Computer Vision and Image Understanding 148 (2016) 87–96

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Spatio-Temporal Hough Forest for efficient detection–localisation–recognition of fingerwriting in egocentric camera

Hyung Jin Chang[*], [**], Guillermo Garcia-Hernando[**], Danhang Tang, Tae-Kyun Kim

Department of Electrical and Electronic Engineering, South Kensington Campus, Imperial College London, SW7 2AZ, United Kingdom

ARTICLE INFO

Article history:
Received 15 April 2015
Accepted 7 January 2016

Keywords:
Spatio-Temporal Hough forest
Fingerwriting posture recognition
Fingertip detection
Handwritten character recognition
Egocentric vision

ABSTRACT

Recognising fingerwriting in mid-air is a useful input tool for wearable egocentric camera. In this paper we propose a novel framework to this purpose. Specifically, our method first detects a writing hand posture and locates the position of index fingertip in each frame. From the trajectory of the fingertip, the written character is localised and recognised simultaneously. To achieve this challenging task, we first present a contour-based view independent hand posture descriptor extracted with a novel signature function. The proposed descriptor serves both posture recognition and fingertip detection. As to recognising characters from trajectories, we propose Spatio-Temporal Hough Forest that takes sequential data as input and perform regression on both spatial and temporal domain. Therefore our method can perform character recognition and localisation simultaneously. To establish our contributions, a new *handwriting-in-mid-air* dataset with labels for postures, fingertips and character locations is proposed. We design and conduct experiments of posture estimation, fingertip detection, character recognition and localisation. In all experiments our method demonstrates superior accuracy and robustness compared to prior arts.

1. Introduction

Recent introduction of different wearable cameras in the market such as Google Glass or GoPro has given rise to the study of vision from an egocentric viewpoint and its potential novel applications. One of the most important characteristics of egocentric viewpoint is that hands are very present in the scene as was discussed in [1–9]. Thus, users' hands play an important role in the user interaction with the device and the world. Market demand of interactivity with these new devices leads to the study of new paths in terms of human–computer interaction (HCI) and human–robot interaction (HRI). As wearable cameras are usually small and lack a keyboard or similar input accessories, user hand gestures can serve as a natural and unobtrusive input. Spatio-temporal trajectories generated by hand movements in mid-air can represent handwritten characters, which in later steps can be used as text input to the wearable system.

A vision-based system for recognising handwritten trajectories in mid-air is not a new problem and some authors have proposed different approaches in the last two decades highly depending on available hardware and mainly from a third person viewpoint. In the context of augmented reality (AR) and using a sophisticated device with an infrared and colour sensor, Oka *et al.* [10] obtained promising results tracking fingertips and recognising simple geometric shapes trajectories. Using a standard colour camera, a mid-air handwritten recognition framework was proposed by Alon *et al.* [11] as one application of their approach to gesture recognition and spatio-temporal segmentation. Trajectories representing digits were collected using colored gloves with the user facing the camera. Using a stereo camera system in front of a virtual blackboard, Schick *et al.* [12] proposed a hand-tracking approach that relieved users of wearing special sensors or clothes. With the arrival of commodity depth sensors such as Microsoft Kinect or Leap Motion, in [13–17] the use of these sensors to recognise fingertips and handwritten trajectories is explored. In our application, using depth sensors allows segmenting the hands easily using a simple distance filter in contrast to other RGB camera based approaches where lighting conditions and background severely affects the segmentation quality.

To the best of our knowledge, from an egocentric point of view the problem remains quite unexplored; however there are some early approaches related to our work. In [18–22], colour cameras from a first-person viewpoint were used to recognise fingerwriting

[*] Corresponding author. Fax: +44 0 20 7594 6274.

[**] These authors contributed equally to this work.
E-mail address: hj.chang@imperial.ac.uk, ssacjin@gmail.com (H.J. Chang).

http://dx.doi.org/10.1016/j.cviu.2016.01.010
1077-3142/© 2016 Elsevier Inc. All rights reserved.

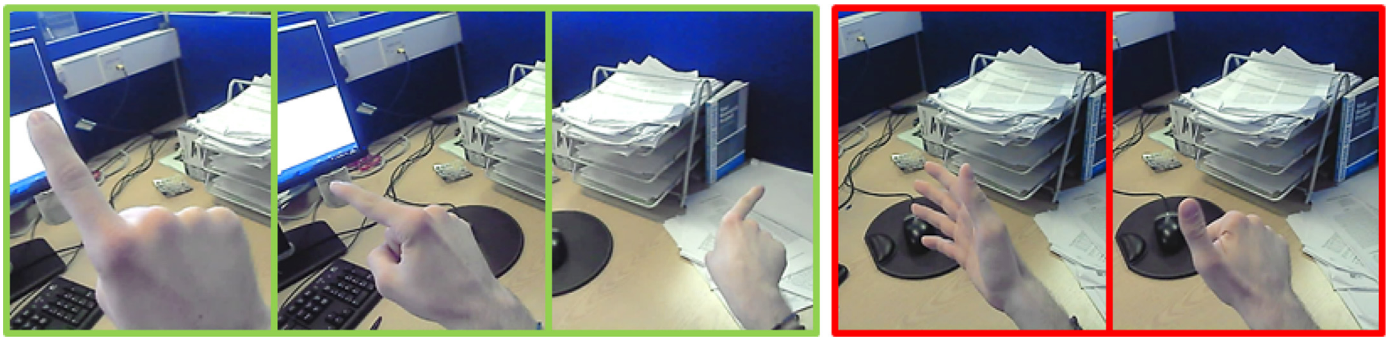

Fig. 1. Examples of images from our dataset when the user is writing (green) or not (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

by applying different techniques from sequence recognition field. In these previous approaches, the problematic of egocentric viewpoint is not considered as all the experiments were undertaken in very controlled conditions, where no challenging hand postures were present and the start and the end of writing were well specified.

In this paper, we focus on the problem of detecting and recognising human handwriting in mid-air using a head-mounted camera, and there are several unique challenges. Recognising the writing (pointing) hand posture in egocentric video suffers from the problem of fast viewpoint/scale changes. While writing, users tend to incline the hand forward unconsciously to be comfortable. This paradigm makes the problem of recognising the hand posture difficult due to high intra-class shape/scale variability. Because the hand is constantly present in egocentric view, it is necessary to differentiate between writing posture and other gestures based on appearance of user hands in the scene as shown in Fig. 1. After detecting the writing posture, accurate fingertip detection is needed to acquire the handwriting trajectory. Furthermore, how to recognise and localise each character reliably in an online manner is non-trivial because of different temporal length and speed.

To address all of this, we propose:

- A new view independent hand posture descriptor based on a novel signature function that leads to robust writing pose and fingertip detection.
- A novel framework called Spatio-Temporal Hough Forest (STHF), which leverages spatio-temporal information in one classification-regression framework, and performs character recognition and localisation simultaneously. To our best knowledge, this is the first Decision Forest extension that deals with sequential trajectory data.
- The first *Fingerwriting in mid-air* dataset captured in egocentric view, which has positive and negative poses, position of fingertips, as well as character labels of trajectories.

The paper is organised as follows: Section 2 discusses the related work. Section 3 presents the proposed framework detailing the new hand pose/fingertip detection and the STHF for character recognition. Section 4 introduces a new handwriting dataset and its use for evaluation. More detailed discussions with experimental analysis are presented. Finally, in Section 5 conclusions and issues to be addressed for future developments of the approach are discussed.

2. Literature review

Recognising different hand postures is a difficult and open problem in computer vision. Variation of illumination, point of view (e.g. 3D rotations, scale) and acquisition noise make the task very challenging. In the literature, two big families of methods can be found: generative and discriminative approaches. Generative approaches [23], which aim to recover the full 3D pose of the hand via 3D model fitting, are not suitable for our application, since its high computational cost is unfavourable of fast hand movement. Discriminative methods [9,24–26] directly construct mappings between training and testing poses, which is efficient but requires large amount of training data. In our work we aim to recognise a particular hand posture from a binary image describing a silhouette as a result of a previous segmentation stage. Thus, discriminative methods are the most suitable to our purpose.

