Using Appearance-Based Hand Features For Dynamic RGB-D Gesture Recognition

Xi Chen and Markus Koskela
Department of Information and Computer Science
Aalto University School of Science
PO Box 15400, FI-00076 AALTO, Finland
Email: xi.chen@aalto.fi, markus.koskela@aalto.fi

***Abstract*—Gesture recognition using RGB-D sensors has currently an important role in many fields such as human–computer interfaces, robotics control, and sign language recognition. However, the recognition of hand gestures under natural conditions with low spatial resolution and strong motion blur still remains an open research question. In this paper we propose an online gesture recognition method for multimodal RGB-D data. We extract multiple hand features with the assistance of body and hand masks from RGB and depth frames, and full-body features from the skeleton data. These features are classified by multiple Extreme Learning Machines on the frame level. The classifier outputs are then modeled on the sequence level and fused together to provide the final classification results for the gestures. We apply our method on the ChaLearn 2013 gesture dataset consisting of natural signs with the hand diameters in the images around 20–40 pixels. Our method achieves an 85% recognition accuracy with 20 gesture classes and can perform the recognition in real-time.**

I. Introduction

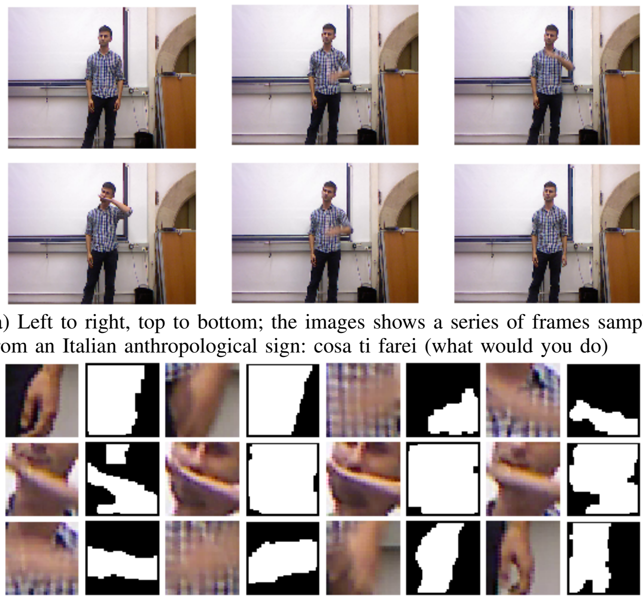
Human action and gesture recognition have been popular research topics for the last few decades [1], [2]. The research has mainly been conducted based on image and video data, with some attention on motion capture. The RGB image is, however, vulnerable e.g. to illumination variation and to cluttered backgrounds. On the other hand, motion capture systems provide very accurate skeletal data, but the high cost of the equipment and software and the mobility limitations prevent wide application of the technology. Nowadays, the commodity RGB-D (RGB and depth) sensors, such as the Microsoft Kinect, which provide depth information along with the standard RGB video, are widely used e.g. in gaming, HCI [3], and robotics [4] due to the portability and low cost. Several algorithms have been developed to extract the human skeleton from the depth maps in real-time [5]. These algorithms classify a large 3D point cloud into about a dozen human skeleton joint coordinates and thus provide analogous, albeit noisier, data compared to motion capture.

Previously the research on gesture recognition has often been based on only a single data modality, such as the RGB frames, depth information, or the skeleton model. As the availability of multiple data sources has recently increased, multimodal methods have started to show their advantages over a single data modality. In our work, we use multimodal data from the skeleton model, RGB, and depth through fusion in a common gesture recognition framework.

Gesture recognition systems can be roughly divided into *body gesture* and *hand gesture* recognition systems. As the name implies, the former systems focus on whole body movements and the latter ones concentrate on hand configurations and movement of the hands and arms. Hand gesture recognition can further be divided into two classes, *static* and *dynamic* gesture recognition. For static recognition, each frame represents a hand gesture, for example an OK sign, and by processing one frame the system recognizes one gesture. On the other hand, dynamic gestures are composed of multiple frames, and these series of frames containing hand information form together one hand gesture, such as waving hand. In other words, a dynamic gesture can be considered as a combination of multiple static gestures, which makes the dynamic gesture recognition more challenging.

