Probability-based Dynamic Time Warping and Bag-of-Visual-and-Depth-Words for Human Gesture Recognition in RGB-D

Antonio Hernández-Vela[a,b], Miguel Ángel Bautista[a,b], Xavier Perez-Sala[b,c,d], Víctor Ponce-López[a,b,e], Sergio Escalera[a,b], Xavier Baró[b,e], Oriol Pujol[a,b], Cecilio Angulo[d]

[a]*Dept. MAIA, Universitat de Barcelona, Gran Via 585, 08007 Barcelona, Spain.*
[b]*Computer Vision Center, Campus UAB, Edifici O, 08193 Bellaterra, Barcelona, Spain.*
[c]*Fundació Privada Sant Antoni Abat, Rambla de l'Exposició, 59-69, 08800 Vilanova i la Geltrú, Spain.*
[d]*UPC - BarcelonaTECH, Av. Víctor Balaguer 1, 08800 Vilanova i la Geltrú, Spain.*
[e]*EIMT/IN3, Universitat Oberta de Catalunya, Rbla. del Poblenou 156, 08018 Barcelona, Spain.*

Abstract

We present a methodology to address the problem of human gesture segmentation and recognition in video and depth image sequences. A Bag-of-Visual-and-Depth-Words (BoVDW) model is introduced as an extension of the Bag-of-Visual-Words (BoVW) model. State-of-the-art RGB and depth features, including a newly proposed depth descriptor, are analysed and combined in a late fusion form. The method is integrated in a Human Gesture Recognition pipeline, together with a novel Probability-based Dynamic Time Warping (PDTW) algorithm which is used to perform prior segmentation of idle gestures. The proposed DTW variant uses samples of the same gesture category to build a Gaussian Mixture Model driven probabilistic model of that gesture class. Results of the whole Human Gesture Recognition pipeline in a public data set show better performance in comparison to both standard

BoVW model and DTW approach.

Keywords: RGB-D, Bag-of-Words, Dynamic Time Warping, Human Gesture Recognition

1. Introduction

Nowadays, human gesture recognition is one of the most challenging tasks in computer vision. Current methodologies have shown preliminary results on very simple scenarios, but they are still far from human performance. Due to the large number of potential applications involving human gesture recognition in fields like surveillance [1], sign language recognition [2], or clinical assistance [3] among others, there is a large and active research community devoted to deal with this problem. Independently of the application field, the usual human gesture recognition pipeline is mainly formed by two steps: *gesture representation* and *gesture classification*.

Regarding the gesture representation step, literature shows a variety of methods that have obtained successful results. Commonly applied in image retrieval or image classification scenarios, *Bag-of-Visual-Words* (BoVW) is one of the most used approaches. This methodology is an evolution of *Bag-of-Words* (BoW) [4] representation, used in document analysis, where each document is represented using the frequency of appearance of each word in a dictionary. In the image domain, these words become visual elements of a certain visual vocabulary. First, each image is decomposed into a large set of patches, either using some type of spatial sampling (grids, sliding window, etc.) or detecting points with relevant properties (corners, salient regions, etc.). Each patch is then described obtaining a numeric descriptor. A set of

V representative visual words are selected by means of a clustering process over the descriptors. Once the visual vocabulary is defined, each new image can be represented by a global histogram containing the frequencies of visual words. Finally, this histogram can be used as input for any classification technique (i.e. $k-$Nearest Neighbor or SVM) [5, 6]. In addition, extensions of BoW from still images to image sequences have been recently proposed in the context of human action recognition, defining Spatio-Temporal-Visual-Words (STVW) [7].

The release of the Microsoft Kinect^{TM} sensor in late 2010 has allowed an easy and inexpensive access to almost synchronized range imaging with standard video data. Those data combine both sources into what is commonly named RGB-D images (RGB plus Depth). This data fusion has reduced the burden of the first steps in many pipelines devoted to image or object segmentation, and opened new questions such as how these data can be effectively described and fused. Motivated by the information provided by depth maps, several 3-D descriptors have been recently developed [8, 9] (most of them based on codifying the distribution of normal vectors among regions in the 3D space), as well as their fusion with RGB data [10] and learning approaches for object recognition [11]. This depth information has been particularly exploited for gesture recognition and human body segmentation and tracking. While some works focus on just the hand regions for performing gesture recognition [12, 13, 14, 15, 16, 17], in [18] Shotton introduced one of the greatest advances in the extraction of the human body pose using RGB-D, which is provided as part of the Kinect^{TM} human recognition framework. The method is based on inferring pixel label probabilities through

Random Forest from learned offsets of depth features. Then, mean shift is applied to estimate human joints and representing the body in skeletal form. Hernández-Vela et al. [19] extended Shotton's work applying Graph-cuts to the pixel label probabilities obtained through Random Forest, in order to compute consistent segmentations in the spatio-temporal domain. Girshick and Shotton [20] proposed later a different approach in which they directly regress the positions of the body joints, without the need of an intermediate pixel-wise body limb classification as in [18]. The extraction of body pose information opens the door to one of the most challenging problems nowadays, i.e. human gesture recognition.

In the gesture classification step there exists a wide number of methods based on dynamic programming algorithms for both alignment and clustering of temporal series [21]. Other probabilistic methods such as Hidden Markov Models (HMM) or Conditional Random Fields (CRF) have been commonly used in the literature [2]. Nevertheless, one of the most common methods for Human Gesture Recognition is Dynamic Time Warping (DTW) [22], since it offers a simple yet effective temporal alignment between sequences of different lengths. However, the application of such methods to gesture detection in complex scenarios becomes a hard task due to the high variability of the environmental conditions among different domains. Some common problems are: wide range of human pose configurations, influence of background, continuity of human movements, spontaneity of human actions, speed, appearance of unexpected objects, illumination changes, partial occlusions, or different points of view, just to mention a few. These effects can cause dramatic changes in the description of a certain gesture, generating a great intra-class variability. In this sense, since usual DTW is applied between a sequence and a single pattern, it fails when taking into account such variability.

The problem of gesture recognition in which an idle or reference gesture is performed between gestures is addressed in this paper. In order to solve this problem, we introduce a continuous human gesture recognition pipeline based on: First, a new feature representation by means of a Bag-of-Visual-and-Depth-Words (BoVDW) approach that takes profit of multi-modal RGB-D data to tackle the gesture representation step. The BoVDW is empowered by the combination of both RGB images and a new depth descriptor which takes into account the distribution of normal vectors with respect to the camera position, as well as the rotation with respect to the roll axis of the camera. Next, we propose the definition of an extension of DTW method to a probability-based framework in order to perform temporal gesture segmentation. In order to evaluate the presented approach, we compare the performances achieved with state-of-the-art RGB and depth feature descriptors separately, and combine them in a late fusion form. All these experiments are performed in the proposed framework using the public data set provided by the ChaLearn Gesture Challenge[1]. Results of the proposed BoVDW method show better performance using late fusion in comparison to early fusion and standard BoVW model. Moreover, our BoVDW approach outperforms the baseline methodology provided by the ChaLearn Gesture Recognition Challenge 2012. In the same way, the results obtained with the proposed PDTW outperform the ones from the classical DTW approach.