Many different approaches have been proposed for the general problem of hand shape feature representation and recognition [27,28] and some of them have been applied to hand posture recognition. These previous works can be divided into region-based [24] and contour-based [29–31], depending on whether features are extracted only from the entire shape region or from the contour. As to region-based techniques, recently Hu and Yin [24] has proposed a topological-based feature descriptor which describes the behaviour of the holes between the hand region and its convex hull under morphological operations. This feature representation has been proved to be relatively accurate to differenciate between hand postures from similar view points, but it is not suitable to distinguish under drastic view point and shape changes. Among contour-based methods, shape context [29] performed well in hand posture recognition under controlled conditions, but its performance drastically dropped while varying the viewpoint. Another popular contour-based approach is the use of Fourier Descriptors (FD) [30], which permits to have an invariant hand-shape representation suitable for hand posture recognition [32–34]. A step further in the use of FD has been the use of signature functions [31]. *Signature functions* are one-dimensional functions that represent features derived from the shape contour: curvature, distance to the shape centroid, turning angle, etc.

Detection and tracking of fingertips has been an active topic in the fields of HCI and Augmented Reality (AR) using both colour and depth cameras. Model-based tracking of hands [9,25,26] can be used as a preceding stage to detect fingertips, but the high computational cost of these approaches and the need of a big amount of training data make them unsuitable for our purpose. A popular modeless approach consists in first segmenting the hand silhouette using colour or depth cues and detecting fingertips from the extracted binary shape. Following this line, many works focused on the structure of the hand by exploiting its geometrical properties to localise fingertip points. In contrast to other parts of the hand, fingertips are high curvature points, a property exploited by [35,36], where they used the contour curvature as a cue to detect fingertips. Another important characteristic of fingertips is that they are usually far from the hand palm. Using this, Bhuyan et al. and Liang et al. [37,38] used a distance metric from hand palm to the contour furthest points to localise candidate points, which

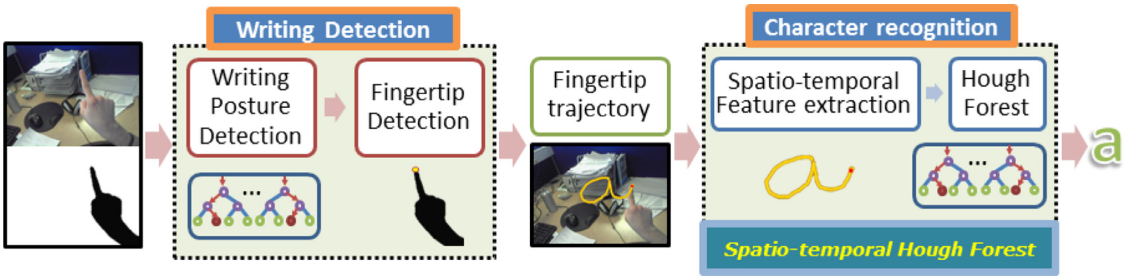

Fig. 2. Overview of the proposed method.

were afterwards refined by different techniques. Raheja et al. [39] proposed a two-step algorithm where fingertip is localised from hand edges after estimating the hand direction while Maisto et al. [40] also had into account the topological structure of the hand extracting points from the convex hull of the silhouette. An important drawback of these methods is that, for some hand postures, fingertips are not always over the hand silhouette edge. In order to lighter this assumption, Raheja et al. [13] detected fingertips as the hand points which are closer to the sensor after segmenting hand palm and fingers, an approach later followed by [41] and reinforced with a hand graph model similar to [42]. Krejov and Bowden [42] extended the distance concept enforcing it with the natural structure of hand using a geodesic distance. This helped to localise fingertips in hand configurations where previous methods failed. Most of these approaches assume that the palm is always faced to the camera, which is not an appropriate assumption in our application. However, as we are only interested in localising fingertip in one hand pointing posture makes our problem relatively easier to the works presented which aim to detect fingertip in any hand configuration.

There has been quite a few works about Random Forests [52] for spatio-temporal data analysis and modelling. Spatio-temporal relational probability trees were proposed [43,44] for probability estimation of spatially and temporally varying relational data. The approach was applied to weather process understanding [45], but their relational feature based tree building is not rigorous enough for visual data analysis. [46] extended the 2D object detecting Hough forests [47] to multi-class action detection in spatio-temporal domain. However, the method requires many dense spatio-temporal local features of relatively long video sequences for robust Hough voting, so on-line detection is impractical. In [48] a simultaneous action recognition and localisation method based on a vocabulary forest of local motion-appearance features was proposed. It works on data from uncontrolled environment with camera motion, background clutter and occlusion. However, this method also requires a large number of local features represented in many vocabulary trees which capture joint appearance-motion information. Yu et al. [49] proposed a random forest based voting method for action detection and search. They indexed each spatio-temporal feature by an unsupervised random forest indexing method. Local feature matching becomes much faster than existing nearest-neighbour-based methods. Although indexing each feature is computationally very fast, coarse-to-fine subvolume search scheme for action detection requires full sequences in an off-line fashion, which it is not suitable for on-line detection, especially for wearable device applications. Jang et al. [50] also proposed a spatio-temporal forest based 3D finger clicking action and position estimation method from egocentric view, but they consider one simple clicking action only and the temporal length is relatively short.

In this paper, we propose a unified framework that can process multiclass sequential trajectories for real-time writing character recognition and character centre position estimation simultaneously. To the best of our knowledge, there is no such method that can fulfil all the requirements except our newly proposed Spatio-Temporal Hough Forest.

3. Algorithm

In general, the overall proposed process can be described as three-fold (as shown in Fig. 2): (1) recognise writing hand poses from other gestures; (2) fingertip detection for each frame; (3) recognise characters from trajectories formed by sequentially detected fingertip locations. The following sections are organised accordingly.

3.1. Handwriting posture detection

Our approach to handwriting posture detection assumes that a hand has been pre-segmented successfully with, for instance, depth value thresholding or skin colour selection. For our application, where the user focus on writing and is not manipulating any object, we found that this is not a hard assumption using a depth camera.

To make our method independent from sensors and use only depth values for segmentation, we propose a new contour-based hand posture descriptor using Fourier Descriptors [30] extracted from a novel shape signature function. As shown in Fig. 3, the segmented hand is represented as a binary image, and then we consider a simple planar contour curve s_f extracted from the binary image of fth frame. Signature functions [31] are one-dimensional functions which represent discriminative features derived from the shape contour s_f: curvature, distance to the shape centroid, turning angle, etc. We propose a novel signature function based on a distance-weighted scale invariant measure of the contour curvature. The advantages of this new signature function are: it is a discriminative feature which permits a high accurate description of the hand posture, it is not computationally demanding as we only need to examine one scale and it allows us to reuse it for fingertip detection.

