Feature Weighting in Dynamic Time Warping for Gesture Recognition in Depth Data

Miguel Reyes
Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona
Gran Via de les Corts Catalanes 585, 08007, Barcelona, Spain
Computer Vision Center
Campus UAB, Edifici O, 08193, Bellaterra, Barcelona, Spain
mreyese@gmail.com

Gabriel Domínguez
Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona
Gran Via de les Corts Catalanes 585, 08007, Barcelona, Spain
gabry.dominguez@gmail.com

Sergio Escalera
Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona
Gran Via de les Corts Catalanes 585, 08007, Barcelona, Spain
Computer Vision Center
Campus UAB, Edifici O, 08193, Bellaterra, Barcelona, Spain
sergio@maia.ub.es, http://www.maia.ub.es/∼sergio/

Abstract

We present a gesture recognition approach for depth video data based on a novel Feature Weighting approach within the Dynamic Time Warping framework. Depth features from human joints are compared through video sequences using Dynamic Time Warping, and weights are assigned to features based on inter-intra class gesture variability. Feature Weighting in Dynamic Time Warping is then applied for recognizing begin-end of gestures in data sequences. The obtained results recognizing several gestures in depth data show high performance compared with classical Dynamic Time Warping approach.

1. Introduction

Visual analysis of human motion is currently one of the most active research topics in Computer Vision. Several segmentation techniques for body pose recovery have been recently presented, allowing for better generalization of gesture recognition systems. The evaluation of human behavior patterns in different environments has been a problem studied in social and cognitive sciences, but now it is raised as a challenging approach to computer science due to the complexity of data extraction and its analysis.

In this work, we present a system for gesture recognition using depth data. From the point of view of data acquisition, many methodologies treat images captured by visible-light cameras. Computer Vision are then used to detect, describe, and learn visual features [4, 6]. The main difficulties of visual descriptors on RGB data is the discrimination of shapes, textures, background objects, changing in lighting conditions and viewpoint. On the other hand, depth information is invariant to color, texture and lighting objects, making it easier to differentiate between the background and the foreground object. The first systems for depth estimation were expensive and difficult to manage in practice. Earlier research used stereo cameras to estimate human poses or perform human tracking [10]. In the past few years, some research has focused on the use of time-of-flight range cameras (TOF) [5, 9, 13]. Nowadays, it has been published several works related to this topic because of the emergence of inexpensive structured light technology, reliable and robust to capture the depth information along with their corresponding synchronized RGB image. This technology has been developed by the PrimeSense [8] company and marketed by Microsoft XBox under the name

of Kinect. Using this sensor, Shotton et al. [11] present one of the greatest advances in the extraction of the human body pose from depth images, representing the body as a skeletal form comprised by a set of joints.

Once features are extracted from video data, the second step is the classification of gestures with the aim of describing human behavior. This step is extremely challenging because of the huge number of possible configurations of the human body that defines human motion. In our case, we base on the fifteen joints extracted by the approach of [11] as the set of features that will define the different gestures to recognize. A common approach for gesture recognition to model sequential data is Hidden Markov Model (HMM) [3], which is based on learning the transition probabilities among different human state configurations. Recently, there has been an emergent interest in Conditional Random Field (CRF) [12] for the learning of sequences. However, all these methods assume that we know the number of states for every motion. Other approaches make use of templates or global trajectories of motion [2], being highly dependent of the environment where the system is built. In order to avoid all these situations, our proposal is focused within the Dynamic Time Warping framework (DTW) [14]. Dynamic Time Warping allows to align two temporal sequences taking into account that sequences may vary in time based on the subject that performs the gesture. The alignment cost can be then used as a gesture appearance indicator.

The main contribution of this paper is the introduction of a new method based on DTW for gesture recognition using depth data. We propose a Feature Weighting approach within the DTW framework to improve gesture/action recognition. First, we estimate a temporal feature vector of subjects based on the 3D spatial coordinates of fifteen skeletal human joints. From a set of different ground truth behaviors of different length, DTW is used to compute the inter-class and intra-class gesture joint variability. These weights are used in the DTW cost function in order to improve gesture recognition performance. We test our approach on several human behavior sequences captured by the Kinect sensor. We show the robustness of the novel approach recognizing multiple gestures, identifying beginning and end of gestures in long term sequences, and showing performance improvements compared with classical DTW framework.

