where $A_{\max ROI} = \left(h_s \cdot w_s\right) / 20$ is the estimated maximum area of ROIs, h_s and w_s are the height and width of the video respectively. $A_{HA} = \left(h_s \cdot w_s\right) / 60$ is the estimated area of the hand region. $P_{\min}$ is the estimated minimum acceptable value of P'_{t-1} and the value of 10 is used in our experiments. E_{boost} is a factor to ensure that the lower the value of P'_{t-1} is, the higher the enlargement is given to this ROI, the value of 0.3 is used in our experiments; and E_r is the enlargement scale for the r^{th} ROI, where

$$E_r = \begin{cases} \left[\left(h_{r,0} + w_{r,0}\right) / 2\right] \cdot F_s, h_{r,0} \cdot w_{r,0} < \overline{A}_f \cdot 2.5 \\ \sqrt{\overline{A}_f} \cdot F_s, otherwise \end{cases} \quad (7)$$

$h_{r,0}$ and $w_{r,0}$ are the initial height and width of this r^{th} ROI in the first frame; $F_s = h_s \cdot w_s / 30$ is the enlargement factor corresponding to the frame size. Hence e_t depends on the original size of this ROI in the first frame. The enlargement of the ROI would increase the coverage of the ROI on the target hand candidate, without enlarging too much to cover other objects.

The dominant movement directions of all ROIs in every frame are extracted as the trajectory feature of the ROI. Since there are P'_{t-1} matched SURF pairs between frames *t* and *t-1* after pruning in the r^{th} ROI. The corresponding dominant movement direction of this ROI in frame *t* is defined as:

$$\mathrm{drt}(t,r) = \arg\max_d \{q_d\}_{d=1}^D \quad (8)$$

$\{q_d\}_{d=1}^D$ is the histogram of the movement direction, and *d* indicates the index of directions. q_d is the d^{th} bin of the histogram. The width of each bin is α, and D = 360°/ α. Various values for α have been tested, 20° is employed which can produce the best results. q_d is defined as

$$q_d = C \sum_{p=1}^{P'_{t-1}} k\left(\left\|S_t^p\right\|^2\right) \delta\left(S_t^p, d\right) \quad (9)$$

where *k(x)* is an monotonic kernel function which makes the key points that are located far away from the centre of the ROI having smaller weight. $\delta\left(S_t^p, d\right)$ is a simple Kronecker delta function used to see whether the direction of $\left\langle S_{t-1}^p, S_t^p \right\rangle$ falls in the d^{th} bin. The constant *C* is a normalisation coefficient defined as

$$C = 1 \Big/ \sum_{p=1}^{P'_{t-1}} k\left(\left\|S_t^p\right\|^2\right) \quad (10)$$

III. GESTURE CLASSIFICATION

Since Adaptive SURF Tracking uses texture feature to track hand candidates, using the SURF features from different neighbouring frames may produce different tracking results. In order to take as many tracking results as possible into consideration, we introduced a weighting algorithm called *Partition Matrix* to classify gestures with combined tracking results from different ROIs and frame selection patterns. The frame selection pattern means picking frames at different frame rates. Assuming that $V = \left\{f_i \mid i = 0,1,2,\ldots,N-1\right\}$ is a video with N frames, f_n is the n^{th} frame, video with pattern $F_p = \left\{f_i \mid i = 0, p, 2p, 3p...\right\}$ is a subset of V. In our experiments, patterns F_1 to F_4 are used to collect tracking information.

After the tracking stage, the movement direction vectors, namely the input vectors for HCRF model $X = \left\{x_{u,r} \mid u = 0,1,\ldots U, r = 0,1,\ldots R\right\}$ are fed into a multi-class chain HCRF model (see Fig.3). The total amount of input vectors equals to the number of frame selection patterns U times the number of ROIs R, In our experiments, $U = 4$ and one single frame is treated as a single node in the HCRF model. In this paper, since the task is recognising two sets of hand-signed digits (Fig.4), we define the hidden states *h* as the strokes of gestures. There are in total 13 states in the HCRF model for our own database (Fig.3 shows 4 of the 13 states, which form the gesture of digit 4), and 15 states in the Palm Graffiti Digits database [3]. The optimisation scheme