Dynamic gesture recognition is often used for human–computer interfaces, gaming, and recognition of natural gestures and sign language. Compared to control applications, where the performer can often control the speed of the hand movement to ease the recognition, e.g. the signing of sign language is performed in a faster and more natural way, which easily results in more motion blur (see Fig. 1). In our work we build a dynamic hand gesture recognition system for language-related applications using RGB-D data. This is a challenging problem due to the complexity of the visual cues, uncontrolled setting, and low spatial and temporal resolution [6].

A lot of research has been conducted on hand gesture recognition. The main approaches can be roughly grouped into appearance-based and model-based modeling [7]. Especially recently, due to the depth information being easily accessible, the model-based methods are gaining more attention [8], [9]. However, in almost all studies the hands are presented in good lighting conditions and in relatively high resolution: they either dominate the image [10], or are at least in the range of 100×100 pixels [11]. In order to segment the hands, skin color is often used as a cue with RGB data [8], [12]; in some situations, the gestures are constrained by stretching the hands far front from the body so the hands can be easily segmented by depth information [13]. In our task, the gestures are performed in a natural way in varying lighting conditions and with the hand regions of about 20–40 pixels in width and height, as illustrated in Fig. 1. In this paper, we propose the use of simple appearance-based hand features in online recognition of natural dynamic hand gestures based on RGB-D sensor data. This paper is a continuation of our initial experiments in [14]; here we provide a comprehensive evaluation of different appearance-based hand features, propose hand segmentation, and show considerably improved recognition results.



(a) Left to right, top to bottom; the images shows a series of frames sampled from an Italian anthropological sign: cosa ti farei (what would you do)

(b) A closer look of the left hand with double sampling rate from the above series, each image is 40×40 pixels, and the corresponding depth-based masks

Fig. 1: Example of a dynamic gesture (an Italian cultural sign)

Hidden Markov Models (HMM), Conditional Random Fields (CRF) and Support Vector Machines (SVM) are popular classifiers often used in gesture recognition systems. These algorithms require, however, high computation resources and a lot of data for training the models. Online recognition of the gestures can also be difficult due to the computational requirements of the evaluation of the classifiers, especially in the case of SVMs. In our work we use extreme learning machines (ELM) [15] in a frame-based recognition scheme with simple and low-complexity features, enabling gesture recognition in real-time. Furthermore, the ELM model training for one feature only requires a couple of minutes with datasets of realistic size, which may be considered as "online" training for many applications.

II. Related Work

Hand gesture recognition using RGB-D data has recently gained more focus in the computer vision community [10], [11], [16], [17]. In many applications, in order to be able to extract clear hand shapes or features, the hands are required to occupy a significantly large portion of the input image. For example, in [10], 19 static hand gestures are recognized for robot control. The hand is separated from the background using the depth image with depth thresholding and is represented by the hand contour and by the tips of the fingers. Similarly, [13] segments the hand also by depth information, with the hand restricted to lie within a certain depth range and be outstretched from the body. The contour and the tips of the hand are extracted and nine static hand gestures are recognized.

As the resolution of the hand size gets smaller, it becomes more challenging to robustly detect and segment the hands. In some solutions, certain accessories are used to facilitate the recognition. In [18], the authors claim that it is impossible to recognize hand gestures over one meter distances due to the low resolution of the Kinect sensor, so they use an additional HD camera and color gloves. The contour of static hand gestures is extracted as a polygon and classified by polygon matching, with 11 static gestures in the test set. In [16], the user is required to wear a black belt on the wrist to separate the hand from the arm, and the shape of the hands is described as a time-series curve. Finger Earth Mover's Distance is then used to measure the dissimilarity between 10 static hand shapes.

Instead of extracting the shape of the hands as features, some feature extraction methods are directly applied to the bounding box of the hand. In [11], a Gabor filter is applied on the hand bounding box and classification with 26 static hand gestures is performed with random forests.