The rest of the paper is organized as follows: Section 2 reviews the human segmentation process through depth maps and the feature space representation. Section 3 presents the novel Feature Weighting approach within the DTW framework. Section 5 shows the experimental results, and finally, Section 6 concludes the paper.

2. Data Acquisition

This section describes the processing of depth data in order to perform the segmentation of the human body, obtaining its skeletal model, and computing its feature vector.

For the acquisition of depth maps we use the public API OpenNI software[1]. This middleware is able to provide sequences of images at a rate of 30 frames per second. The depth images obtained are 340×280 pixels resolution. These features are able to detect and track people to a maximum distance of six meters from multi-sensor device, as shown in Figure 1.

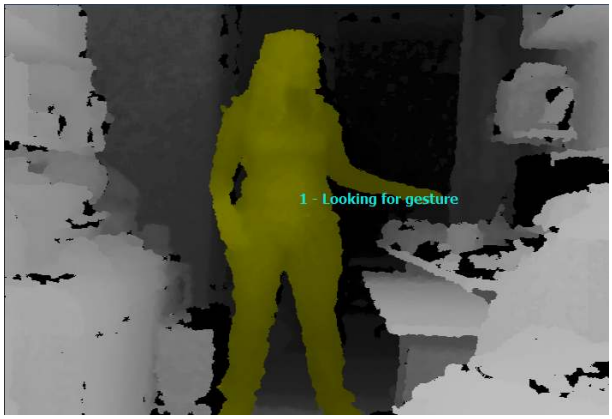

Figure 1. Human detection and tracking in uncontrolled environments.

We use the method of [11] to detect the human body and its skeletal model. The approach of [11] uses a huge set of human samples to infer pixel labels through Random Forest estimation, and skeletal model is defined as the centroid of mass of the different dense regions using mean shift algorithm. Experimental results demonstrated that it is efficient and effective for reconstructing 3D human body poses, even against partial occlusions, different points of view or no light conditions. The main problem of the skeletal representation is that it requires from a reference pose for initialization. In this sense, we perform an automatic calibration to fit human model and automatically obtain skeletal representation.

2.1. Automatic calibration

In order to define an automatic fitting of the human body without the need of detecting a specific pose for calibration, we defined a set of silhouettes associated to plausible normalized models of the skeletal form $\mathbf{C}$. These models make possible initialization of the skeletal model for a set of silhouettes. In order to find the model that best fits the subject of the scene we use a similarity function of structure θ between consecutive frames. This similarity function is based on the alignment of the consecutive skeletal joints to ob-

tain their respective Euclidean distance in two-dimensional space, using ζ as a similarity threshold value. Thus, the initialization of silhouette is given by,

$$\text{argmin}_i \theta(\mathbf{C}, \mathbf{C_i}), \quad \theta(\mathbf{C}, \mathbf{C_i}) < \zeta \tag{1}$$

being $\mathbf{C}$ the skeletal model at the current frame and $\mathbf{C_i}$ the i-th trained model.

2.2. Feature Vector Extraction

The articulated human model is defined by the set of 15 reference points shown in Figure 2. This model has the advantage of being highly deformable, and thus, able to fit to complex human poses.

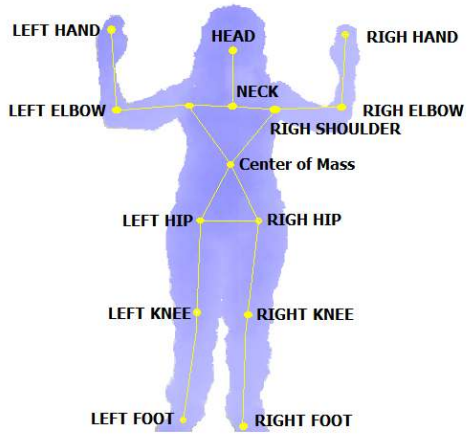

Figure 2. The 3D articulated human model consisting of 15 distinctive points.