[1] `http://gesture.chalearn.org/`

The BoVDW model for gesture recognition is introduced in Section 2, as well as the PDTW. Experimental results and their analysis are presented in Section 3. Finally, Section 4 concludes the paper.

2. BoVDW and Probability-based DTW for Human Gesture Recognition

As pointed out in the Introduction, we address the problem of gesture recognition, with the constraint that an idle or reference gesture is performed between gestures. The main reason for such constraint is that in many real-world settings there always exists an idle gesture between movements rather than a continuous flux of gestures. Some examples are sports like tennis, swordplay, boxing, martial arts, or choreographic sports. However, the existence of an idle gesture is not only related to sports, some other daily tasks like cooking or dancing contain idle gestures in certain situations. Moreover, the proposed system can be extended to be applied to other gesture recognition domains without the need of modelling idle gestures, but any other kind of gesture categories.

In this sense, our approach consists of two steps: *a temporal gesture segmentation* step (the detection of the idle gesture), and *the gesture classification* step. The former one aims to provide a temporal segmentation of gestures. To perform such temporal segmentation, a novel probabalistic-based DTW models the variability of the idle gesture by learning a GMM on the features of the idle gesture category. Once the gestures have been segmented, the latter step is gesture classification. Segmented gestures are represented and classified by means of a BoVDW method, which integrates

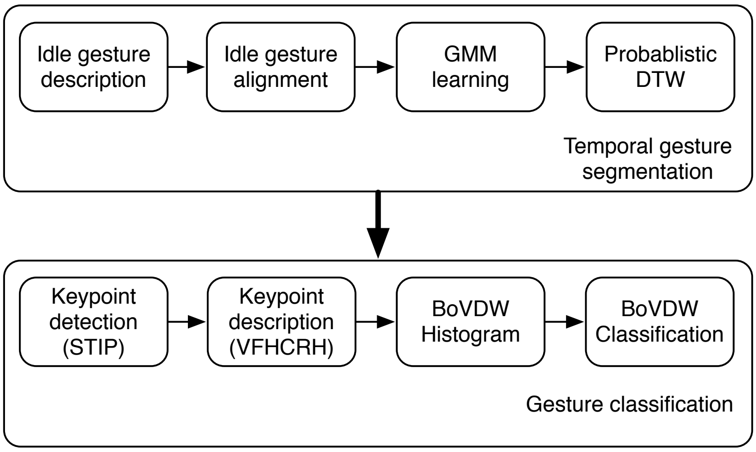


Figure 1: General pipeline of the proposed approach.

in a late fusion form the information of both RGB and Depth images.

The global pipeline of the approach is depicted in Figure 1. The proposal is divided in two blocks, the temporal gesture segmentation step and the gesture classification step, which are detailed in next sections.

2.1. Gesture Segmentation: Probability-based DTW

The original DTW is introduced in this section, as well as its common extension to detect a certain sequence given an indefinite data stream. In the following subsections, DTW is extended in order to align patterns taking into account the probability density function (PDF) of each element of the sequence by means of a Gaussian Mixture Model (GMM). A flowchart of the whole methodology is shown in Figure 2.

2.1.1. Dynamic Time Warping

The original DTW algorithm was defined to match temporal distortions between two models, finding an alignment/warping path between two time

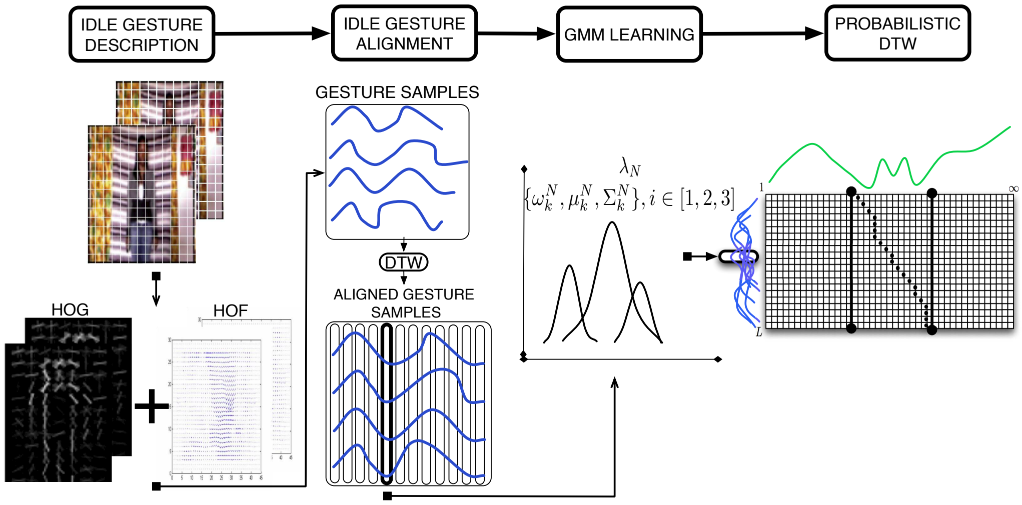


Figure 2: Flowchart of the Probabilistic DTW gesture segmentation methodology.

series: an input model $Q = \{q_1, .., q_n\}$ and a certain sequence $C = \{c_1, .., c_m\}$. In our particular case, the time series Q and C are video sequences, where each q_j and c_i will be feature vectors describing the $j-$th and $i-$th frame respectively. In this sense, Q will be an input video sequence and C will be the gesture we are aiming to detect. Generally, in order to align these two sequences, a $M_{m \times n}$ matrix is designed, where position (i, j) of the matrix contains the alignment cost between c_i and q_j. Then, a warping path of length τ is defined as a set of contiguous matrix elements, defining a mapping between C and Q: $W = \{w_1, .., w_\tau\}$, where w_i indexes a position in the cost matrix M. This warping path is typically subject to several constraints,

Boundary conditions: $w_1 = (1, 1)$ and $w_\tau = (m, n)$.

Continuity and monotonicity: Given $w_{\tau'-1} = (a', b')$, $w_{\tau'} = (a, b)$, then $a - a' \leq 1$ and $b - b' \leq 1$. This condition forces the points in the cost matrix with the warping path W to be monotonically spaced in time.