Dynamic gesture recognition is closely related with sign language recognition. [19] uses a depth sensor and an action graph to classify between 12 dynamic American Sign Language (ASL) signs with a reported accuracy of 87.7%. A cell occupancy feature and a silhouette feature are extracted from the hand region. In [17], 19 ASL signs are recognized by a combination of Kinect skeleton models and hand features with an accuracy of 76.12%. The angular information between the joints is extracted from the skeleton, and each hand is modeled by a mixture of six Gaussians. PCA is then applied onto the 72 MoG parameters to reduce the dimensionality of the hand feature to 20 dimensions.

III. Hand Features

Gesture and action recognition based purely on skeletal data has proved to be useful in many situations [20], [21]. The skeletal features are nevertheless not able to capture hand configurations, which often represent meaningful linguistic symbols. Examples shown in Fig. 2 illustrate one case where the skeletal features are not alone sufficient to distinguish between the gestures.

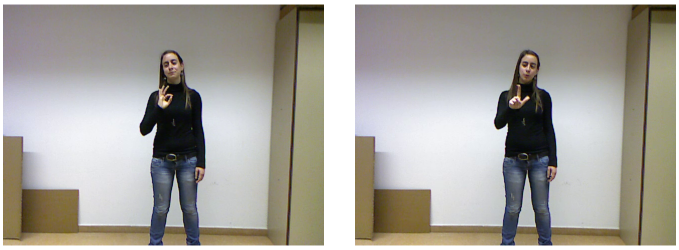


Fig. 2: Different gestures with similar skeleton alignment

The tracking of the hands in uncontrolled settings and with natural gestures is a challenging task due to high dimensionality of the hand configuration, illumination variation, self-occlusion, and motion blur. Several methods (e.g. [8]) require robust skin color segmentation which can be difficult in uncontrolled environments, especially if no initialization can be performed. In this work, we use the 2D hand locations provided by the Kinect skeleton model and extract features from fixed hand regions centered at the hand locations. We divide the hand region into a regular grid of pixel cells. To compensate for the inaccuracies in the determination of the hand locations, we use reasonably large cells of pixels.

A. Static Hand Features

Histograms of oriented gradients (HOG) [22], originally proposed for human pedestrian detection, have recently been successfully used in many other applications as well. In this work we extract HOG features according to [23], resulting in a 31-dimensional feature for each cell.

Local binary patterns (LBP) [24]. We extract LBPs with an 8-neighborhood and a radius of 2 pixels, and perform the uniform mapping of the LBP labels, resulting in a 59-dimensional histogram feature for each cell.

Gabor filter based hand feature (Gab.), extracted according to [11]. The hand images are convolved with a bank of Gabor filters at 4 scales and 4 orientations. A Gaussian function centered at the center of each cell is applied on the convolved image and the values are summed across the whole image to form each element of the feature vector. The dimensionality of the feature is 16 for each cell in the used grid.

B. Hand Segmentation

Robust segmentation of the hand regions for dynamic hand gesture recognition is challenging due to several aspects. As already mentioned, skin color segmentation can be challenging in an uncontrolled setting. We therefore use the depth information to obtain segmentations of the hands. The *body mask* is obtained by segmenting the full body of the performer of the gestures from the background. This is straightforward provided that there is enough depth difference between the performer and the background.

For obtaining the *hand mask*, we assume that the corresponding hand is the closest object to the RGB-D sensor within the considered hand region. We then mark all pixels with depth value less that a certain threshold to belong to the hand object. As a post-processing step, we use morphological opening and closing to smooth the mask boundaries (see Fig. 1).

C. Temporal Hand Features

We also extract two kinds of temporal hand features. *HOG3D* [25] is a spatio-temporal descriptor of histograms of 3D gradient orientations. The binning of the gradients is done in polar coordinate space. We use the default values of 5 and 3 bins for the xy and xt planes, respectively, resulting in a 15-dimensional feature for each cell.

Histogram of oriented 4D normals (HON4D) [26] was recently proposed to describe a depth sequence as a histogram of surface normal orientations in 4D. The holistic descriptor described in [26] represents the whole sequence, whereas we are extracting frame-level descriptors. The authors however propose also a local HON4D descriptor extracted around skeletal joints, which is the variant we use in this paper. The dimensionality of the descriptor is 120.