In order to subsequently make comparisons and analyze the different extracted skeletal models, we need to normalize them. In this sense, we use the neck joint of the skeletal model as the origin or coordinates (OC). Then, the neck is not used in the frame descriptor, and the remaining 14 joints are using in the frame descriptor computing their 3D coordinates with respect to the OC. This transformation allows us to relate pose models that are at different depths, being invariant to translation, scale, and tolerant to corporal differences of subjects. Thus, the final feature vector $\mathbf{V_j}$ at frame j that defines the human pose is described by 42 elements (14 joints $\times$ three spatial coordinates),

$$V_j = \{\{v^1_{j,x}, v^1_{j,y}, v^1_{j,z}\}, ..., \{v^{14}_{j,x}, v^{14}_{j,y}, v^{14}_{j,z}\}\}$$

3. Feature Weighting in DTW

The original DTW algorithm [7] was defined to match temporal distortions between two models, finding an alignment warping path between the two time series $Q = \{q_1, .., q_n\}$ and $C = \{c_1, .., c_m\}$. In order to align these two sequences, a $M_{m\times n}$ matrix is designed, where the position (i, j) of the matrix contains the distance between c_i and q_j. The Euclidean distance is the most frequently applied. Then, a warping path,

$$W = \{w_1, .., w_T\}, \max(m, n) \leq T < m + n + 1$$

is defined as a set of "contiguous" matrix elements that defines a mapping between C and Q. This warping path is typically subjected to several constraints:

Boundary conditions: $w_1 = (1, 1)$ and $w_T = (m, n)$.

Continuity: Given $w_{t-1} = (a', b')$, then $w_t = (a, b)$, $a - a' \leq 1$ and $b - b' \leq 1$.

Monotonicity: Given $w_{t-1} = (a', b')$, $w_t = (a, b)$, $a - a' \leq 1$ and $b - b' \leq 1$, this forces the points in W to be monotically spaced in time.

We are generally interested in the final warping path that satisfying these conditions minimizes the warping cost,

$$DTW(Q, C) = \min \left\{ \frac{1}{T} \sqrt{\sum_{t=1}^{T} w_t} \right\} \tag{2}$$

where T compensates the different lengths of the warping paths. This path can be found very efficiently using dynamic programming to evaluate the following recurrence which defines the cumulative distance $\gamma(i, j)$ as the distance $d(i, j)$ found in the current cell and the minimum of the cumulative distance of the adjacent elements,

$$\gamma(i, j) = d(i, j) + \min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \tag{3}$$

Given the nature of our system to work in uncontrolled environments, we continuously review the stage for possible actions or gestures. In this case, our input feature vector Q is of "infinite" length, and may contain segments related to gesture C at any part.

Next, we describe our algorithm for begin-end of gesture recognition and the Feature Weighting proposal within the DTW framework.

3.1. Begin-end of gesture detection

In order to detect a begin-end of gesture $C = \{c_1, .., c_m\}$ in a maybe infinite sequence $Q = \{q_1, .., q_\infty\}$, a $M_{m\times\infty}$ matrix is designed, where the position (i, j) of the matrix contains the distance between c_i and q_j, quantifying its value by the Euclidean distance, as commented before. Finally, our warping path is defined by $W = \{w_1, .., w_\infty\}$ as in the standard DTW approach. Our aim is focused on finding segments of Q sufficiently similar to the sequence C. The system considers that there is correspondence between the current block k in Q and a gesture if satisfying the following condition,

$$M(m, k) < \mu, k \in [1, .., \infty]$$

for a given cost threshold μ. This threshold value is estimated in advance for each of the categories of actions or gestures using leave-one-out cross-validation strategy. This involves using a single observation from the original sample as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as the validation data. At each iteration, we evaluate the similarity value between the candidate and the rest of the training set. Finally we choose the threshold value which is associated with the largest number of hits within a category.

Once detected a possible end of pattern of gesture or action, the working path W can be found through backtracking of the minimum path from $M(m,k)$ to $M(0,z)$, being z the instant of time in Q where the gesture begins. The algorithm for begin-end of gesture detection for a particular gesture C in a large sequence Q using DTW is summarized in Table 1. Note that $d(i,j)$ is the cost function which measures the difference among our descriptors V_i and V_j. An example of a begin-end gesture recognition for a model and infinite sequence together with the working path estimation is shown in Figure 3.