Interest is focused on the final warping path that, satisfying these conditions, minimizes the warping cost,

$$DTW(M) = \min_{W} \left\{ \frac{M(w_\tau)}{\tau} \right\}, \tag{1}$$

where τ compensates the different lengths of the warping paths at each time t. This path can be found very efficiently using dynamic programming. The cost at a certain position $M(i,j)$ can be found as the composition of the Euclidean distance $d(i,j)$ between the feature vectors c_i and q_j of the two time series, and the minimum cost of the adjacent elements of the cost matrix up to that position, as,

$$M(i,j) = d(i,j) + \min\{M(i-1,j-1), M(i-1,j), M(i,j-1)\}. \tag{2}$$

However, given the streaming nature of our problem, the input video sequence Q has no definite length (it may be an infinite video sequence) and may contain several occurrences of the gesture sequence C. In this sense, the system considers that there is correspondence between the current block k in Q and the gesture when the following condition is satisfied, $M(m,k) < \theta, k \in [1,..,\infty]$ for a given cost threshold θ. At this point, if $M(m,k) < \theta$ k is consider a possible end of a gesture sequence C.

Once detected a possible end of the gesture sequence, the warping path W can be found through backtracking the minimum cost path from $M(m,k)$ to $M(0,g)$, being g the instant of time in Q where the detected gesture begins. Note that $d(i,j)$ is the cost function which measures the difference among descriptors c_i and q_j, which in standard DTW is defined as the euclidean distance between c_i and q_j. An example of a begin-end gesture recognition together with the warping path estimation is shown in Figure 2 (last 2 steps: GMM learning and Probabilistic DTW).

2.1.2. Handling variance with Probability-based DTW

Consider a training set of N sequences, $S = \{S_1, S_2, \ldots, S_N\}$, that is, N gesture samples belonging to the same gesture category. Then, each sequence $S_g = \{s_1^g, \ldots, s_{L_g}^g\}$, (each gesture sample) is composed by a feature vector [2] for each frame t, denoted as s_t^g, where L_g is the length in frames of sequence S_g. In order to avoid temporal deformations of the gesture samples in S, all sequences are aligned with the median length sequence using the classical DTW with Euclidean distance. Let us assume that sequences are ordered according to their length, so that $L_{g-1} \leq L_g \leq L_{g+1}, \forall g \in [2, .., N-1]$, then, the median length sequence is $\bar{S} = S_{\lceil \frac{N}{2} \rceil}$.

It is worth noting that this alignment step by using DTW has no relation to the actual gesture recognition, as it is consider a pre-processing step to obtain a set of gesture samples with few temporal deformations and a matching length.

Finally, after this alignment process, all sequences have length $L_{\lceil \frac{N}{2} \rceil}$. The set of warped sequences is defined as $\tilde{S} = \{\tilde{S}_1, \tilde{S}_2, \ldots, \tilde{S}_N\}$ (See Figure 3(b)). Once all samples are aligned, the N feature vectors corresponding to each sequence element at a certain frame t, denoted as $\widetilde{F}_t = \{f_t^1, f_t^2, ..., f_t^N\}$, are modelled by means of a $G-$component Gaussian Mixture Model (GMM) $\lambda_t = \{\alpha_k^t, \mu_k^t, \Sigma_k^t\}, \quad k = 1, \ldots, G$, where α_k^t is the mixing value, and μ_k^t and Σ_k^t are the parameters of each of the G Gaussian models in the mixture. As

[2] HOG/HOF descriptors in our particular case, see Sec. 3.2.1 for further details.

a result, each one of the GMMs that model each $\widetilde{F}_t$ is defined as follows,

$$p(\widetilde{F}_t) = \sum_{k=1}^{G} \alpha_k^t \cdot e^{-\frac{1}{2}(x-\mu_k^t)^T \cdot (\Sigma_k^t)^{-1} \cdot (x-\mu_k^t)}. \tag{3}$$

The resulting model is composed by the set of GMMs that model each set $\widetilde{F}_t$ among all warped sequences of a certain gesture class. An example of the process is shown in Figure 3.

2.1.3. Distance measures

In the classical DTW, a pattern and a sequence are aligned using a distance metric, such as the Euclidean distance. However, since our gesture samples are modelled by means of probabilistic models, in order to use the principles of DTW, the distance must be redefined. In thise sense, a soft-distance based on the probability of a point x belonging to each one of the G components in the GMM is consider, i.e. the posterior probability of x is obtained according to Eq. (3). Therefore, since $\sum_{k=1}^{G} \alpha_k^t = 1$, the probability of a element $q_j \in Q$ belonging to the whole GMM λ_t can be computed as,

$$P(q_j, \lambda_t) = \sum_{k=1}^{G} \alpha_k^t \cdot P(q_j)_k, \tag{4}$$

$$P(q_j)_k = e^{-\frac{1}{2}(q_j-\mu_k^t)^T \cdot (\Sigma_k^t)^{-1} \cdot (q_j-\mu_k^t)}, \tag{5}$$

which is the sum of the weighted probability of each component. Nevertheless, an additional step is required since the standard DTW algorithm is conceived for distances instead of similarity measures. In this sense, a soft-distance based measure of the probability is used, which is defined as,

$$D(q_j, \lambda_t) = \exp^{-P(q_j, \lambda_t)}. \tag{6}$$

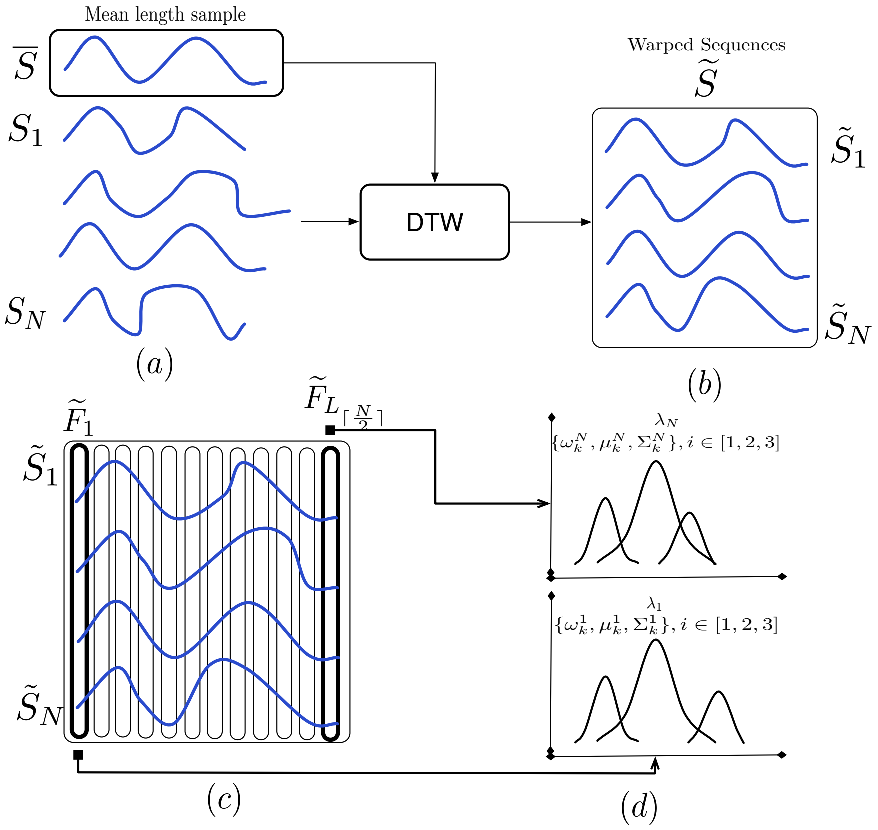


Figure 3: (a) Different sequences of a certain gesture category and the median length sequence. (b) Alignment of all sequences with the median length sequence by means of Euclidean DTW. (c) Warped sequences set $\tilde{S}$ from which each set of t-th elements among all sequences are modelled. (d) Gaussian Mixture Model learning with 3 components.