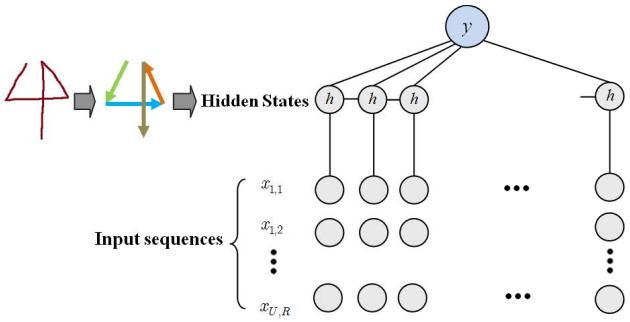

Figure 3: HCRF model, the hidden states are defined as strokes of gestures, input sequence x is the movement direction vector of one hand candidate under one frame selection pattern. $x_{u,r}$ means vector with u^{th} frame selection pattern and r^{th} ROI.

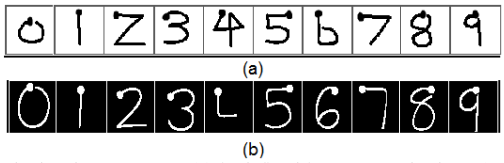

Figure 4: The hand gesture sets, (a) is defined in our own database (Warwick Hand Gesture Database), (b) is Palm Graffiti Digits [4].

used in our 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.

In the original HCRF model, the probability of gesture y given $x_{u,r}$, hidden states h and θ is

$$P(y \mid X,\theta)=\sum_h P(y,h \mid X,\theta)=\frac{\sum_h \exp\{\Psi(y,h,X;\theta)\}}{\sum_{y',h}\exp\{\Psi(y,h,X;\theta)\}} \tag{11}$$

where θ is a set of weights for every feature functions in the potential function. The partition function is defined as:

$$Z(y \mid X,\theta)=\sum_h \exp\{\Psi(y,h,X;\theta)\} \tag{12}$$

In the proposed framework, when a new video clip comes in for classification, all vector $x_{u,r}$ in X will be evaluated against each gesture class, namely the partition value will be calculated for each vector. Then a normalisation process is applied on all partition values. The reason for the normalisation is that, vectors with different frame rates have various length. The amount of feature functions in potential function depends on the amount of frames in the vector. The definition of potential function is:

$$\begin{aligned}\Psi(y,h,x_{u,r};\theta)&=\sum_j \phi(x_{u,r}(j))\cdot\theta(h_j)+\sum_j \theta(y,h_j)\\&+\sum_{(j,k)\in E}\theta(y,h_j,h_k)\end{aligned} \tag{13}$$

where j is the frame number of the vector $x_{u,r}$, E is the set of adjacent states, the three components represent three types of feature function (for details please refer to [7]). Hence, the vectors with more frames will have larger partition values. In order to compare partition values of vectors with different frame rates, the normalised partition value is defined as:

$$PT(y \mid x_{u,r},\theta)=\sum_h \exp\{\Psi(y,h,x_{u,r};\theta)\}/n_{u,r} \tag{14}$$

$n_{u,r}$ is the frame number of $x_{u,r}$. Since the only trajectory feature in the proposed framework is the movement direction, the size and location invariance are naturally achieved. Also the Partition Matrix model does not depend on the length of the vector, which makes the proposed framework robust against the changing of the gesture speed.