Input: A gesture model $C = \{c_1, .., c_m\}$, its similarity threshold value μ, and the testing sequence $Q = \{q_1, .., q_\infty\}$. Cost matrix $M_{m\times\infty}$ is defined, where $N(w), w = (i,t)$ is the set of three upper-left neighbor locations of w in M.
Output: Working path W of the detected gesture, if any
// Initialization
for $i = 1 : m$ **do**
  for $j = 1 : \infty$ **do**
    $M(i,j) = \infty$
  end
end
for $j = 1 : \infty$ **do**
  $M(0,j) = 0$
end
for $t = 0 : \infty$ **do**
  for $i = 1 : m$ **do**
    $x = (i,t)$
    $M(w) = d(w) + \min_{w' \in N(w)} M(w')$
  end
  if $M(m,t) < \mu$ **then**
    $W = \{\text{argmin}_{w' \in N(w)} M(w')\}$
    return
  end
end

Table 1. DTW begin-end of gesture recognition algorithm.

3.2. Feature Weighting in DTW

In this section, we propose a Feature Weighting approach to improve the cost distance computation $d(w)$ of previous begin-end DTW algorithm.

In standard DTW algorithm, cost distances among feature vectors c_i and q_j (3D coordinates of the skeletal models in our case) are computed equally for each feature of the

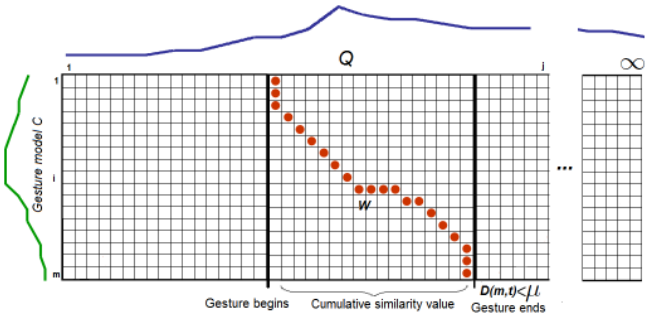

Figure 3. Begin-end of gesture recognition of a model C in an infinite sequence Q.

descriptors. However, it is intuitive that not all skeletal elements of the model participate equally for discriminating the performed gesture. For instance, the movement of the legs when performing hand shaking should not have influence, and thus, computing their deviation to a correspondence model of the gesture adds noise to the cost similarity function. In this sense, our proposal is based on associating a discriminatory weight to each joint of the skeletal model depending on its participation in a particular gesture. In order to automatically compute this weight per each joint, we propose an inter-intra gesture similarity algorithm.

First, we perform a weight training algorithm based on a ground truth data of gestures. Given the data composed by $\{n_1, .., n_N\}$ gesture categories described using skeletal descriptors, the objective is to obtain the inter-intra coefficient of the joints for the data set. This estimation is performed per each joint using a symmetric cost matrix $D_{N\times N}$. Each matrix element $D^p(i,j)$ for the matrix of joint p contains the mean DTW cost between all pairs of samples $C_i, C_j, \forall C_i \in n_i, \forall C_j \in n_j$ only considering the features of the descriptor related to the p-th joint, where n_i and n_j represent the set of samples for gesture categories i and j of the data set.

The mean DTW value at each position of the matrix D^p represents the variability of joint p between a pair of gestures. Note that the diagonal of D represents the intra-gesture variability per joint for all the gesture categories, meanwhile the rest of the elements compare the variability of joint p for two different gesture categories, codifying the inter-gesture variability. Since gestures, as any other object recognition system, will be more discriminative when increasing inter-distance and reducing intra-distance, a discriminative weight is defined as shown in Algorithm 2, which assigns high cost to joints high high intra-inter difference values and low cost otherwise. Moreover, the assigned weight is normalized in the same range to be comparable for all joints. Note that at the end of this procedure we have a final global weight vector $\nu = \{\nu^1, .., \nu^z\}$, with a weight value ν^p for the p-th joint, which is included

in the re-definition of the begin-end DTW algorithm cost function $d(w)$ to improve gesture recognition performance as follows,

$$d(c_i, c_j) = \sqrt{\sum_{p=1}^{|c_i|}((c_i^p - c_j^p) \cdot \nu^p)^2}, \quad (4)$$

where $|c_i|$ is the length of the feature vector c_i. The Feature Weighting algorithm for computing the weight vector $\nu = \{\nu^1, .., \nu^z\}$ is summarized in 2.