Scale invariant curvature measure: We propose to use a scale invariant measure of the curvature presented in [51] which we will refer to as *curvature entropy* u. If the contour s is length of L_s and sampled at n_s uniformly spaced points, we can consider an interval $\Delta s = L_s/n_s$. From point to point along the sampled contour curve, the tangent directional changes by an angle α can be defined. The curvature κ is a change in tangent direction as we move along the curve s [51]. It can be approximated by the ratio between the change in the tangent direction α and Δs:

$$\kappa(s_f) \approx \frac{\alpha}{\Delta s_f}. \qquad (1)$$

As derived in [51], *curvature entropy* can be approximated as follows:

$$u(\kappa(s_f)) \propto -cos(\kappa(s_f)\cdot\Delta s_f). \qquad (2)$$

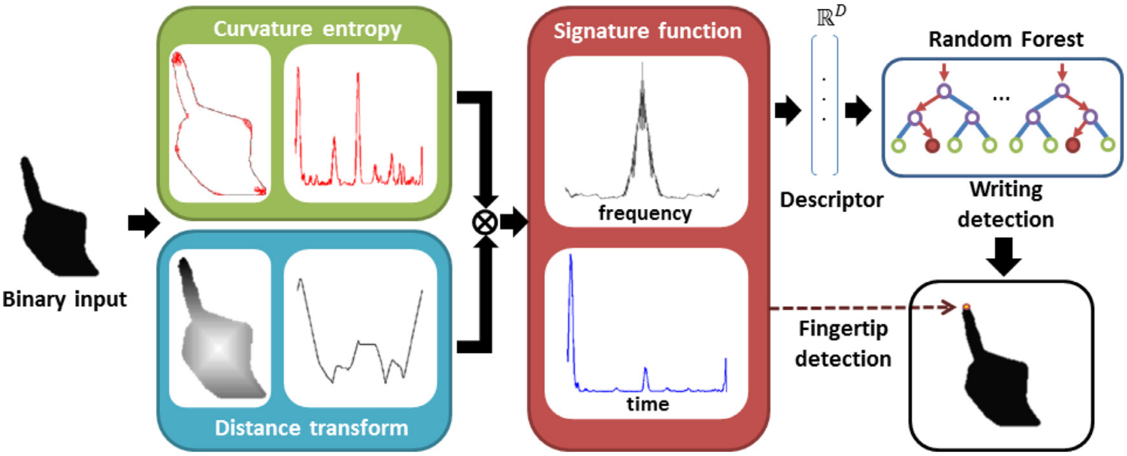

Fig. 3. Hand posture and fingertip detection framework.

The *curvature entropy* u is scale-invariant and it is locally proportional to its curvature $\kappa(s_f)$. This measure allows us to localise high curvature points without the necessity of exploring different scales (as shown in [35]) and relieve us from heavy computational load and extra parameter tunings. For the calculation of information along contours, our implementation is based on [51][1].

Signature function (Ψ):

We define a signature function of a contour $\Psi(s_f)$ as a combination of the *curvature entropy* along the contour $u(\kappa(s_f))$ and a distance transform $\delta(s_f)$ which represents distances of every contour point to the centre of mass of the hand (see Fig. 3):

$$\Psi(s_f) = u(\kappa(s_f)) \cdot \delta(s_f)^{\gamma}. \tag{3}$$

The parameter γ weights the impact of the distance in the signature function. It also attenuates high curvature points that are not fingertips reducing the false positives mainly caused by noise. This allows us to reuse the function to localise the fingertip.

Hand posture descriptor (**v**)*:* A signature function can be represented as a time series (for further clarity we will refer this as a time domain) with variable length due to the different scales of contours in images. In order to extract discriminative features, we work in the frequency domain rather than the temporal domain in behalf of the desirable properties of rotation and scale invariance that can be achieved with Fourier Descriptors. Once the Fourier series $a[n]$ and $b[n]$ are extracted from the signature function Ψ, we perform a normalisation step similar to [32], which make features invariant to rotation, translation and scale changes. This normalisation consists in defining a function $S(n)$ as:

$$S(n) = \frac{r(n)}{r(1)} \tag{4}$$

where $r(n) = [a(n)^2 + b(n)^2]^{1/2}$. We sample the function $S(n)$ to conform our hand posture descriptor $\mathbf{v} = (S(1), \ldots, S(D)) \in \mathbb{R}^D$. The number of samples (harmonics) D is determined experimentally and discussed on the experimental section.

Hand writing/no writing posture classifier: We use a standard random forest algorithm [52] as a classifier for our binary classification problem. Its desirable properties of being a powerful discriminative classifier and real-time capability makes it a good choice. We define the binary classification problem as two different classes: $\mathcal{C}_W = \{\text{writing, no writing}\}$. A random forest is an ensemble of T decision trees that assign a posterior class distribution to each leaf $p_t(c_W|\mathbf{v})$. The final class assignment is performed averaging all the trees: $p(c_W|\mathbf{v}) = \frac{1}{T}\sum_t^T p_t(c_W|\mathbf{v})$ and picking the most likely one using a Maximum A Posteriori (MAP) decision rule.

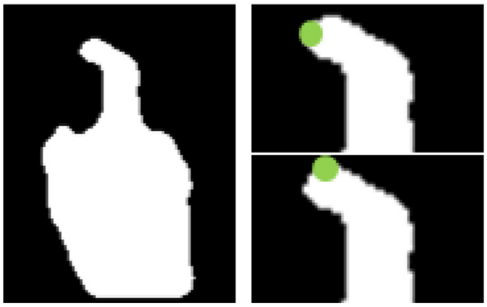

Fig. 4. Fingertip detection results (Right-top: proposed method, Right-bottom: Distance based method [37]).

3.2. Fingertip detection

The signature function presented in the previous section permits us to find the fingertip in a straightforward way. Fingertips have the property of being points of high curvature and distant from hand centre. Our signature function will have a peak at the fingertip position caused by a high curvature entropy value. This point will be also highly distant from the centre of the hand, thus it will be kept when combining with the distance function, while false positive points mainly caused by noise will be attenuated (see Fig. 3).

The main advantage of this approach over other curvature based ones [35,36] is that we do not need to examine different scales to find maximum curvature points relieving us from computational cost. The advantage over distance based methods [37] is that we obtain more accurate detections in cases where the furthest point is not exactly the fingertip, which occurs when the user lightly bends their finger or in certain viewpoints as shown in Fig. 4. In order to obtain smooth trajectories for the next stage of our algorithm, we filter the detected points with a Kalman filter.

3.3. Spatio-Temporal Hough Forest for character recognition and localisation

In this section, we present the STHF for recognising characters from the fingertip trajectories. In order to take sequences as inputs to the forest, we present a new input vector encoding method first. For each frame, we encode the information within an

[1] Code available at http://ruccs.rutgers.edu/~manish/demos/curveinfo.html

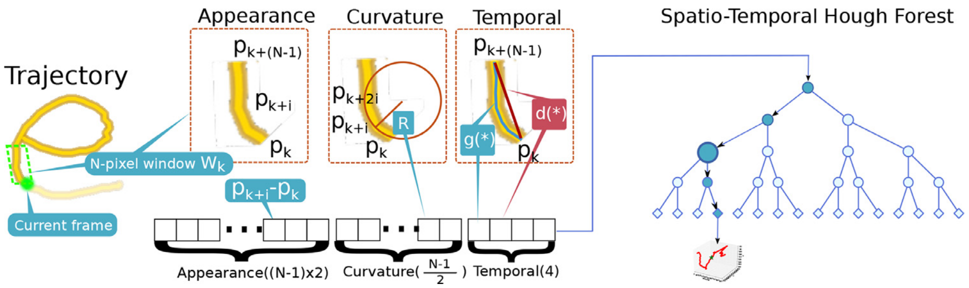

Fig. 5. Spatio-temporal feature for character recognition. It consists of three terms: appearance, curvature and temporal. The numbers in brackets indicate dimension of each term.

N-points sliding window $W_k = \{p_k, \dots, p_{k+(N-1)}\}$ into a tuple as feature $f(W_k) = [\mathbb{A}, \mathbb{C}, \mathbb{T}]$ where $\mathbb{A}$ is a non-parametric appearance term, $\mathbb{C}$ is a parametric term describing the curvature information, and $\mathbb{T}$ gives us temporal information within W_k (see Fig. 5).

Appearance term ($\mathbb{A}$) is a $2 \times (N-1)$ dimension vector, which is defined as:

$$\mathbb{A}(W_k) = \|_{i=1}^{N-1} \ (p_{k+i} - p_k), \tag{5}$$

where p_i is a 2D position vector of point i ($p_i = [x_i; y_i]$) and the p_k indicates the first point of W_k. $\|$ is a vector element concatenation operator. This term represents the relative shape of the cropped trajectory W_k.