In conclusion, possible temporal deformations of different samples of the same gesture category are taken into account by aligning the set of N gesture samples with the median length sequence. In addition, by modelling with a GMM each set of feature vectors which compose the resulting warped sequences, we obtain a methodology for gesture detection that is able to deal with multiple deformations in gestures both temporal (which are modelled by the DTW alignment), or descriptive (which are learned by the GMM modelling). The algorithm that summarizes the use of the probability-based DTW to detect start-end of gesture categories is shown in Table 1. Figure 6 illustrates the application of the algorithm in a toy problem.

Table 1: Probability-based DTW algorithm.

Input: A set of GMM models $\lambda = \{\lambda_1, .., \lambda_m\}$ corresponding to a gesture category, a threshold value μ, and the streaming sequence $Q = \{q_1, .., q_\infty\}$. Cost matrix $M_{m \times \infty}$ is defined, where $\mathcal{N}(x), x = (i, t)$ is the set of three upper-left neighbor locations of x in M.

Output: Warping 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 $j = 0 : \infty$ **do**
    for $i = 1 : m$ **do**
        $x = (i, j)$
        $M(x) = D(q_j, \lambda_i) + \min_{x' \in \mathcal{N}(x)} M(x')$
    end
    if $M(m, j) < \mu$ **then**
        $W = \{\operatorname{argmin}_{x' \in \mathcal{N}(x)} M(x')\}$
        return
    end
end

2.2. Gesture Representation: BoVDW

In this section, the BoVDW approach for Human Gesture Representation is introduced. Figure 4 contains a conceptual scheme of the approach. In this figure, it is shown that the information from RGB and Depth images is merged, while circles representing the spatio-temporal interest points are described by means of the proposed novel VFHCRH descriptor.

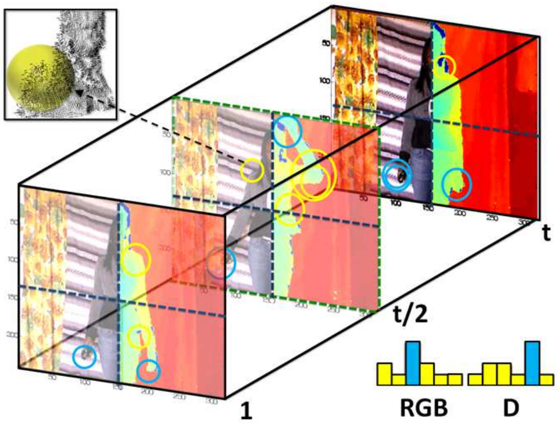


Figure 4: BoVDW approach in a Human Gesture Recognition scenario. Interest points in RGB and depth images are depicted as circles. Circles indicate the assignment to a visual word in the shown histogram – computed over one spatio-temporal bin. Limits of the bins from the spatio-temporal pyramids decomposition are represented by dashed lines in blue and green, respectively. A detailed view of the normals of the depth image is shown in the upper-left corner.

2.2.1. Keypoint detection

The first step of BoW-based models consists of selecting a set of points in the image/video with relevant properties. In order to reduce the amount of points in a dense spatio-temporal sampling, the Spatio-Temporal Interest Point (STIP) detector [23] is used, which is an extension of the well-known Harris detector in the temporal dimension. The STIP detector firstly computes the second-moment $3{\times}3$ matrix η of first order spatial and temporal derivatives. Finally, the detector searches regions in the image with significant eigenvalues λ_1,λ_2,λ_3 of η, combining the determinant and the trace of η,

$$H = |\eta| - K \cdot T_r(\eta)^3, \tag{7}$$

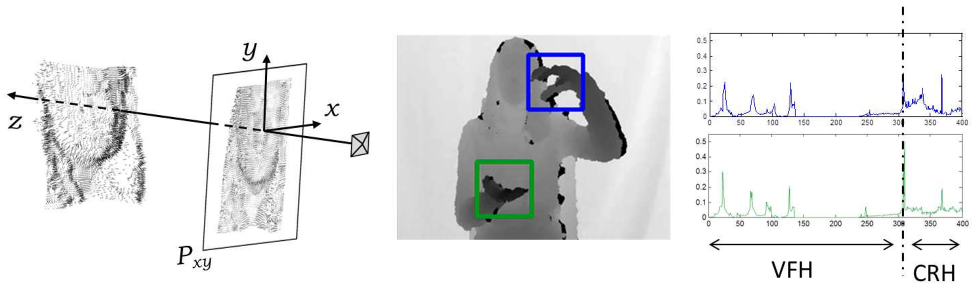
where $|.|$ corresponds to the determinant, $T_r(.)$ computes the trace, and K stands for a relative importance constant factor. As multi-modal RGB-D data is employed, the STIP detector is applied separately on the RGB and Depth volumes, so two sets of interest points S_{RGB} and S_D are obtained.

2.2.2. Keypoint description

In this step, the interest points detected in the previous step should be described. On one hand, state-of-the-art RGB descriptors are computed for S_{RGB}, including Histogram of Gradients (HOG) [24], Histogram of Optical Flow (HOF), and their concatenation HOG/HOF [25]. On the other hand, a new descriptor VFHCRH (Viewpoint Feature Histogram Camera Roll Histogram) is introduced for S_D, as detailed below.