Input: Ground-truth data formed by N sets of gestures $\{n_1, .., n_N\}$.
Output: Weight vector $\nu = \{\nu^1, .., \nu^z\}$ associated with skeletal joints so that $\sum_{i=1}^{z} \nu^i = 1$.
$\nu = \emptyset$
for $p = 1 : z$ **do** // Number of joints
  for $i = 1 : N$ **do**
    for $j = i : N$ **do**
      $D^p(i,j) = \mathsf{mean}(DTW(C_v^i, C_w^j)), \forall v, w$ gesture samples of categories i and j.
    end
  end
  $\nu_{\mathsf{intra}} = \frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} D^p(i,j)}{\frac{N\times(N-1)}{2}}$// Computer intra-class variability
  $\nu_{\mathsf{inter}} = \frac{\mathsf{Trace}(D^p)}{m}$// Computer inter-class variability
  $\nu^p = max(0, \frac{\nu_{\mathsf{intra}} - \nu_{\mathsf{inter}}}{\nu_{\mathsf{intra}}})$// Compute global weight for joint p
  $\nu = \nu \cup \nu^p$
end
Normalize ν so that $\sum_{i=1}^{z} \nu^i = 1$

Table 2. Feature Weighting in DTW cost measure.

4. Results

Before the presentation of the results, first, we discuss the data, methods and parameters, and validation protocol of the experiments.

Data: We designed a new data set of gestures using the Kinect device consisting of five different categories: jumping, bendding, clapping, greeting, and noting with the hand[1]. It has been considered 10 different actors, 10 different backgrounds, and 100 sequences per subject for recording the data set. Thus, the data set contains the high variability from uncontrolled environments. The resolution of the video depth sequences is 340×280 at 30 FPS. The data set contains a total of 1000 gesture samples considering all the categories. The ground-truth of each sequence is performed manually by examining and noting the position in the video when some actor begin-ends a gesture. Some samples of the captured gestures for different categories are shown in Figure 4.

[1]The data set is public upon to request to the authors of the paper.

Methods and parameters: For the implementation of the system we used C/C++, efficiently using dynamic programming to evaluate the recurrence which defines the cumulative distance between vectors of features on each frame. The people detection system used is provided by the public library OpenNI. This library has a high accuracy in people detection, allowing multiple detection even in cases of partial occlusions. The detection is accurate as people remain at a minimum of 60cm from the camera and up to 4m, but can reach up to 6m but with less robust and reliable detection. For automatically initialization of the system we have used 20 calibration poses. These calibration models are also obtained through the library OpenNI. This calibration set has been built with high variability in order to automatically obtain the feature vector in different human pose configurations. During the calibration process we used a structural coherence function θ from the fifth consecutive frame. This assures stabilization to obtain a reliable ζ for a better fit of the skeletal model. In Figure 5 we show different initialization models.

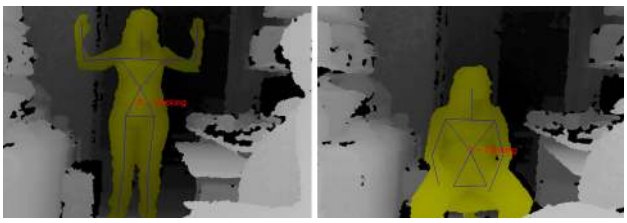

Figure 5. Different calibration poses.

Validation protocol: For the validation our approach and classical DTW algorithm, we compute the Feature Weighting vector ν and gesture cost threshold μ over a leave-one-out validation. The validation sequences may have different length size since they can be aligned using DTW algorithm and trained for the different estimated values of μ. We validate the begin-end of gesture DTW approach and compare with the Feature Weighting methodology within the same framework. As a validation measurement we compute the confusion matrix for each test sample of the leave-out-out strategy. This methodology allows us to perform an exhaustive analysis of the methods and data set. Adding all test confusion matrices in a performance matrix C_m, final accuracy A is computed using the following formula,

$$A = 100 \cdot \frac{\mathsf{Trace}(C_m)}{NC + \sum_{i=1}^{m}\sum_{j=1}^{m} C_m(i,j)} \quad (5)$$

Where NC contains the number of samples of the data set that has not been classified by any gesture since the