Curvature term ($\mathbb{C}$) has the dimension of $\frac{N-1}{2}$. Here we apply Menger Curvature [53] to capture the shape of the curvature within W_k. The idea of [53] is to approximate the curvature with a circle that is given by a triple of points on the curvature, and then use reciprocal of the circle radius as final representation. This approximation makes the feature generalised better to different writings. To be more robust, we incrementally select 3 points from the curvature, as formulated below:

$$\mathbb{C}(W_k) = \|_{i=1}^{(N-1)/2} \ \frac{4Area(p_k, p_{k+i}, p_{k+2i})}{|p_k - p_{k+i}||p_k - p_{k+2i}||p_{k+i} - p_{k+2i}|}, \tag{6}$$

where $Area(p_k, p_{k+i}, p_{k+2i})$ is the area spanned by selected point triplet (p_k, p_{k+i}, p_{k+2i}). The $Area(p^1, p^2, p^3)$ of three points (p^1, p^2, p^3) is calculated by

$$Area(p^1, p^2, p^3) = |\frac{p_x^1(p_y^2 - p_y^3) + p_x^2(p_y^3 - p_y^1) + p_x^3(p_y^1 - p_y^2)}{2}| \tag{7}$$

where p_x and p_y are the x and y coordinates of the point p.

Temporal term ($\mathbb{T}$) is a 4-dimensional vector defined as below:

$$\begin{aligned}\mathbb{T}(W_k) &= g(p_k, p_{k+(N-1)}) \parallel d(p_k, p_{k+(N-1)}) \\ &\quad \| \dot{g}(p_k, p_{k+(N-1)}) \parallel \dot{d}(p_k, p_{k+(N-1)}),\end{aligned} \tag{8}$$

where $g(\ \cdot\ )$ stands for geodesic (along the writing trajectory of the fingertip) distance, $d(\ \cdot\ )$ stands for the Euclidean distance, $\dot{g}(\cdot)$ and $\dot{d}(\cdot)$ stand for velocity in geodesic and Euclidean space respectively. This term can imply various temporal writing properties such as various stroke speeds depending on each character. Also by considering both the geodesic and Euclidean distance, this term can represent various stroke combinations (*e.g.*, an arc after a straight line, a straight line or a circle, etc.) especially when all the combinations are written at the same speed.

Classification and regression: We formulate the character recognition as a multi-class classification problem and character centre localisation as regression. To perform them simultaneously, we build upon Hough forest [47] for its good properties. Firstly, the Hough forest is an ensemble of classification-regression trees that interweaves classification and regression tasks; secondly, its multi-class handling ability inherited from the standard random decision forest [52] makes it well-suited for our 26-class (26-character; $\mathcal{C}_C = \{a, b, \dots, z\}$) problem; last but not least, its efficiency in both training and testing is favourable when large sequential data is given. To extend it to temporal domain, for each training sequence, we calculate *character centres* $\{\bar{\Delta}$ and $\bar{\Upsilon}\}$ (in spatial domain and temporal domain respectively) of each character.

Each tree T in the forest $\mathcal{T}$ is constructed from a set of features $\{\mathcal{P}_k = (f(W_k), c_k, d_k^s, d_k^t)\}$ that are sequentially generated from fingertip trajectories where $f(W_k)$ is the encoded features of fixed size in $\mathbb{R}^{(2(N-1)+(N-1)/2+4)}$, c_k is the class label, and d_k^s and d_k^t are displacement vector from the first point p_k of W_k to the spatial/temporal character centre respectively.

Each leaf node L stores the probability of the cropped trajectory W_k belonging to the class $p(c|L)$, estimated by the proportion of features per class label reaching the leaf after training, and $D_c^L = \{d_k^s, d_k^t\}_{c_k=c}$ the cropped trajectories' respective displacement vectors. During training, each split node is assigned a binary test in relation to the encoded vector $f(W_k)$. The binary test at a split node is defined by a comparison of a feature value in feature channel j with a threshold τ:

$$\theta_{j,\tau}(\mathcal{P}) = \begin{cases} 0 & \text{if } \mathcal{P}^j < \tau \\ 1 & \text{otherwise.} \end{cases} \tag{9}$$

The ideal binary test splits a set of the sequentially encoded vectors $\mathbb{P} = \{\mathcal{P}_k = (f(W_k), c_k, d_k^s, d_k^t)\}$ to minimise the uncertainty of their class label and spatio-temporal displacement vectors. To this goal, we use three measures to evaluate the uncertainty for a set $\mathbb{P}$. Accordingly, the information gain (IG) for each split node can be described as below:

$$IG = \begin{cases} U_1 := \sum_{i=1}^{|\mathcal{C}_C|} -p_{c_i} log(p_{c_i}), \\ U_2 := \sum_{j=1}^{|n=\{Left,Right\}|} ||(d_k^s)_j^n - \bar{\Delta}^n||^2, \\ U_3 := \sum_{j=1}^{|n=\{Left,Right\}|} ||(d_k^t)_j^n - \bar{\Upsilon}^n||^2, \end{cases} \tag{10}$$

where U_1 is a class entropy and c_i indicates each class; U_2 is the spatial variance and $\bar{\Delta}$ is the spatial centre of each character; similarly defined, U_3 is a temporal variance and $\bar{\Upsilon}$ is the aforementioned temporal centre.

At each node during training, a pool of binary tests $\{\theta^k\}$ is generated with random values of j and τ, and either class or spatial/temporal displacement uncertainty is randomly chosen to be minimised. Similarly to the standard Hough forest growing, the encoded vector set arriving at the node is evaluated with all binary tests in the pool. The binary test satisfying the following objective is stored:

$$\underset{k}{\text{argmin}}(U_\star(\{\mathcal{P}_i|\theta^k = 0\}) + U_\star(\{\mathcal{P}_i|\theta^k = 1\})), \tag{11}$$

where $\star$ indicates the chosen uncertainty measure for the node (U_1, U_2 or U_3). By randomly selecting the uncertainty measure, nodes decreasing both class and displacement uncertainty are interleaved throughout the tree. At the leaf nodes, both class distribution and offset vectors are stored.