2.2.3. VFHCRH

The recently proposed Point Feature Histogram (PFH) and Fast Point Feature Histogram (FPFH) descriptors [8] represent each instance in the 3-D cloud of points with a histogram encoding the distribution of the mean curvature around it. Both PFH and FPFH provide $\mathcal{P}$6 DOF (Degrees of Freedom) pose invariant histograms, being $\mathcal{P}$ the number of points in the cloud. Following their principles, Viewpoint Feature Histogram (VFH)[9] describes each cloud of points with one descriptor of 308 bins, variant to object rotation around pitch and yaw axis. However, VFH is invariant to



(a) (b)

Figure 5: (a) Point cloud of a face and the projection of its normal vectors onto the plane P_{xy}, orthogonal to the viewing axis z. (b) VFHCRH descriptor: Concatenation of VFH and CRH histograms resulting in 400 total bins

rotation about the roll axis of the camera. In contrast, Clustered Viewpoint Feature Histogram (CVFH) [26] describes each cloud of points using a different number of descriptors r, where r is the number of stable regions found on the cloud. Each stable region is described using a non-normalized VFH histogram and a Camera's Roll Histogram (CRH), and the final object description includes all region descriptors. CRH is computed by projecting the normal of the point cloud $\tau^{(i)}$ of the i-th point $\rho^{(i)}$ onto a plane P_{xy} that is orthogonal to the viewing axis z, the vector between the camera center and the centroid of the cloud, under orthographic projection,

$$\tau_{xy}^{(i)} = ||\tau^{(i)}|| \cdot \sin(\phi), \tag{8}$$

where ϕ is the angle between the normal $\tau^{(i)}$ and the viewing axis. Finally, the histogram encodes the frequencies of the projected angle ψ between $\tau_{xy}^{(i)}$ and y-axis, the vertical vector of the camera plane (see Fig. 5(a)).

In order to avoid descriptors of arbitrary lengths for different point clouds, the whole cloud is described using VFH. In addition, a 92 bins CRH is computed for encoding $6DOF$ information. The concatenation of both histograms results in the proposed VFHCRH descriptor of 400 bins shown in Figure 5(b). Note how the first 308 bins of the concatenated feature vector correspond to the VFH, that encode the normals of the point cloud. Finally, the remaining bins corresponding to the CRH descriptor, encode the information of the relative orientation of the point cloud to the camera.

2.2.4. BoVDW histogram

Once all the detected points have been described, the vocabulary of V visual/depth words is designed by applying a clustering method over all the descriptors. Hence, the clustering method –k-means in our case– defines the words from which a query video sequence will be represented, shaped like a histogram h that counts the occurrences of each word. Additionally, in order to introduce geometrical and temporal information, spatio-temporal pyramids are applied. Basically, spatio-temporal pyramids consist of dividing the video volume in b_u, b_v, and b_p bins along the u, v, and p dimensions of the volume, respectively. Then, $b_u \times b_v \times b_p$ separate histograms are computed with the points lying in each one of these bins, and they are concatenated jointly with the general histogram computed using all points.

These histograms define the model for a certain class of the problem –in our case, a certain gesture. Since multi-modal data is considered, different vocabularies are defined for the RGB-based descriptors and the depth-based ones, and the corresponding histograms, h^{RGB} and h^{D}, are obtained. Finally, the information given by the different modalities is merged in the next and final classification step, hence using *late fusion*.

2.2.5. BoVDW-based classification

The final step of the BoVDW approach consists of predicting the class of the query video. For that, any kind of multi-class supervised learning technique could be used. In our case, a simple k-Nearest Neighbour classification is used, computing the complementary of the histogram intersection as a distance,

$$d^F = 1 - \sum_i \min(h^F_{model}(i), h^F_{query}(i)), \tag{9}$$

where $F \in \{RGB, D\}$. Finally, in order to merge the histograms h^{RGB} and h^D, the distances d^{RGB} and d^D are computed separately, as well as the weighted sum,

$$d_{hist} = (1-\beta)d^{RGB} + \beta d^D, \tag{10}$$

to perform late fusion, where β is a weighting factor.

3. Experiments and Results

To better understand the experiments, firstly the data, methods, and evaluation measurements are discussed.

3.1. Data

Data source used is the ChaLearn [27] data set, provided by the CVPR2011 Workshop's challenge on Human Gesture Recognition. The data set consists of 50,000 gestures each one portraying a single user in front of a fixed camera. The images are captured by the Kinect device providing both RGB and depth images. A subset of the whole data set has been considered, formed by 20 development batches with a manually tagged gesture segmentation, which is used to obtain the idle gestures. Each batch includes 100 recorded gestures grouped in sequences of 1 to 5 gestures performed by the same user. The gestures from each batch are drawn from a different lexicon of 8 to 15 unique gestures and just one training sample per gesture is provided. These lexicons are categorized in nine classes, including: (1) body language gestures (scratching your head, crossing your arms, etc.), (2) gesticulations performed to accompany speech, (3) illustrators (like Italian gestures), (4) emblems (like Indian Mudras), (5) signs (from sign languages for the deaf), (6) signals (diving signals, mashalling signals to guide machinery or vehicle, etc.), (7) actions (like drinking or writing), (8) pantomimes (gestures made to mimic actions), and (9) dance postures.

For each sequence, the actor performs an idle gesture between each gesture to classify. These idle gestures are used to provide the temporal segmentation (further details are shown in the next section). For this data set, background subtraction was performed based on depth maps, and a 10×10 grid approach was defined to extract HOG+HOF feature descriptors per cell, which are finally concatenated in a full image (posture) descriptor. Using this data set, the recognition of the idle gesture pattern will be tested, using 100 samples of the pattern in a ten-fold validation procedure.

3.2. Methods and Evaluation

The experiments are presented in two different sections. The first section considers the temporal segmentation experiment while the second section aims the gesture classification experiments.

3.2.1. Temporal Segmentation Experiments

In order to provide with quantitative measures of the temporal segmentation procedure, we first describe the subset of the data used and the feature extraction.

- **Data and Feature extraction**

For the temporal segmentation experiments we used the 20 development batches provided by the organization of the challenge. These batches contain a manual labelling of gesture start and end points. Each batch includes 100 recorded gestures, grouped in sequences of 1 to 5 gestures performed by the same user. For each sequence the actor performs an idle gesture between each gesture of the gestures drawn from lexicons. Finally, this means that we have a set of approximately 1800 idle gestures.

Each video sequence of each batch was described using a 20×20 grid approach. For each patch in the grid we obtain a 208 feature vector consisting of HOG (128 dimensions) and HOF (80 dimensions) descriptors which are finally concatenated in a full image (posture descriptor). Due to the huge dimensionality of the descriptor of a single frame (83200 dimensions), we utilized a Random Projection to reduce dimensionality to 150 dimensions.

- **Experimental Settings**

For both of the DTW approaches the cost-threshold value θ is estimated in advance using ten-fold cross-validation strategy on the set of 1800 idle gesture samples. This involves using 180 idle gestures 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. Finally, the threshold value θ chosen is the one associated with the largest overlapping performance. For the probabilistic DTW approach, each GMM was fit with 4 components. The value of G was obtained using a ten-fold cross-validation procedure on the set of 1800 idle gestures as well. In this sense, the cross-validation procedure for the probability-based DTW is a double loop (optimizing on the number of GMM components G, and then, on the cost-threshold θ). In the HMM case, we used the Baum-Welch algorithm for training, and 3 states were experimentally set for the idle gesture, using a vocabulary of 60 symbols computed using K-means over the training data features. Final recognition is performed with temporal sliding windows of different wide sizes, based on the idle gesture samples length variability.