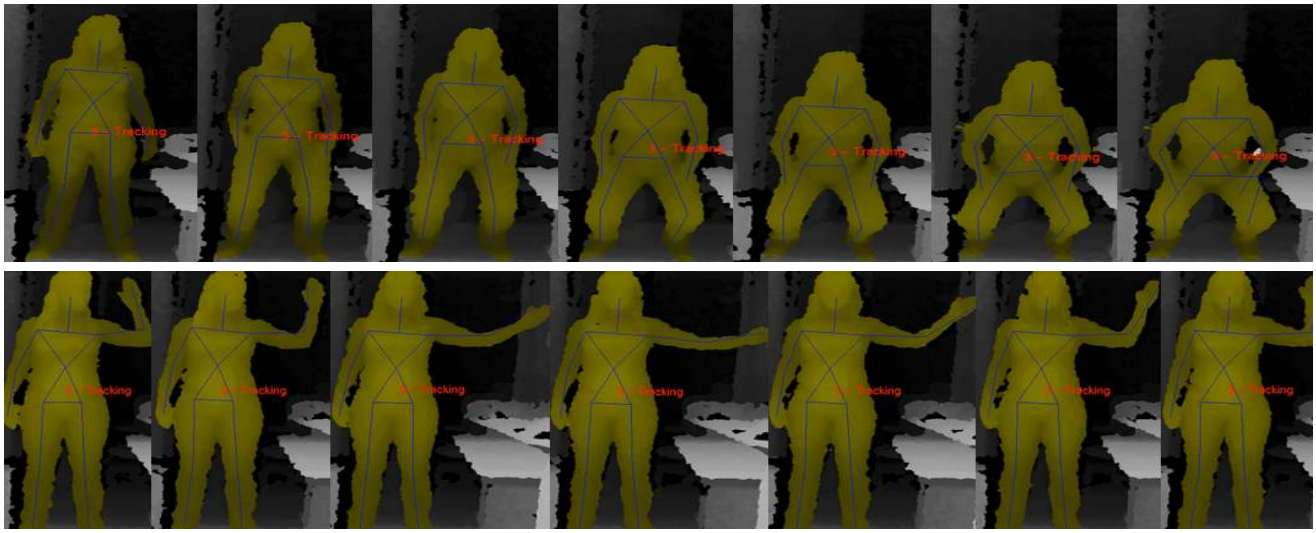

Figure 4. Samples of gestures for different categories of the data set.

Classification Results Feature Weighting DTW		
Gesture	Begin-end DTW	Feature Weighting
Jump	**68**	**68**
Bend	63.4	**68**
Clap	42	**55**
Greet	64.2	**73**
Note	68	**76**

Table 3. Classification performance A over the gesture data set for the five gesture categories using DTW begin-end approach and including the Feature Weighting methodology.

classification threshold μ has not been satisfied. This evaluation is pessimistic and realistic since both a sample which is not classified or is classified more than once penalizes the final evaluation measurement.

The obtained results applying DTW begin-end gesture recognition and including the Feature Weighting approach on the new data set are shown in Table 3. The results show the final performance per gesture over the whole data set using both classification strategies. The best performance per category is marked in bold. Note that for all gesture categories, the begin-end DTW technique with Feature Weighting improves the accuracy of standard DTW. Only in the case of the jump category the performance is maintained. An example of gesture recognition in a sequence of the data set is shown in Figure 6.

5. Conclusion

In this paper, we proposed a fully-automatic general framework for real time action/gesture recognition in uncontrolled environments using depth data. The system analyzes data sequences based on the assignment of weights to gesture descriptors so that DTW cost measure improves discrimination. The feature vectors are extracted automatically through a calibration set, obtaining 3D coordinates of skeletal models with respect an origin of coordinates, making description invariant to translation, scale, and tolerant to corporal differences among subjects. The final gesture is recognized by means of a novel Feature Weighting approach, which enhance recognition performance based on the analysis on inter-intra class variability of vector features among gesture descriptors. The evaluation of the method has been performed on a novel depth data set of gestures, automatically detecting begin-end of gesture and obtaining performance improvements compared to classical DTW algorithm.

Acknowledgments

This work has been supported in part by the projects TIN2009-14404-C02 and CONSOLIDER-INGENIO CSD 2007-00018.

References

[1] Open natural interface. November 2010. Last viewed 14-07-2011 13:00.

[2] C. S. Chan, H. Liu, and D. J. Brown. Recognition of human motion from qualitative normalised templates. *J. Intell. Robotics Syst.*, 48:79–95, January 2007.