The idea of Partition Matrix is to take trajectories of multiple hand candidates under different frame rates into consideration. Therefore, the entire content of the uncontrolled environment which includes the gesture performer will be taken under evaluation. Even for the same ROI, with different frame rates, the Adaptive SURF Tracking scheme can producing largely different tracking results, due to all the randomly moving objects and people in the background. Hence, we define the Partition Matrix and the new partition function for observation X,

$$\begin{pmatrix}(PT_{0,0},L_{0,0}) & \cdots & (PT_{0,R},L_{0,R})\\ \vdots & \ddots & \vdots\\ (PT_{U,0},L_{U,0}) & \cdots & (PT_{U,R},L_{U,R})\end{pmatrix} \tag{15}$$

$$Z'(y \mid X,\theta)=\sum_{x_{u,r}\in X}\{PT(y \mid x_{u,r},\theta)\cdot w_{u,r}\} \tag{16}$$

where $(PT_{u,r},L_{u,r})$ is the partition-label pair of $x_{u,r}$, $w_{u,r}$ is the weight for the partition value of $x_{u,r}$, which is defined as:

$$w_{u,r}=1+W_{FR}(x_{u,r})+W_{ROI}(x_{u,r}) \tag{17}$$

$W_{FR}(x_{u,r})$ is the Frame Rate Weight function, which gives a larger weight to the vector with maximum $PT(y \mid x_{u,r},\theta)$ value among all ROIs with the same frame rate, namely a row in the Partition Matrix. The definition of Frame Rate Weight is:

$$W_{FR}(x_{u,r})=\begin{cases}1/U, x_{u,r}=\underset{x_{u,r}\in\{x_{u',r}\mid u'=u\}}{\arg\max} PT(y \mid x_{u,r},\theta)\\ 0, otherwise\end{cases} \tag{18}$$

$W_{ROI}(x_{u,r})$ is the ROI Weight function, which represents the confidence of the r^{th} ROI being the target hand. Definition is:

$$W_{ROI}(x_{u,r})=\left|\{x_{u',r'} \mid W_{FR}(x_{u',r'})\neq 0, r'=r\}\right|/U \tag{19}$$

The ROI Weight depends on how many row maximum value are there among the vectors from the same ROI, namely a column of the Partition Matrix. At last, the final gesture label is $\arg\max_y P(y \mid X,\theta)$, the gesture with the largest weighted sum of the normalised partition values over all vectors of X.

Gesture class	Hard Set					Easy Set				
	Training samples	Testing samples	Recognition Results			Training samples	Testing samples	Recognition Results		
			True	Detected	Accuracy(%)			True	Detected	Accuracy (%)
0	30	14	11	15	78.57	30	30	30	35	100.00
1	30	14	13	13	92.86	30	30	30	31	100.00
2	30	14	13	15	92.86	30	30	28	28	93.33
3	30	14	13	14	92.86	30	30	28	28	93.33
4	30	14	13	14	92.86	30	30	30	30	100.00
5	30	14	14	16	100.00	30	30	30	32	100.00
6	30	14	6	6	42.86	30	30	24	25	80.00
7	30	14	12	13	85.71	30	30	29	30	96.67
8	30	14	13	16	92.86	30	30	30	30	100.00
9	30	14	13	18	92.86	30	30	27	31	90.00
Overall	300	140	121	140	**86.43**	300	300	286	300	**95.33**

Figure 5. Results and sample (Gesture 4) of experiment on Palm Graffiti Digits database [3]

IV. EXPERIMENTS AND DISCUSSION

Two experiments are conducted on two databases for testing the proposed framework. The first one is on the Palm Graffiti Digits database used in [3]. Fig.5 shows the results of the proposed framework on both easy and hard test set. Video clips in the easy set do not have any moving objects in the background, while the videos in hard set have 1-3 people moving in the background. Since the videos are not segmented in this database, a single gesture spotting rule is used. For video from frame i to $i+t$, if the partition values from all gesture classes are lower than a threshold, this part of the video will be treated as garbage 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. The comparisons of performances are shown in Table 1 and Fig.7.

TABLE I. EXPERIMENTS ON PALM GRAFFITI DIGITS DATABASE

10 Palm Graffiti Digits database [3]		
	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 [3]	94.60%	85.00%
Bao et al. ICEICE 2011 [1]	52.00%	28.57%
The proposed method	**95.33%**	**86.43%**

All reported experimental results ([3],[5] and [6]) on 10 Palm Graffiti Digits database, and results produced by this paper (the proposed method and [1]).