IV. Gesture Recognition Framework

In this section, we provide a brief description of the full gesture recognition framework used in the experiments of this paper. See Fig. 3 for a block diagram of the framework. A more complete description of the framework is given in [14].

A. Skeletal Features

We extract two features from the skeletal data: normalized 3D joint positions and pairwise distances between joints.

Normalized 3D Joint Positions (NP). The skeletal data provides 3D joint positions of the whole body. The 3D coordinates of these joints are, however, not invariant to the position and size of the actors. Therefore we transform all skeletons into the same orientation by aligning the plane formed by the root and the hips from all frames into the same plane [20]. To make the feature size-invariant, we also normalize the skeletons so that the sum of the distances of all connected joints is one. For gestures related to whole body movement, the whole set of joints from the above feature can be used; for gestures only with partial body movement, such as hand and arm gestures, a subset can be selected. In this work, we use the following upper-body joints: the spine, shoulder center, head, shoulders, elbows, wrists and the hands.

Pairwise Distances (PD). We also extract the pairwise distances between the joints from the skeletal data. The distances form a vector which is then L_1-normalized to one. In this work, the used joints include the above 11 joints and the hip center.

Temporal Differencing. A gesture is formed by a sequence of frames. In order to preserve temporal information in the sequence, we calculate the temporal difference of features in the sequence using a fixed temporal offset [20]. The final skeletal features (NP and PD) are obtained by concatenating the original features and the temporal differences.

B. Early Fusion

Each kind of feature has its own advantages to capture distinctive information from the original data whereas a combination of features can compensate each other and enforce the distinctiveness of the features. Therefore, in the early fusion stage, we concatenate two or more features before the frame-level classification.

C. Frame-Level Classification

Let us assume there are M gestures $\mathcal{A} = \{A_1, \ldots, A_M\}$ and let us define $c_m \in \{0, 1\}$, $1 \leq m \leq M$. If c_m is one, then the sequence belongs to the gesture A_m, otherwise it does not. The row vector $\mathbf{y} = [c_1 \ \ldots \ c_m \ \ldots \ c_M]$ indicates the gesture that the sequence belongs to. Each gesture sequence s is represented by the features of its frames, i.e. $s = \{\mathbf{x}_1, ..., \mathbf{x}_k, ..., \mathbf{x}_K\}$, where K is the number of frames. Now, $(\mathbf{x}_k, \mathbf{y})$ form K training input–output pairs for the classifier.

The Extreme Learning Machine (ELM) [15] belongs to the class of single-hidden layer feed-forward neural networks. Traditionally such networks have been trained using a gradient-based method such as the backpropagation algorithm. In ELM, the hidden layer weights and biases do not need to be learned but are assigned randomly, which makes the learning extremely fast. The only unknown parameters are the output weights which can be obtained by finding a least-squares solution.

Given P training samples $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{P}$, where $\mathbf{x}_i \in \mathbb{R}^n$ and $\mathbf{y}_i \in \mathbb{R}^M$, the standard ELM model with L hidden neurons