[3] A. W. T. S. D. Gehrig, H. Kuehne. Hmm-based human motion recognition with optical flow data. In *IEEE International Conference on Humanoid Robots (Humanoids 2009)*, Paris, France, 2009.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 1:886–893, 2005.

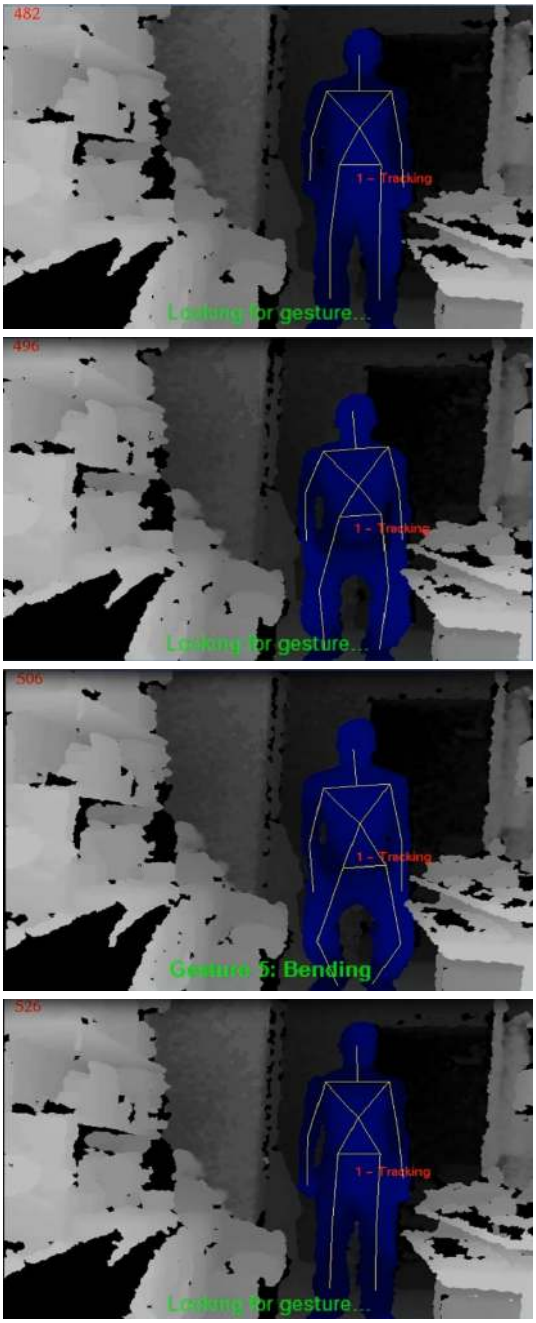

Figure 6. Gesture recognition example in a sequence of the data set.

[5] H. Jain and A. Subramanian. Real-time upper-body human pose estimation using a depth camera. *HP Technical Reports*, 1(190), 2010.

[6] E. N. Mortensen, H. Deng, and L. Shapiro. A sift descriptor with global context. *CVPR*, 1:184–190 vol. 1, 2005.

[7] M. Parizeau and R. Plamondon. A comparative analysis of regional correlation, dynamic time warping, and skeletal tree matching for signature verification.

[8] PrimeSense Inc. *Prime Sensor NITE 1.3 Algorithms notes*, 2010. Last viewed 14-07-2011 13:19.

[9] J. Rodgers, D. Anguelov, P. Hoi-Cheung, and K. D. Object pose detection in range scan data. *CVPR*, pages 2445–2452, 2006.

[10] B. Sabata, F. Arman, and J. Aggarwal. Segmentation of 3d range images using pyramidal data structures,. *CVGIP: Image Understanding*, 57(3):373–387, 1993.

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. *CVPR*, 2011.

[12] C. Sminchisescu, A. Kanaujia, and D. Metaxas. Conditional models for contextual human motion recognition. *CVIU*, 104(2-3):210–220, 2006.

[13] V. V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun. Real time motion capture using a single time-of-flight camera. *CVPR*, pages 755–762, 2010.

[14] F. Zhou, F. De la Torre, and J. K. Hodgins. Aligned cluster analysis for temporal segmentation of human motion. In *IEEE Conference on Automatic Face and Gestures Recognition (FG)*, September 2008.