In [3], the amount of hand candidates must be specified, while the proposed framework does not make any assumptions or require any prior knowledge on the content of the background. Also, extra computation on estimating the location and scale of the gestures is required in [3], while the proposed framework achieved scale, speed and location invariance without any extra computation.

Moreover, the method of [3] does not allow the model states to be skipped, and a strong assumption is made: the number of states is significantly smaller than the number of frames. That means the transition probabilities of the states are not used for classification. The Partition Matrix model in the proposed framework uses transition probabilities of the hidden states as one of the three feature functions in the potential function, that is another reason why our method outperformed [3]. Since the distractions in the background are moving randomly, not only in fixed position and viewpoint, out of plane rotation, changing of speed and overlapping of objects are also largely involved. Hence the texture of the background is changing rapidly. Although there are no constrains on how subjects should perform the gestures, the gesture performers tend to remain in relatively stationary position. That makes the changing of texture on the gesture performer relatively in small scale. The proposed Partition Matrix model uses this characteristic of Hand Gesture Recognition, and repeatedly applies the proposed Adaptive SURF Tracking scheme on the testing sample under different frame rates. Hence for videos in the hard set, the Partition Matrix model is able to capture the less changing target gesture trajectory pattern, out of the dramatically changing background noise.

In order to demonstrate that the proposed framework can perform well in arbitrary uncontrolled environments, we collected an even more challenging database called Warwick Hand Gesture Database (Fig.2). For the hard set, instead of 1-3 people moving in the background in the Palm Graffiti Digits database, there are 2-4 people in the background of our database. Hence the extent of distractions in the background is much more severe than the Palm Graffiti Digits database. Fig.6 shows the tracking results of Adaptive SURF Tracking on samples of gesture six of hard sets from the two databases. It is obvious that the intra-class variance of our database is larger than the Palm Graffiti Digits database. Also, unlike [3] the training set of our database does not require the performers wearing coloured gloves. Since the work of [1] is the one most similar with the proposed method, we compared the performances between these two methods. The results are shown in Table 2 and Fig.7.

TABLE II. EXPERIMENTS ON PALM GRAFFITI DIGITS DATABASE

Warwick hand gesture database		
	Easy set	Hard set
Bao et al. ICEICE 2011 [1]	71.00%	18.20%
The proposed method	**93.00%**	**84.40%**

The results of running method of [1] and the proposed method on Warwick Hand Gesture Database.

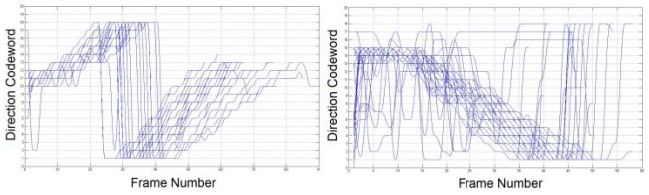

Figure 6. Movement direction vectors for the gesture of digit 6 of the training set of: Palm Graffiti Digits database (left) and Warwick Hand Gesture database (right). The horizontal axis is the frame number while the vertical axis is the direction codeword (1-18).

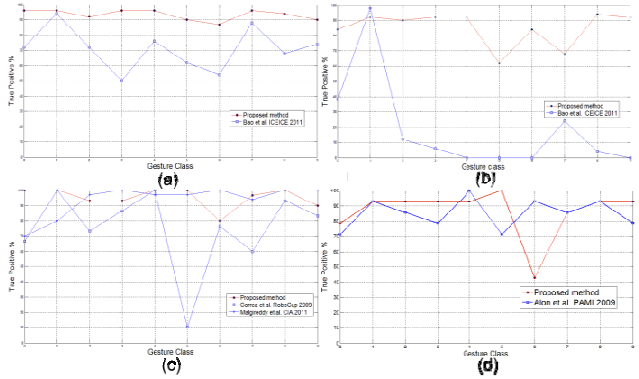

Figure 7. The comparison of performances on (a): easy set of the Warwick hand gesture database; (b) hard set of the Warwick hand gesture database; (c): easy set of the Palm Graffiti Digits database; (d): hard set of the Palm Graffiti Digits database. The horizontal axis is the gesture label while the vertical axis is the recognition rate.