- **Methods, Measurements and Results**

Our probability-based DTW approach using the proposed distance D shown in Eq. (6) is compared to the usual DTW algorithm and the Hidden Markov Model approach. The evaluation measurements presented are *overlapping* and *accuracy* of the recognition for the idle gesture, considering that a gesture is correctly detected if overlapping in the idle gesture sub-sequence is greater than 60% (the standard overlapping value, computed as the intersection over the union between the temporal bounds in the ground truth, and the ones computed by our method). The accuracy is computed frame-wise as

$$Acc = \frac{TruePositives + TrueNegatives}{TruePositives + TrueNegatives + FalsePositives + FalseNegatives}. \tag{11}$$

The results of our proposal, HMM and the classical DTW algorithm are shown in Table 2. It can be seen how the proposed probability-based DTW outperforms the usual DTW and HMM algorithms in both experiments. Moreover, confidence intervals of DTW and HMM do not intersect with the probability-based DTW in

Table 2: *Overlapping* and *accuracy* results.

	Overlap.	Acc.
Probability-based DTW	**0.3908± 0.0211**	**0.6781 ± 0.0239**
Euclidean DTW	0.3003 ± 0.0302	0.6043 ± 0.0321
HMM	0.2851 ± 0.0432	0.5328 ± 0.0519

any case. From this results it can be concluded that performing dynamic programming increases the generalization capability of the HMM approach, as well as a model defined by a set of GMMs outperforms the classical DTW on RGB-Depth data without increasing the computational complexity of the method. Figure 6 shows qualitative results from two sample video sequences.

3.2.2. BoVDW Classification Experiments

In all the experiments shown in this section, the vocabulary size was set to $N = 200$ words for both RGB and depth cases. For the spatio-temporal pyramids, the volume was divided in $2 \times 2 \times 2$ bins (resulting in a final histogram of 1800 bins). Since the nature of our application problem is one-shot learning (only one training sample is available for each class), a simple Nearest Neighbor classification is employed. Finally, for the late fusion, the weight $\beta = 0.8$ was empirically set, by testing the performance of our method in a small subset of development batches from the dataset. We observed that when increasing β, starting from $\beta = 0$, the performance keeps increasing in a linear fashion, until the value $\beta = 0.45$. From $\beta = 0.45$ to $\beta = 0.8$ the performance keeps improving more slightly, and finally, from $\beta = 0.8$ to $\beta = 1$ the performance drops again.

For the evaluation of the methods, in the context of Human Gesture Recognition, the Levenshtein distance or edit distance was considered. This edit distance between two strings is defined as the minimum number of operations (insertions,

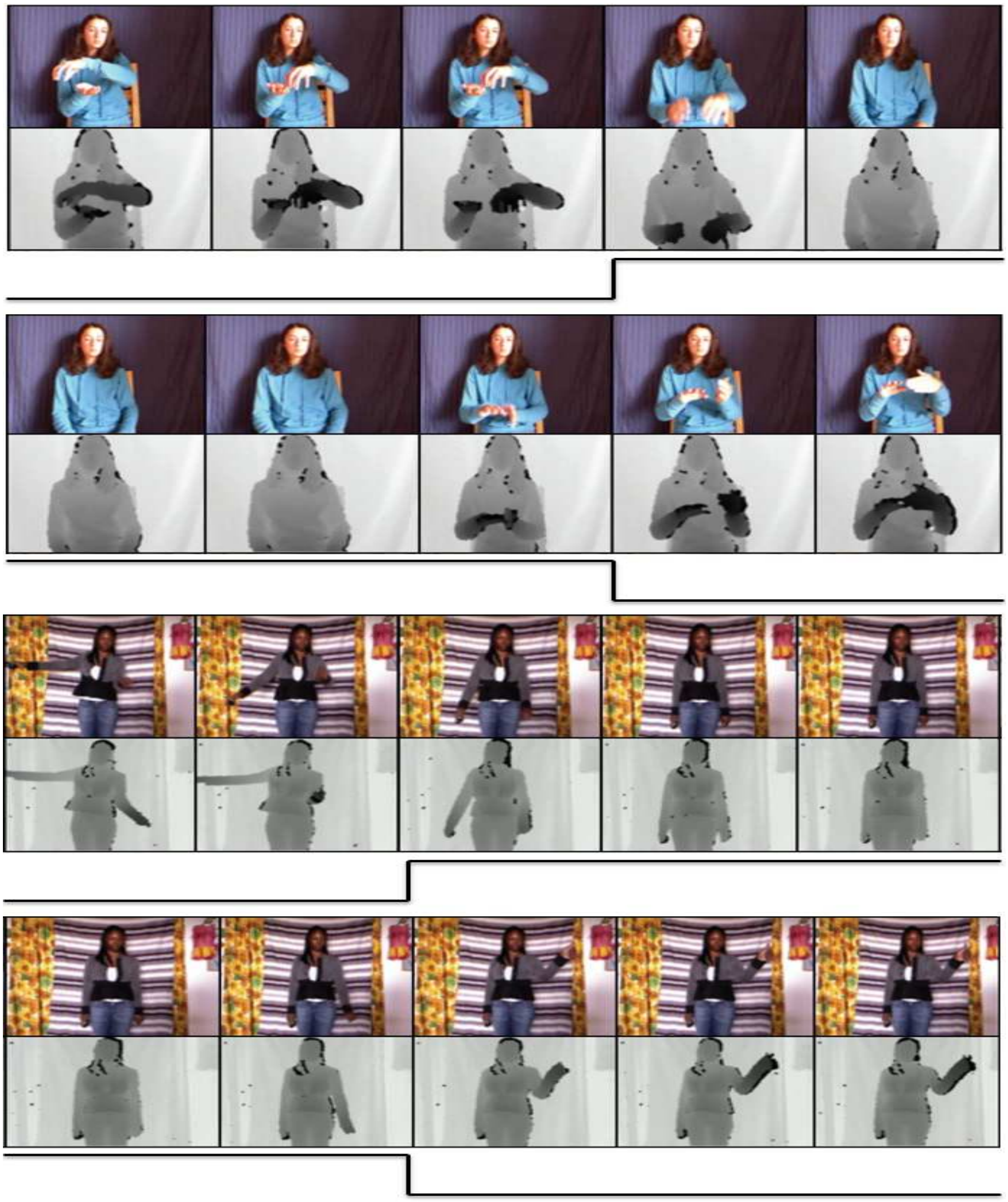


Figure 6: Examples of idle gesture detection on the Chalearn data set using the probability-based DTW approach. The line below each pair of depth and RGB images represents the detection of a idle gesture (step up: beginning of idle gesture, step down: end)

substitutions or deletions) needed to transform one string into the other. In our case, strings contain gesture labels detected in a video sequence. For all the comparison, the mean Levenshtein distance (MLD) was computed over all sequences and batches.