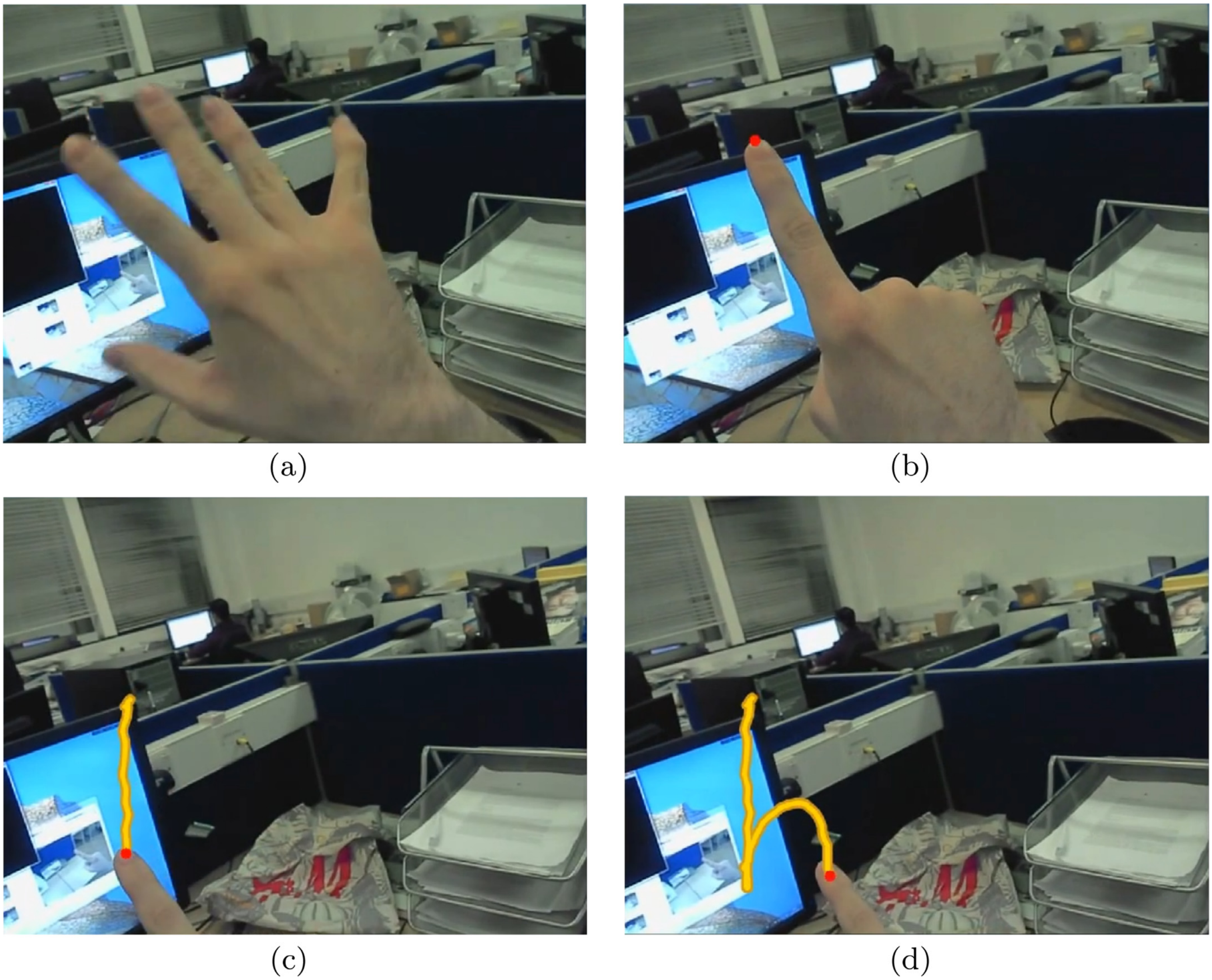

Fig. 6. All figures show different stages of our framework in action. (a) A non-writing hand posture, no fingertip is detected and the system is in pause. (b) User starts to write, handwriting posture is detected. Fingertip is tracked in successive frames. (c) User in process of writing, when enough spatial-temporal points are buffered, on-line recognition starts. (d) User finished writing character and a ‘h’ is recognised.

For testing, sequentially encoded $(2(N-1)+(N-1)/2+4)$-dimensional feature vectors are passed through each tree in the trained forest. Starting at the root of the STHF, the feature vector traverses the tree, branching left or right according to the split node function, until reaching a leaf node. Using the stored class distribution and offsets at the leaf nodes, each leaf node votes for its corresponding class label and spatio-temporal centre location. Aggregating votes of all trees, we locate the final class and centre position of the writing trajectory. Especially to find the centre point, we used a mean shift mode seeking method [54].

4. Experiments

4.1. Experimental environment and new dataset

To conduct the experiments, we recorded a dataset which contains two parts to test our work: labelled images for testing our hand posture descriptor with fingertip ground-truth in writing poses and fingertip trajectories describing alphabet characters. We mounted a depth sensor (Creative∗ Interactive Gesture Camera) to a cap to record gestures in egocentric view. However, depth information is only used to segment the hand applying a distance filter. For every frame, we get the binary mask resulting from the segmentation. The dataset also contains depth frames for further research.

The hand posture dataset consists of 8000 images from two classes: {‘writing’, ‘no writing’}. It has an approximate ratio of 1: 3 for ‘writing/no writing’ containing challenging poses that naturally occur in egocentric vision such as rotations out-of-plane of the hand, poses corrupted by noise and missing points due to the limitations of the sensor and the simple segmentation stage. The 2500 images in ‘writing’ class have been manually labelled with fingertip positions.

The character dataset contains 260 fingertip trajectories by a single actor (As an egocentric device is a personalised device, we focus on recognising a specifically personalised writing pattern rather than general cases). A trajectory represents one English alphabet character (from *a* to *z*) from an egocentric viewpoint. In total, 10 samples of 26 different characters have been recorded. Each trajectory consists of a set of points (x, y, t) which correspond to the position (x, y) of the fingertip at time t. This dataset will be available after the publication of this paper.

Table 1
Hand posture recognition performance of various hand descriptors with different number and depth of trees.

Maximum depth	Number of trees	Descriptor method				
		FD [32]	MSBNM [24]	Distance	Curvature entropy	Proposed
12	5	69.8	88.3	93.8	94.7	98.6
	10	71.8	87.8	95.9	94.9	98.7
	20	69.1	88.0	95.1	95.3	98.7
	30	69.4	88.2	95.8	95.4	98.8
	40	69.7	88.0	95.6	95.4	98.8
	50	68.6	88.4	95.8	95.3	98.9
14	5	72.4	88.4	93.8	94.7	98.6
	10	71.0	88.9	95.9	94.9	98.7
	20	69.8	89.1	95.1	95.3	98.7
	30	69.3	89.2	95.8	95.4	98.8
	40	69.1	89.2	95.6	95.4	98.8
	50	69.8	89.4	96.3	95.8	99.0
16	5	74.0	89.5	95.3	95.4	98.8
	10	72.4	89.6	95.9	95.8	98.8
	20	70.1	89.9	96.3	96.0	98.8
	30	71.1	90.0	96.7	96.0	99.0
	40	70.4	90.0	96.6	96.1	98.9
	50	70.5	90.2	96.4	96.1	98.9
18	5	**75.2**	89.8	96.1	95.5	99.0
	10	74.9	89.7	96.4	96.0	98.9
	20	71.6	90.3	96.6	**96.4**	98.9
	30	72.9	90.3	96.8	96.2	99.0
	40	72.9	90.4	96.9	96.3	**99.1**
	50	72.9	**90.4**	**97.2**	96.2	99.0

Table 2
Recognition performance comparisons of fingertip detection and writing characters.

Recognition	Method	Accuracy (%)
Fingertip detection	Raheja *et al.* [39]	91.5
	Distance based [37]	94.9
	Proposed	**97.7**
Character recognition	HMM (20 states) [32]	66.4
	DTW [16]	78.5
	Conventional Random Forest [52]	79.6
	Proposed (w/o temporal term)	82.7
	Proposed (spatio-temporal term)	**90.4**

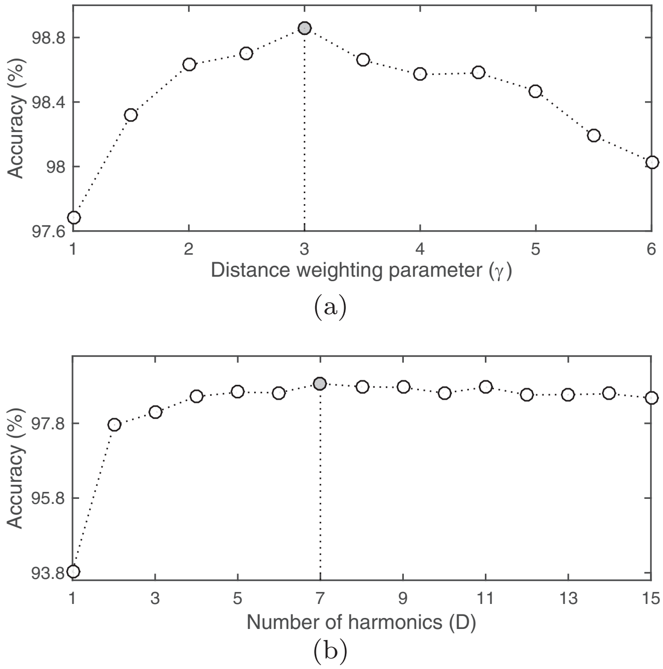

Fig. 7. Parameter influences of the distance weighting parameter γ and the number of harmonics D in hand writing posture detection.