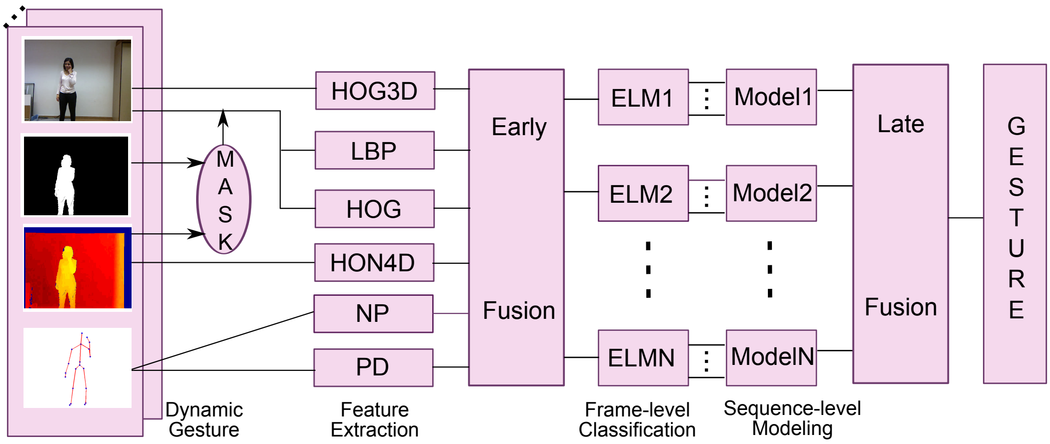


Fig. 3: The framework of the dynamic gesture recognition system

can be represented as

$$\mathbf{y}_i = f(\mathbf{x}_i) = \sum_{j=1}^{L} \boldsymbol{\beta}_j g(\boldsymbol{\omega}_j \cdot \mathbf{x}_i + b_j) \ , \tag{1}$$

where $g(\cdot)$ is a nonlinear activation function, $\boldsymbol{\beta}_j \in \mathbb{R}^M$ are the output weights, $\boldsymbol{\omega}_j \in \mathbb{R}^n$ is the weight vector connecting the input layer to the *j*th hidden neuron and b_j is the bias of the *j*th hidden neuron. Both $\boldsymbol{\omega}_j$ and b_j are assigned randomly during the learning process. With $\mathbf{Y} = [\mathbf{y}_1^T \ \mathbf{y}_2^T \cdots \mathbf{y}_P^T]^T \in \mathbb{R}^{P \times M}$ and $\boldsymbol{\beta} = [\boldsymbol{\beta}_1^T \ \boldsymbol{\beta}_2^T \cdots \boldsymbol{\beta}_L^T]^T \in \mathbb{R}^{L \times M}$, Eq. (1) can be written compactly as

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y} \ , \tag{2}$$

where the hidden layer output matrix $\mathbf{H}$ is

$$\mathbf{H} = \begin{bmatrix} g(\boldsymbol{\omega}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\boldsymbol{\omega}_L \cdot \mathbf{x}_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(\boldsymbol{\omega}_1 \cdot \mathbf{x}_P + b_1) & \cdots & g(\boldsymbol{\omega}_L \cdot \mathbf{x}_P + b_L) \end{bmatrix}_{P \times L} . \tag{3}$$

If $L = P$, the matrix $\mathbf{H}$ is square and invertible, and the model can approximate the P training samples with zero error. However, in most cases the number of hidden neurons is much smaller than the number of training samples, i.e. $L \ll P$, and we obtain the smallest norm least-squares solution of (2) as

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{Y} \ , \tag{4}$$

where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse of $\mathbf{H}$.

D. Sequence-Level Gesture Recognition

Given a test sequence $\{\mathbf{x}_1, \ldots, \mathbf{x}_q, \ldots, \mathbf{x}_Q\}$, ELM provides the output weight (Eq. (1)) of each class m for each frame. We convert the outputs into probabilities with logistic sigmoid

$$p(c_m = 1 | \mathbf{x}_q) = \frac{1}{1 + \exp(-\gamma y_{qm})} \ , \tag{5}$$

where y_{qm} is the mth component of $\mathbf{y}_q$.

We aggregate the frame-level probabilities to form the sequence-level classification result using a function d_m : $\mathbb{R}^Q \rightarrow \mathbb{R}$. We use here the weighted arithmetic mean

$$d_m = \sum_{q=1}^{Q} w_q \, p(c_m = 1 \mid \mathbf{x}_q) \tag{6}$$

where the weights w_q are obtained from a normalized Gaussian distribution, $w_q = \frac{1}{Z}\mathcal{N}(q; \frac{Q}{2}, \sigma^2)$, normalized so that $\sum_{q=1}^{Q} w_q = 1$.

E. Late Fusion

The final stage in the recognition framework is late fusion, where we use the geometric mean to fuse the sequence-level classification outputs of the different subsets of the feature-wise and early-fusion results, $\bar{d}_m = \prod_j d_m^j$, where d_m^j is the sequence-level classification result for the jth feature.