Table 3 shows a comparison between different state-of-the-art RGB and depth descriptors (including our proposed VFHCRH), using our BoVDW approach. Moreover, we compare our BoVDW framework with the baseline methodology provided by the ChaLearn 2012 Gesture Recognition challenge. This baseline first computes differences of contiguous frames, which encode movement information. After that, these difference images are divided into cells forming a grid, each one containing the sum of movement information among it. These 2D grids are then transformed then into vectors, one for each difference image. Moreover, the model for a gesture is computed via Principal Component Analysis (PCA), using all the vectors belonging to that gesture. The eigenvectors are just computed and stored, so when a new sequence arrives, its movement signature first is computed, and then projected and reconstructed using the different PCA models from each gesture. Finally, the classification is performed by choosing the gesture class with lower reconstruction error. This baseline obtains a MLD of 0.5096. Table 4 shows the results in all the 20 development batches separately.

When using our BoVDW approach, in the case of RGB descriptors, HOF alone performs the worst. In contrast, the early concatenation of HOF to HOG descriptor outperforms the simple HOG. Thus, HOF contributes adding discriminative information to HOG. In a similar way, looking at the depth descriptors, it can be seen how the concatenation of the CRH to the VFH descriptor clearly improves the performance compared to the simpler VFH. When using late fusion in order to merge information from the best RGB and depth descriptors (HOGHOF and VFHCRH, respectively), a value of 0.2714 for MLD is achieved. Figure 7 shows

Table 3: Mean Levenshtein distance for RGB and depth descriptors.

RGB desc.	MLD	Depth desc.	MLD
HOG	0.3452	VFH	0.4021
HOF	0.4144	VFHCRH	**0.3064**
HOGHOF	**0.3314**		

the confusion matrices of the gesture recognition results with this late fusion configuration. In general, the confusion matrices follow an almost diagonal shape, indicating that the majority of the gestures are well classified. However, the results of batches 3, 16, 18, 19 are significantly worse, possibly due to the static characteristics of the gestures in these batches. Furthermore, late fusion was also applied in a 3-fold way, merging HOG, HOF, and VFHCRH descriptors separately. In this case the weight β was assigned to HOG and VFHCRH descriptors (and $1-\beta$ to HOF), improving the MLD to 0.2662. From this result it can be concluded that HOGHOF late fusion performs better than HOGHOF early fusion.

4. Conclusion

In this paper, the BoVDW approach for Human Gesture Recognition has been presented using multi-modal RGB-D images. A new depth descriptor VFHCRH has been proposed, which outperforms VFH. Moreover, the effect of the late fusion has been analysed for the combination of RGB and depth descriptors in the BoVDW, obtaining better performance in comparison to early fusion. In addition, a probabilistic-based DTW has been proposed to asses the temporal segmentation of gestures, where different samples of the same gesture category are used to build a Gaussian-based probabilistic model of the gesture in which possible deformations

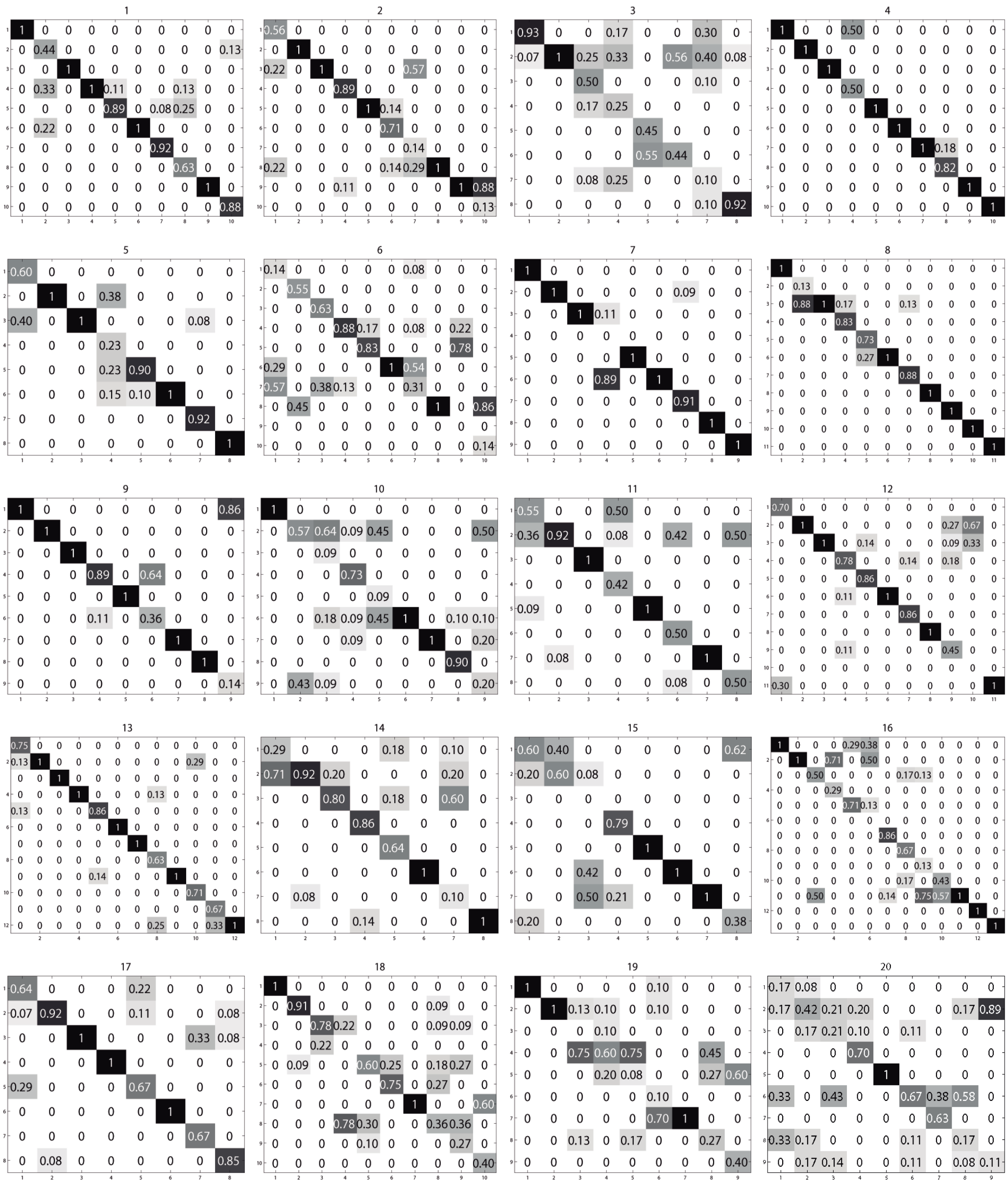


Figure 7: Confusion matrices for gesture recognition in each one of the 20 development batches.