The proposed descriptor and fingertip detection are implemented on an Intel Core i7-2600 with 16GB RAM in C++, and the STHF is implemented in Python separately. Fig. 6 shows captured images in different stages of our framework in action.

4.2. Validation of the hand posture descriptor and fingertip detection

We have performed various experiments to show the suitability of the proposed hand posture descriptor to our problem. All the experiments have been performed using 8000 binary labelled images from our dataset. All the results presented in this section are the result of 10-fold cross validation using a standard Random Forest classifier [52] and using a resolution of 10 pixels in the computation of the curvature entropy.

Hand posture descriptor: We compared our approach with one state-of-the art region-based method [24] and one contour-based method using Fourier Descriptors [32] extracted from contour coordinates. Our signature function is a combination of two signature functions: curvature entropy and distance to hand centre. For this reason, we also tested both functions individually in order to study the impact of their combination. Table 1 summarises the results for each approach varying the two important parameters of a Random Forest classifier: tree number and maximum depth. Our proposed descriptor shows a better performance over the baseline methods and over the individual signature functions for all the combinations of parameters. The best recognition accuracy for our proposed descriptor is achieved with a Random Forest of 40 trees and 18 levels as maximum depth.

We have also performed parameter analysis experiments to show the influence of the parameters of our descriptor, the distance weighting parameter γ and the number of harmonics extracted to conform the feature vector D, for a fixed classifier parameters. As shown in Fig. 7(b), only a limited number of harmonics is needed to conform the feature vector, obtaining the highest accuracy with the first 7 of them. Using a higher number of harmonics does not improve the accuracy as all the information, in form of energy, is concentrated on the low frequencies of the

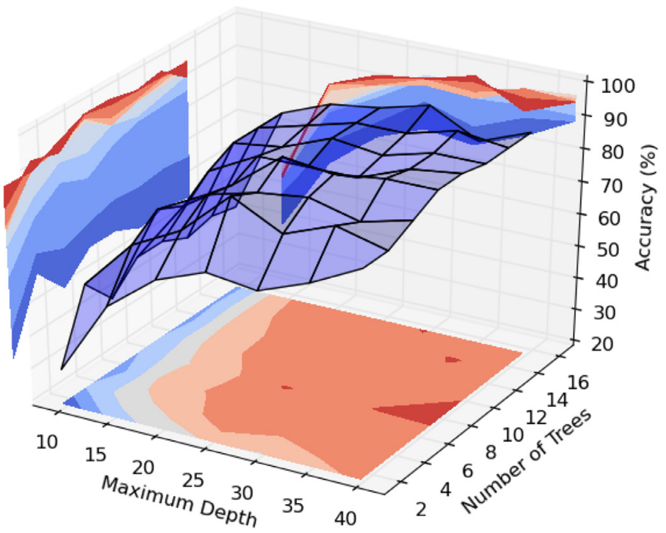

Fig. 8. Various training parameters of STHF *vs.* classification accuracy.

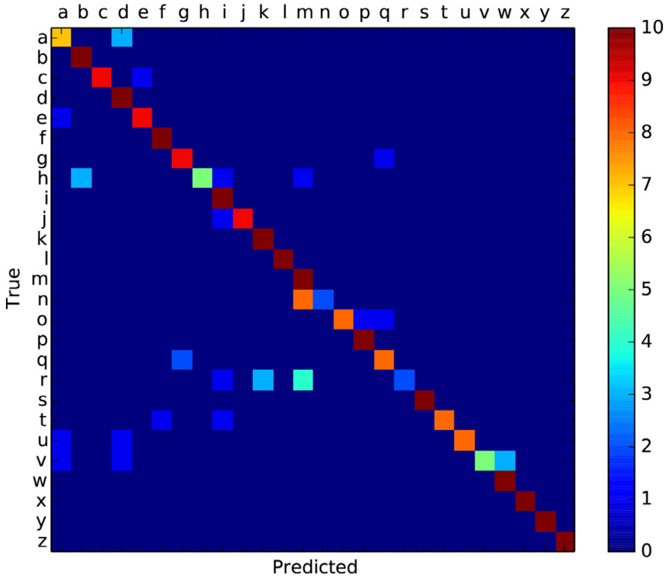

Fig. 9. Confusion matrix of character recognition results by the proposed method.

spectrum as can be seen in Fig. 3. The parameter γ describes approximately a quadratic function (Fig. 7(a)) in terms of accuracy with a maximum found in 3.

Fingertip detection: In order to test our fingertip detection approach quantitatively, we used the 2500 manually labelled images from our dataset. As a measure of error, we computed an Euclidean distance between the estimated fingertip location $\hat{p} = (\hat{x}, \hat{y})$ and the actual ground truth $p = (x, y)$. We considered a detection correct if its distance was less than 3 pixels to the ground truth. We compared our approach against two different methods. The first method ([37]) uses only the geodesic distance from the handshape contour to the centre of the hand palm, without exploiting the curvature cue. We also compare to the method presented by Raheja *et al.* [39] where fingertip points detection is tackled as edge detection of the hand binary shape. The results are presented on Table 2. We observe that our approach outperforms previous methods. The novel combination of curvature and distance information permits us to have accurate estimations of the fingertip position in cases where using only distance information performs poorly (see Fig. 4). Our method is also more robust to false positives than [39] as we do not only look for edge points but also consider its curvature, which is an important characteristic of fingertips. Furthermore, we do not need to first estimate the hand orientation as our signature function is rotation-invariant, which lightens us from extra model parameters.

For this configuration, the computation time of extracting one descriptor, passing it through the forest and the fingertip detection is 2 ms on average. As we can see from the experimental results, the proposed hand posture detection error is 0.9% and the fingertip detection error is 2.3%, which are very low. As a result of these low errors and the use of a Kalman filter for smoothing the trajectory, cascaded error becomes negligible.

4.3. Fingerwriting character recognition and localisation

Character recognition: We investigate the effect of several training parameters on classification accuracy. In Fig. 8 we show how the maximum depth and the number of trees affect accuracy in a 10-fold cross validation setting. For training, we set the window size N of W_k as 21 and on average 9299 features are used for training and 1033 features for testing. All the parameters, maximum depth appears to affect most significantly as it directly controls the model capacity of the forest. Based on the experimental results we fixed the number of trees as 8 and the maximum depth as 25.

We compared the proposed STHF's character recognition performance with other well-known sequential data analysis based handwriting recognition methods [16,32]. Sequential data extracted from fingertip movement trajectory is used for classification among different character classes. Hidden Markov models (HMM) [55,56] are generative models widely used for sequential data modelling. In this paper, we have implemented a HMM system with continuous observation model for classification instead of HMM with discrete observation symbols. For comparison, we have also tested our approach with a discriminative classifier, conventional Random Forest using majority-vote rule [52].

All the results in Table 2 are a result of 10-fold cross validation. The proposed method shows the best recognition performance. Especially we can see that the temporal variance affects recognition performance as well. The recognition rate of the conventional random forest classifier is slightly lower. This is due to high temporal variation in same class induced by temporal information. As we can see from the confusion matrix in Fig. 9, most characters were well recognised, but some similar shape characters were confused often such as *'a'–'d', 'b'–'h','g'–'q', 'm'–'n'–'r', 'o'–'p'–'q'* and *'v'–'w'*. Most errors were caused by these similar characters. Actually those characters are confused sometimes even by human. The erroneous character recognition might be able to be corrected in the context of a word writing in the future work.

Character centre localisation: Our method can also correctly localise spatio-temporal centre of each character writing by spatio-temporal offset Hough voting. The mean shift method [54] is used to find the centre points. Fig. 10 shows some localisation results in spatio-temporal space. As we can see, estimated centres are similar to ground-truth points. The writing centre information of each character can be used as an important clue for segmenting each character in a word.