Finally, we classify a test sequence by

$$\hat{c}_m = \begin{cases} 1 & \text{if} \quad m = m' \quad \text{where } m' = \arg\max_i \bar{d}_i \\ 0 & \text{otherwise} \ . \end{cases} \tag{7}$$

V. Experiments

In this section, we describe gesture recognition experiments performed with the ChaLearn Multi-modal Gesture Recognition Challenge 2013 dataset [27], [6]. Our setup here differs from that of the common challenge as we consider here gesture recognition only, i.e. we assume that the start and end points of the gestures are known, and we do not use the audio modality.

A. Settings

1) ChaLearn 2013 dataset: The ChaLearn 2013 dataset consists of $M = 20$ Italian cultural or anthropological signs, examples of which are shown e.g. in Fig. 1 and Fig. 2, recorded with a Kinect sensor. The data consists of multiple modalities: RGB, depth maps, skeleton models, and audio. Our focus in this work is on visual gesture recognition so we omit the audio modality. This however makes our results rather incomparable to the challenge submissions, as the performance of the audio modality was found to be superior to the other modalities [28]. The dataset contains three parts: training (7754 gestures), validation (3362 gestures), and test data (2742 gestures). The

TABLE I: Results with different hand features

	static features									temporal features	
	no mask			body mask			hand mask				
grid	HOG	LBP	Gab.	HOG	LBP	Gab.	HOG	LBP	Gab.	HOG3D	HON4D
2×2	59.9	50.9	44.9	62.5	59.3	50.9	68.1	59.5	55.3	54.3	
3×3	64.3	50.8	50.1	65.0	60.7	55.0	**68.9**	59.5	57.6	61.1	63.5*
4×4	65.0	-	49.3	64.7	-	54.8	67.1	-	58.5	60.8	

gestures are performed by 27 actors, and each part of the dataset is performed by a different set of actors. We use about 6000 gesture sequences from the training data for learning our models. As our test set, we use the challenge validation data. This is due to the lack of start and end points for the gestures in the actual provided test data of the challenge.

2) Determination of the dominant hand: As the ChaLearn 2013 gestures are cultural signs, most of them can be performed with either hand as the dominant one. Moreover, in some gestures the performers do use both hands but generally in a symmetric way. We therefore determine the dominant hand for each gesture instance by measuring the total scope of movement in 3D of both hands. The hand with a larger movement scope is marked as the dominant one. We train separate ELMs for the left and right dominant hands, and during classification select the used ELM models based on similar scope analysis of the current gesture. This was shown to improve recognition accuracy in [14].

3) Parameters: In all experiments, we use fixed hand regions of 40×40 pixels centered on the hand locations provided by the skeleton model. The region is divided into either 2×2, 3×3, or 4×4 cells. For ELM, the number of hidden neurons L is the only parameter. We also have the parameters γ and σ^2, corresponding to the slope of the logistic sigmoid in (5), and the variance of the Gaussian weighting function in (6), respectively. In these experiments, we use the values $L = 1500$, $\gamma = 1$, and $\sigma = 0.2Q$. As the temporal offset of the skeleton features (Section IV-A), we use an offset of six frames, corresponding to 300 milliseconds.

B. Results

The used gesture recognition framework (Section IV) supports any number of parallel features to be used for recognition. The results of using the different hand features as single features are shown in Table I. The LBP features were not calculated for 4×4 cells due to the high dimensionality of the features, and the HON4D feature was extracted without any cell structure. First, we can observe that, for HOG and Gabor features, using 3×3 cells seems to be a good compromise of accuracy and feature dimensionality: the results are improved over 2×2 cells, but further increasing the number of cells does not improve the accuracy. For LBPs, using 2×2 cells seems to be quite enough. Second, the hand segmentation is clearly beneficial. The recognition accuracies with the hand masks obtained with depth thresholding are higher than without the masks. Interestingly, the body mask seems to be sufficient for LBP features. The temporal features HOG3D and HON4D are not superior to the static features, making it quite hard to justify the extra complexity.