Table 4: Mean Levenshtein Distance of the best RGB and depth descriptors separately, as well as the 2-fold and 3-fold late fusion of them. Results obtained by the baseline from the ChaLearn challenge are also shown. Rows 1 to 20 represent the different batches.

	HOGHOF	VFHCRH	2-fold L.F.	3-fold L.F.	Baseline
1	0.19	0.17	**0.12**	0.20	0.42
2	**0.24**	0.30	**0.24**	0.26	0.57
3	0.76	**0.39**	0.40	0.49	0.78
4	0.14	**0.08**	**0.08**	0.11	0.32
5	**0.08**	0.33	0.17	0.17	0.25
6	0.41	0.47	0.44	**0.34**	0.54
7	**0.10**	0.18	0.11	0.13	0.64
8	0.12	0.26	0.14	**0.08**	0.40
9	**0.11**	0.18	0.15	0.13	0.30
10	0.57	0.40	**0.39**	0.46	0.79
11	0.47	0.36	**0.27**	0.34	0.54
12	0.37	0.20	0.21	**0.17**	0.42
13	0.16	0.14	0.10	**0.09**	0.34
14	0.41	0.34	**0.30**	**0.30**	0.69
15	0.38	**0.28**	0.34	**0.28**	0.54
16	**0.22**	0.41	0.34	0.29	0.42
17	0.38	0.16	**0.15**	0.17	0.55
18	**0.38**	0.43	0.40	**0.38**	0.53
19	0.67	0.50	0.50	**0.44**	0.61
20	**0.46**	0.57	0.56	0.48	0.52

are implicitly encoded. In addition, to embed these models into the DTW framework, a soft-distance based on the posterior probability of the GMM was defined. In conclusion, a novel methodology for gesture detection has been presented, which is able to deal with multiple deformations in data.

Acknowledgements

This work has been partially supported by the “Comissionat per a Universitats i Recerca del Departament d’Innovació, Universitats i Empresa de la Generalitat de Catalunya” and the following projects: IMSERSO Ministerio de Sanidad 2011 Ref. MEDIMINDER, RECERCAIXA 2011 Ref. REMEDI, TIN2012-38187-C03-02 and CONSOLIDER INGENIO CSD 2007-00018. The work of Antonio is supported by an FPU fellowship from the Spanish government. The work of Víctor is supported by the 2013FI-B01037 fellowship. The work of Miguel Ángel is supported by an FI fellowship from SUR-DEC of Generalitat de Catalunya and FSE.

[1] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking, Signal Processing Magazine, IEEE 22 (2) (2005) 38–51.

[2] T. Starner, A. Pentland, Real-time american sign language recognition from video using hidden markov models, in: Computer Vision, 1995. Proceedings., International Symposium on, 1995, pp. 265 –270.

[3] A. Pentland, Socially aware computation and communication, Computer 38 (2005) 33–40.

[4] D. D. Lewis, Naive (bayes) at forty: The independence assumption in information retrieval, in: ECML, Springer Verlag, 1998, pp. 4–15.

[5] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: ECCV, 2004, pp. 1–22.

[6] M. Mirza-Mohammadi, S. Escalera, P. Radeva, Contextual-guided bag-of-visual-words model for multi-class object categorization, in: CAIP, 2009, pp. 748–756.

[7] J. C. Niebles, H. Wang, L. Fei-Fei, Unsupervised learning of human action categories using spatial-temporal words, IJCV 79 (3) (2008) 299–318.

[8] R. Bogdan, N. Blodow, M. Beetz, Fast point feature histograms (fpfh) for 3d registration, in: ICRA, 2009, pp. 3212–3217.

[9] R. Rusu, G. Bradski, R. Thibaux, J. Hsu, Fast 3d recognition and pose using the viewpoint feature histogram, in: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, 2010, pp. 2155 –2162.

[10] K. Lai, L. Bo, X. Ren, D. Fox, Sparse distance learning for object recognition combining rgb and depth information, in: ICRA, 2011, pp. 4007 –4013.

[11] L. Bo, X. Ren, D. Fox, Depth kernel descriptors for object recognition, in: IROS, 2011, pp. 821–826.

[12] T. Wan, Y. Wang, J. Li, Hand gesture recognition system using depth data, in: Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on, 2012, pp. 1063–1066.

[13] Y. Li, Hand gesture recognition using kinect, in: Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on, 2012, pp. 196–199.

[14] F. Pedersoli, N. Adami, S. Benini, R. Leonardi, Xkin -: extendable hand pose and gesture recognition library for kinect, in: ACM Multimedia, 2012, pp. 1465–1468.

[15] K. K. Biswas, S. Basu, Gesture recognition using microsoft kinect, in: Automation, Robotics and Applications (ICARA), 2011 5th International Conference on, 2011, pp. 100–103.

[16] P. Doliotis, A. Stefan, C. McMurrough, D. Eckhard, V. Athitsos, Comparing gesture recognition accuracy using color and depth information, in: Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments, PETRA '11, 2011, pp. 20:1–20:7.

[17] C. Keskin, F. Kıraç, Y. E. Kara, L. Akarun, Real time hand pose estimation using depth sensors, in: Consumer Depth Cameras for Computer Vision, Springer, 2013, pp. 119–137.

[18] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-time human pose recognition in parts from single depth images, in: CVPR, 2011, pp. 1297 –1304.

[19] A. Hernández-Vela, N. Zlateva, A. Marinov, M. Reyes, P. Radeva, D. Dimov, S. Escalera, Human limb segmentation in depth maps based on spatio-temporal graph-cuts optimization, Journal of Ambient Intelligence and Smart Environments 4 (6) (2012) 535–546.

[20] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, A. Fitzgibbon, Efficient regression of general-activity human poses from depth images, in: ICCV, 2011, pp. 415 –422.

[21] F. Zhou, F. De la Torre, J. Hodgins, Hierarchical aligned cluster analysis for temporal clustering of human motion, IEEE TPAMI (2012) 1.

[22] M. Reyes, G. Dominguez, S. Escalera, Feature weighting in dynamic time warping for gesture recognition in depth data, in: ICCV, 2011.

[23] I. Laptev, On space-time interest points, Int. J. Comput. Vision 64 (2-3) (2005) 107–123.

[24] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, CVPR 1 (2005) 886–893.

[25] I. Laptev, M. Marszalek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, in: CVPR, 2008, pp. 1–8.

[26] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. Rusu, G. Bradski, Cad-model recognition and 6dof pose estimation using 3d cues, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, 2011, pp. 585 –592.

[27] Chalearn gesture dataset, california (2011).
URL http://gesture.chalearn.org/data