5. Conclusion and future works

In this paper we presented a new efficient framework for mid-air fingerwriting recognition for egocentric camera. In natural cases, same hand poses look totally different in egocentric view point. By proposing a new hand posture descriptor, we could

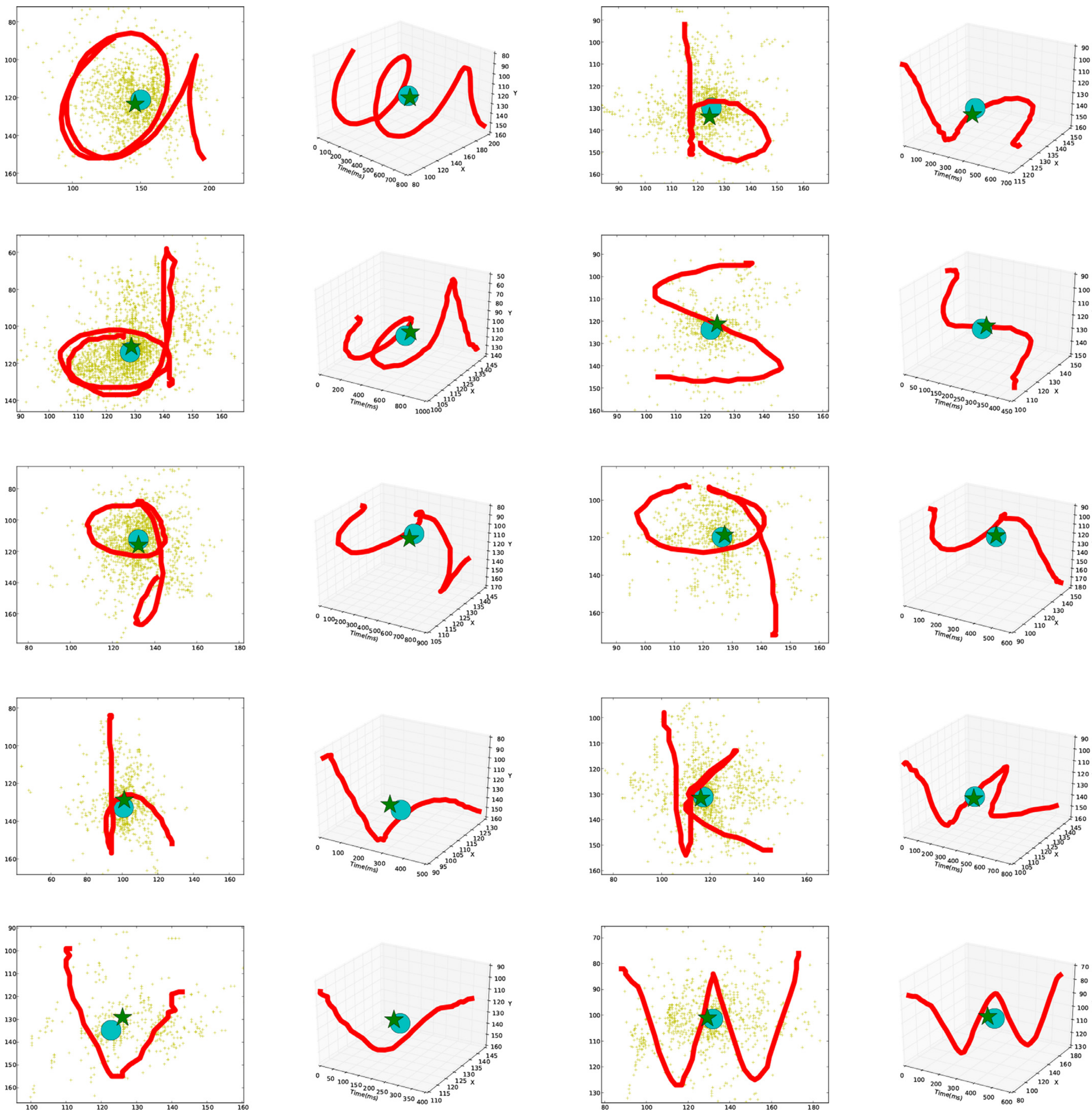

Fig. 10. Character centre localisation results. Small yellow crosses are spatio-temporal offset voting points. Blue circles are estimated centre positions of each character and green stars indicate ground-truth centre locations.(Best shown in colour. More results will be in supplementary materials.). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

achieve robust writing hand posture detection and fingertip localisation simultaneously. Also, the newly proposed STHF could successfully localise and recognise each character from fully connected trajectory sequence. Furthermore, we presented a new mid-air fingerwriting dataset which is the first dataset in this application. Experimental results showed that the proposed framework achieves the best recognition rate with localisation. As a future work, we are going to extend the STHF framework in a hierarchical manner to recognise words by localising and recognising each character first and then modelling temporal sequence of each character. Moreover, our newly proposed algorithm for hand posture and fingertip detection can be solely applied to HCI/HRI applications as a new simple input technique; applications such as interactive attention point indication, interactive focus changing in photo taking, and giving order to a robot by a direction.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.cviu.2016.01.010

References

[1] W. Mayol, D. Murray, Wearable hand activity recognition for event summarization, in: Proceedings of the Ninth IEEE International Symposium on Wearable Computers, 2005, pp. 122–129.

[2] A. Fathi, A. Farhadi, J.M. Rehg, Understanding egocentric activities, in: Proceedings of the ICCV, 2011.

[3] A. Fathi, X. Ren, J.M. Rehg, Learning to recognize objects in egocentric activities, in: Proceedings of the CVPR, IEEE, 2011, pp. 3281–3288.

[4] H. Pirsiavash, D. Ramanan, Detecting activities of daily living in first-person camera views, in: Proceedings of the CVPR, IEEE, 2012, pp. 2847–2854.

[5] A. Behera, D.C. Hogg, A.G. Cohn, Egocentric activity monitoring and recovery, in: Proceedings of the ACCV, 7726, 2012.

[6] C. Li, K.M. Kitani, Pixel-level hand detection in ego-centric videos, in: Proceedings of the CVPR, 2013.

[7] C. Li, K.M. Kitani, Model recommendation with virtual probes for egocentric hand detection, in: Proceedings of the ICCV, 2013.

[8] Y. Li, A. Fathi, J.M. Rehg, Learning to predict gaze in egocentric video, in: Proceedings of the ICCV, 2013.

[9] G. Rogez, M. Khademi, J. Supancic III, J. Montiel, D. Ramanan, 3d hand pose detection in egocentric RGB-d images, in: Proceedings of the Consumer Depth Cameras for Computer Vision- ECCV Workshop, 2014.

[10] K. Oka, Y. Sato, H. Koike, Real-time fingertip tracking and gesture recognition, IEEE Comput. Gr. Appl. 22 (6) (2002) 64–71.

[11] J. Alon, V. Athitsos, Q. Yuan, S. Sclaroff, A unified framework for gesture recognition and spatiotemporal gesture segmentation, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 31 (9) (2009) 1685–1699.

[12] A. Schick, D. Morlock, C. Amma, T. Schultz, R. Stiefelhagen, Vision-based handwriting recognition for unrestricted text input in mid-air, in: Proceedings of the 14th ACM International Conference on Multimodal Interaction, ACM, 2012, pp. 217–220.

[13] J.L. Raheja, A. Chaudhary, K. Singal, Tracking of fingertips and centers of palm using kinect, in: Proceedings of the Third International Conference on Computational Intelligence, Modelling and Simulation (CIMSiM), IEEE, 2011, pp. 248–252.

[14] Z. Feng, S. Xu, X. Zhang, L. Jin, Z. Ye, W. Yang, Real-time fingertip tracking and detection using kinect depth sensor for a new writing-in-the air system, in: Proceedings of the 4th International Conference on Internet Multimedia Computing and Service, in: ICIMCS '12, 2012.