TABLE II: Selected fusion results of skeletal and hand features; the symbols "||" and "+" denote early and late fusion

used features	accuracy
NP	71.5
PD	70.4
NP||PD	73.5
NP+PD	73.1
NP+PD+$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$	82.9
NP+PD+$\mathrm{LBP}^{\mathrm{bo}}_{3\times3}$	79.5
NP+PD+$\mathrm{HOG3D}_{3\times3}$	80.4
NP+PD+HON4D	80.1

used features	accuracy
NP+PD+$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$+$\mathrm{LBP}^{\mathrm{bo}}_{3\times3}$	83.7
NP+PD+$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$+ $\mathrm{HOG3D}_{3\times3}$+HON4D	85.3
NP||$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$	82.7
NP||PD||$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$	83.9
NP||PD||$\mathrm{LBP}^{\mathrm{bo}}_{3\times3}$	78.9
NP+$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$+$\mathrm{HOG3D}_{3\times3}$+ (NP||PD||$\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$)	**85.5**

A selected set of results using early, late, and both feature fusion methods are shown in Table II. The baseline of using only (concatenated) skeletal features obtains an average accuracy of 73.5%. A considerable improvement can then be obtained by including just one hand feature; the $\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$ feature[1] with the highest single-feature accuracy is a natural choice. With either an early or late fusion strategy, this raises the accuracy to about 83–84%. Rather small further improvements can then be obtained by including more features. The overall highest accuracy of 85.5% is obtained by using several features and both early and late fusion. The HOG features perform consistently better than the LBP and Gabor features, but the latter can still be beneficial in fusion in some cases.

As mentioned in Section V-A, due to the use of the validation set ground truth, comparing our results to the challenge evaluation is purposeless. We can however compare our results to those of the winners of challenge, as they provide their recognition results also for the validation set in [28]. The winners' recognition system was based on audio and skeletal information. They used MFCC features and Gaussian HMMs for audio, and a Dynamic Time Warping based classifier for the skeletons. For the validation set, they report accuracies of 60.0%, 93.5%, and 99.6% for the skeleton features, audio features, and feature fusion, respectively.

The experiments are conducted in on a Intel(R) Xeon(R) CPU at 3.3 GHz and 16 GB of memory. For example, Table II shows that the early fusion of NP, PD and $\mathrm{HOG}^{\mathrm{ha}}_{3\times3}$ give a reasonable trade off between accuracy and complexity. The feature extraction of NP, PD, and HOG takes serially about 25 milliseconds and the classification by a single ELM is about 0.1 ms per frame, therefore the recognition can be accomplished in real time. Furthermore, in this configuration only one ELM is needed for each hand, which corresponds to about 1 to 3 minutes required to train the ELM classifiers for the concatenated features for the whole training dataset.

VI. Conclusions

We approach the problem of online dynamic gesture recognition from the viewpoint of static pose recognition and use ELM as a standard multi-class classifier for frame-level classification. The results are then aggregated into the sequence level by weighted averaging. This approach provides an adaptive and fast method for gesture recognition that has also been successfully applied to full-body motion capture

[1] The superscript and subscript refer to the used mask and cell structure.

action classification with a large number of classes [20]. In this paper, we study the usefulness of simple appearance-based hand features in the recognition of natural dynamic hand gestures in difficult conditions. Advanced 3D hand models, such as in [8], undoubtedly have the potential to provide more precise information about the hand configuration, but are challenging to apply in an uncontrolled, temporally and spatially low-resolution setting.

The ChaLearn 2013 dataset used in this work contains gestures that are difficult to separate based on the skeleton model alone, and the introduced Gabor, LBP and HOG based features can provide useful information about the hand configurations. The feature fusion experiments show that the hand features indeed can increase the recognition accuracy even though the used skeletal features were slightly more accurate as single features on average. The dataset contains relatively many gestures and is quite challenging for visual-only approaches. The temporal hand features did not considerably improve the results in this study, so the simpler image-based features might be preferable especially in online recognition setups.

In this work, we have limited the discussion to closed gesture recognition, that is, we have assumed that the start and end points for the gestures are known and that each performed gesture belongs to exactly one of the prespecified gesture classes. Generally, in an online setup, this is not the case and we have to perform both temporal gesture segmentation or gesture spotting and thresholding to reject non-gestures and gestures that do not belong to any of the known gesture classes. There are many proposed approaches for both problems, but they are still largely unsolved. The basis of our method, i.e. static multi-class gesture classification on the frame level, would however suggest that our method can easily be adapted into the sliding-window continuous gesture recognition framework. The proposed method is also readily applicable to sign language recognition.