Compared with other applications of using statistical models for pattern recognition tasks, our experiments have small amount of training samples. Hence the HCRF model can only learn limited patterns. Under the circumstances, our proposed framework can still produce satisfying performances. The reason is that instead of only produce partition value once, as all CRF related model, the Partition Matrix model gives restrictedly trained statistical model as many attempts as possible to use potential function matching the patterns in one testing sample. Then those partition values can reveal the confidence of ROIs being target hands and split the partition values on unstable ROIs into different gesture classes, which increases the inter-class variance. For the challenge of hand out of the scene, arm section with skin or non-skin texture will naturally be tracked until the hand is returning to the scene. When the hand is pausing during the gesture, the ROI will remain the last position with at least one pair of matched SURF key points detected.

As for speed, the proposed framework can run in average at: 55.00 ms/frame for easy sets, 56.75ms/frame for hard sets, on both experiments. That is 18.18 frames/sec and 17.64 frames/sec, respectively. Experiments were performed on a common 3.3GHz 4-core 8GB RAM Windows machine with C++ implementation. Hence our framework is able to perform comfortably in real time.

V. CONCLUSIONS

In this paper, a framework for real-time hand gesture recognition in uncontrolled environments is proposed. A novel tracking scheme called Adaptive SURF Tracking is combined with a novel classification model called Partition Matrix to recognise hand gestures from unconstrained background with multiple people moving randomly. The framework can survive challenges from the uncontrolled environments, including complex background, skin-coloured regions moving in background, performers wearing short-sleeve and face overlapping with hand. Scale, speed and location invariance are also achieved. The proposed framework was tested on Palm Graffiti Digits Database [3], and achieved 95.33% on the easy set, 86.43% on the hard set. It was also tested on Warwick Hand Gesture database, achieved 93.00% on the easy set and 84.40% on the hard set.

ACKNOWLEDGMENT

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

REFERENCES

[1] Jiatong Bao, Aiguo Song, Yan Guo, Hongru Tang, "Dynamic hand gesture recognition based on SURF tracking," International Conference on Electric Information and Control Engineering - ICEICE , 2011.

[2] Mahmoud Elmezain, Ayoub Al-Hamadi, Bernd Michaelis, "A robust method for hand gesture segmentation and recognition using forward spotting scheme in Conditional Random Fields," International Conference on Pattern Recognition-ICPR, pp.3850-3853, 2010.

[3] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff. "A unified framework for gesture recognition and spatiotemporal gesture segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), pp.1685 - 1699, Sep. 2009.

[4] Herbert Bay, Andreas Ess, TinneTuytelaars, Luc Van Gool, "SURF: speeded-up robust features," Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp.346-359, 2008.

[5] Mauricio Correa, Javier Ruiz-del-Solar, Rodrigo Verschae, Jong Lee-Ferng, Nelson Castillo, "Real-time hand gesture recognition for human robot interaction," RoboCup 2009: Robot Soccer World Cup XIII , Springer Berlin Heidelberg, Volume 5949, pp.46-57, 2010.

[6] Manavender R. Malgireddy, Ifeoma Nwogu, Subarna Ghosh, Venu Govindaraju, "A shared parameter model for gesture and sub-gesture analysis," Combinatorial Image Analysis, Springer Berlin Heidelberg, Volume 6636, pp 483-493, 2011.

[7] A. Quattoni, S. Wang, L.P. Morency, M. Collins, and T. Darrell, "Hidden-state Conditional Random Fields," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), pp.1848 - 1852, Oct. 2007.

[8] Paul Viola, Michael J. Jones, "Robust real-time face detection," International Journal of Computer Vision, Volume 57, pp. 137-154, 2004.

[9] Michael J. Jones , James M. Rehg, "Statistical color models with application to skin detection", International Journal of Computer Vision, Volume 46 Issue 1, pp.81 - 96, January 2002

[10] Dong C. Liu, Jorge Nocedal, "On the limited memory bfgs method for large scale optimization," Mathematical Programming, Springer-Verlag, Volume 45, Issue 1-3, pp.503-528, 1989.