[15] X. Zhang, Z. Ye, L. Jin, Z. Feng, S. Xu, A new writing experience: Finger writing in the air using a kinect sensor, IEEE MultiMedia 20 (4) (2013) 85–93.

[16] S. Vikram, L. Li, S. Russell, Handwriting and gestures in the air, recognizing on the fly, in: Proceedings of the Computer Human Interaction (CHI), 2013.

[17] R. Aggarwal, S. Swetha, A.M. Namboodiri, J. Sivaswamy, C. Jawahar, Online handwriting recognition using depth sensors, in: Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2015, pp. 1061–1065.

[18] Y. Liu, X. Liu, Y. Jia, Hand-gesture based text input for wearable computers, in: Proceedings of the IEEE International Conference on Computer Vision Systems (ICVS'06), IEEE, 2006, pages 8, doi:10.1109/ICVS.2006.34.

[19] J. Hannuksela, S. Huttunen, P. Sangi, J. Heikkilä, Motion-based finger tracking for user interaction with mobile devices, in: Proceedings of the 4th European Conference on Visual Media Production, 2007.

[20] L. Jin, D. Yang, L.-X. Zhen, J.-C. Huang, A novel vision-based finger-writing character recognition system, J. Circuits, Syst., Comput. 16 (03) (2007) 421–436.

[21] S. Shah, A. Ahmed, I. Mahmood, K. Khurshid, Hand gesture based user interface for computer using a camera and projector, in: Proceedings of the IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 2011.

[22] H. Ishida, T. Takahashi, I. Ide, H. Murase, A hilbert warping method for handwriting gesture recognition, Pattern Recognit. 43 (8) (2010) 2799–2806.

[23] N.K. Iason Oikonomidis, A. Argyros, Efficient model-based 3d tracking of hand articulations using kinect, in: Proceedings of the BMVC, 2011.

[24] K. Hu, L. Yin, Multi-scale topological features for hand posture representation and analysis, in: Proceedings of the ICCV, 2013.

[25] D. Tang, T.-H. Yu, T.-K. Kim, Real-time articulated hand pose estimation using semi-supervised transductive regression forests, in: Proceedings of the ICCV, 2013.

[26] D. Tang, H.J. Chang, A. Tejani, T.-K. Kim, Latent regression forest: Structured estimation of 3d articulated hand posture, in: Proceedings of the CVPR, 2014.

[27] D. Zhang, G. Lu, Review of shape representation and description techniques, Pattern Recognit. 37 (1) (2004) 1–19.

[28] M. Yang, K. Kpalma, J. Ronsin, et al., A survey of shape feature extraction techniques, Pattern Recognit. (2008) 43–90. Published in November.

[29] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 24 (4) (2002) 509–522.

[30] E. Persoon, K.-S. Fu, Shape discrimination using Fourier descriptors, IEEE Trans. Syst., Man Cybern. 7 (3) (1977) 170–179.

[31] D. Zhang, G. Lu, A comparative study on shape retrieval using Fourier descriptors with different shape signatures, in: Proceeding of International Conference on Intelligent Multimedia and Distance Education (ICIMADE01), 2001, pp. 1–9.

[32] F.-S. Chen, C.-M. Fu, C.-L. Huang, Hand gesture recognition using a real-time tracking method and hidden Markov models, Image Vis. Comput. 21 (8) (2003) 745–758.

[33] S. Bourennane, C. Fossati, Comparison of shape descriptors for hand posture recognition in video, Signal, Image Video Process. 6 (1) (2012) 147–157.

[34] S. Conseil, S. Bourennane, L. Martin, Comparison of Fourier descriptors and HU moments for hand posture recognition, in: Proceedings of the European Signal Processing Conference (EUSIPCO), 2007.

[35] T. Lee, T. Hollerer, Handy ar: Markerless inspection of augmented reality objects using fingertip tracking, in: Proceedings of the 11th IEEE International Symposium on Wearable Computers, 2007.

[36] Z. Pan, Y. Li, M. Zhang, C. Sun, K. Guo, X. Tang, S. Zhou, A real-time multi-cue hand tracking algorithm based on computer vision, in: Proceedings of the IEEE Virtual Reality Conference (VR), 2010.

[37] M. Bhuyan, D. Raj Neog, M.K. Kar, Fingertip detection for hand pose recognition., Int. J. Comput. Sci. Eng. 4 (3) (2012) 501–511.

[38] H. Liang, J. Yuan, D. Thalmann, 3d fingertip and palm tracking in depth image sequences, in: Proceedings of the 20th ACM International Conference on Multimedia, ACM, 2012.

[39] J.L. Raheja, K. Das, A. Chaudhary, Fingertip detection: a fast method with natural hand, Int. J. Embed. Syst. Comput. Eng. 3 (2) (2012) 85–89.

[40] M. Maisto, M. Panella, L. Liparulo, A. Proietti, An accurate algorithm for the identification of fingertips using an RGB-d camera, IEEE J. Emerg. Sel. Topics Circuits Syst. 3 (2) (2013) 272–283.

[41] Y. Yu, Y. Song, Y. Zhang, Real time fingertip detection with kinect depth image sequences, in: Proceedings of the 22nd International Conference on Pattern Recognition (ICPR), 2014, IEEE, 2014, pp. 550–555.

[42] P. Krejov, R. Bowden, Multi-touchless: Real-time fingertip detection and tracking using geodesic maxima, in: Proceedings of the 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2013.

[43] A. McGovern, N. Hiers, M. Collier, D. Gagne, R. Brown, Spatiotemporal relational probability trees: An introduction, in: Proceedings of the Eighth IEEE International Conference on Data Mining (ICDM), 2008, pp. 935–940.

[44] T.A. Supinie, A. McGovern, J. Williams, J. Abernathy, Spatiotemporal relational random forests, in: Proceedings of the IEEE 12th International Conference on Data Mining Workshops, 2009, pp. 630–635.

[45] A. McGovern, D. John Gagne II, N. Troutman, R.A. Brown, J. Basara, J.K. Williams, Using spatiotemporal relational random forests to improve our understanding of severe weather processes, Stat. Anal. Data Min. 4 (4) (2011) 407–429.

[46] A. Yao, J. Gall, L.V. Gool, A hough transform-based voting framework for action recognition, in: Proceedings of the CVPR, 2010.

[47] J. Gall, V. Lempitsky, Class-specific hough forests for object detection, in: Proceedings of the CVPR, 2009.

[48] K. Mikolajczyk, H. Uemura, Action recognition with motion-appearance vocabulary forest, in: Proceedings of the CVPR, 2008.

[49] G. Yu, J. Yuan, Z. Liu, Action search by example using randomized visual vocabularies, IEEE Trans. Image Process. 22 (1) (2013) 377–390.

[50] Y. Jang, S.-T. Noh, H.J. Chang, T.-K. Kim, W. Woo, 3D finger CAPE: Clicking action and position estimation under self-occlusions in egocentric viewpoint, IEEE Trans. Vis. Comput. Gr. 21 (4) (2015) 501–510, doi:10.1109/TVCG.2015.2391860.

[51] J. Feldman, M. Singh, Information along contours and object boundaries., Psychol. Rev. 112 (1) (2005) 243.

[52] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[53] J.C. Léger, Menger curvature and rectifiability, Ann. Math 149 (3) (1999) 831–869.

[54] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell., 24 (5) (2002) 603–619.

[55] M. Elmezain, A. Al-Hamadi, J. Appenrodt, B. Michaelis, A hidden Markov model-based continuous gesture recognition system for hand motion trajectory, in: Proceedings of the 19th International Conference on Pattern Recognition (ICPR), 2008.

[56] Z. Yang, Y. Li, W. Chen, Y. Zheng, Dynamic hand gesture recognition using hidden Markov models, in: Proceedings of the 7th International Conference on Computer Science Education (ICCSE), 2012, pp. 360–365.