References

[1] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 37, no. 3, pp. 311–324, May 2007.

[2] M. Müller and T. Röder, "Motion templates for automatic classification and retrieval of motion capture data," in *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Computer animation*, Vienna, Austria, 2006, pp. 137–146.

[3] T.-T. Chu and C.-Y. Su, "A Kinect-based handwritten digit recognition for TV remote controller," in *IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012)*, 2012.

[4] M. Peris and K. Fukui, "Both-hand gesture recognition based on KOMSM with volume subspaces for robot teleoperation," in *2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2012, pp. 191–196.

[5] 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," in *Proc. Computer Vision and Pattern Recognition*, June 2011.

[6] S. Escalera, J. Gonzàlez, X. Baró, M. Reyes, O. Lopes, I. Guyon, V. Athitsos, and H. J. Escalante, "Multi-modal gesture recognition challenge 2013: Dataset and results," in *ChaLearn Multi-modal Gesture Recognition Grand Challenge and Workshop, 15th ACM International Conference on Multimodal Interaction*, 2013.

[7] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review," *Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, 1997.

[8] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect," in *Proceedings of the 22nd British Machine Vision Conference (BMVC2011)*, 2011.

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

[10] M. Biao, X. Wensheng, and W. Songlin, "A robot control system based on gesture recognition using Kinect," *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 11, no. 5, pp. 2605–2611, 2013.

[11] N. Pugeault and R. Bowden, "Spelling it out: Real-time ASL fingerspelling recognition," in *2011 IEEE International Conference on Computer Vision Workshops*. IEEE, 2011, pp. 1114–1119.

[12] H. Hamer, K. Schindler, E. Koller-Meier, and L. Van Gool, "Tracking a hand manipulating an object," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1475–1482.

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

[14] X. Chen and M. Koskela, "Online RGB-D gesture recognition with extreme learning machines," in *Proceedings of the 15th ACM International Conference on Multimodal Interaction (ICMI 2013)*. Sydney, Australia: ACM, December 2013, pp. 467–474.

[15] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1, pp. 489–501, 2006.

[16] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, "Robust part-based hand gesture recognition using Kinect sensor," *IEEE transactions on multimedia*, vol. 15, no. 5, pp. 1110–1120, 2013.

[17] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, "American sign language recognition with the Kinect," in *Proceedings of the 13th international conference on multimodal interfaces*. ACM, 2011, pp. 279–286.

[18] M. Caputo, K. Denker, B. Dums, and G. Umlauf, "3D hand gesture recognition based on sensor fusion of commodity hardware." in *Mensch & Computer*, 2012, pp. 293–302.

[19] A. Kurakin, Z. Zhang, and Z. Liu, "A real time system for dynamic hand gesture recognition with a depth sensor," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*. IEEE, 2012, pp. 1975–1979.

[20] X. Chen and M. Koskela, "Classification of RGB-D and motion capture sequences using extreme learning machine," in *Proceedings of the 18th Scandinavian Conference on Image Analysis*, ser. LNCS, vol. 7944. Espoo, Finland: Springer Verlag, Jun. 2013.

[21] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 1290–1297.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[23] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[24] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, January 1996.

[25] A. Kläser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3D-Gradients," in *British Machine Vision Conference*, 2008.

[26] O. Oreifej, Z. Liu, and W. Redmond, "HON4D: Histogram of oriented 4d normals for activity recognition from depth sequences," in *Computer Vision and Pattern Recognition (CVPR)*, 2013.

[27] ChaLearn, "Multi-modal Gesture Recognition Challenge 2013," *http://gesture.chalearn.org/.*

[28] J. Wu, J. Cheng, C. Zhao, and H. Lu, "Fusing multi-modal features for gesture recognition," in *Proceedings of the 15th ACM on International Conference on Multimodal Interaction*, ser. ICMI '13. New York, NY, USA: ACM, 2013, pp. 453–460.