Shape Recognition for Irish Sign Language Understanding

LIVIU VLADUTU
Dublin City University
School of Computing
Glasnevin, Dublin 9
IRELAND
lvladutu@computing.dcu.ie

Abstract: The recognition of humans and their activities from video sequences is currently a very active area of research because of its many applications in video surveillance, multimedia communications, medical diagnosis, forensic research and sign language recognition. Our system is designed with the aim to precisely identify human gestures for Irish Sign Language (ISL) recognition. The system is to be developed and implemented on a standard personal computer (PC) connected to a colour video camera. The present paper tackles the problem of shape recognition for deformable objects like human hands using modern classification techniques derived from artificial intelligence.

Key–Words: Statistical Learning, Shape recognition, Sign-Language

1 Introduction

The purpose of the project is to develop a system for Irish Sign Language (ISL) understanding. Sign languages are the native languages by which communities of Deaf communicate throughout the world. Despite the great deal of effort in Sign Language so far, most existing systems can achieve good performance only with small vocabularies or gesture datasets. Increasing vocabulary inevitably incurs many difficulties for training and recognition, such as the large size of required training set, variations due to signers and to recording conditions and so on. Up to now the Deaf people had to communicate usually through an interpreter or through written forms of spoken languages, which are not the native languages of the Deaf community. The aim of the project is to develop this system using vision based techniques, independent of sensor- based technologies (using gloves) that can prove to be expensive, uncomfortable to wear, intrusive and limit the natural motion of the hand. The images are extracted from simple 'one-shot' video-streams, where only an individual gesture is executed, not linked to the consecutive signs (as in a normal sign-language conversation).

1.1 Main steps

The classical steps of this type if human-computer interaction (HCI) system that were implemented by members of the team (see also [2, 3]) are:
- hands and face detection;
- tracking of the above mentioned human body parts using Hidden Markov Models;
- shapes coding and classification using Machine Learning techniques ;
- elimination of small area occlusion problems (hand-hand or hand-face occlusion) using motion estimation and compensation (Figure 1 below);
- construction of faster implementation aiming the real-time ISL understanding system, using faster programming environments (mex programs based on C++ implementation);

1.2 Short description

The work presented in the current paper investigates the detection of subunits (that compose a sign) from the viewpoint of human motion characteristics. In the model the subunit is seen as a continuous hand action in time and space; therefore, the clear shape understanding at certain moments in time Representative Frames (RFrames) for human action understanding is essential. One of the problems we faced is that the hand is a highly deformable articulate object with up to 28 degrees of freedom. Also the skin detection for segmentation, see [2, 7] is based on the assumption that skin color is quite different from colors of other objects and its distribution might form a cluster in some specific color-spaces. Even in the case of specific difficult conditions, i.e. fast segmentation imposed by the online requirement of the design, workarounds were found. The previous coding approaches were dictated by classical implementation in the field, using principal component analysis (PCA), like [11], influenced by new insights of the machine learning and statistical learning theory theory [13].

We detect the skin by combining 3 useful features: colour, motion and position. These features together, represent the skin colour pixels that are more likely to be foreground pixels and are within a predicted position range. Machine Learning is a rich field of knowledge-discovery in data, but the severe restrictions imposed by having in the end to have a real-time ISL recognition system running on a PC (not a server) and with a regular digital camera for providing the input data has imposed us to limit our tests to acceptable classification method. The material, was a database of video streams created by our group using a PC and a handycam, but also a proprietary database of synthetic images (using virtual signers) created using Poser and the Python programming language.

2 The proposed approach

2.1 The skin model

The current work was based on a large experience in skin-model detection ([2, 7] and related) using the presumption of the human bodies being ”under uniform lighting”. The proposed algorithm is fully automatic and adaptive to different signers (real persons or virtual speakers, implemented in Poser). The skin detector is responsible for segmenting skin objects like the face and hands from video frames and it works ’in tandem’ with the tracker which keeps track of the hand location and detects any occlusions that might happen between any skin objects. Due to this feature, (skin detector closely interacting with the HMM-based tracker) has also solved the problem of small occlusions (hand-hand or hand-face), see Figure 1 below. The skin detector + tracker scheme can be seen at work, see [12], but the tracking doesn’t make the object of the current paper. The present work, tackled mainly one-handed gestures corresponding to ISL, see also ”The Standard Dictionary of Irish Sign Language”, [14].

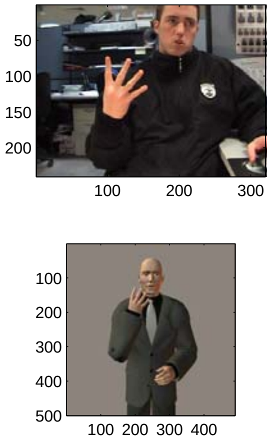

Figure 1: Example of an image from real signer (above) and the equivalent one from a Poser-generated video (below)

2.2 The features space

To evaluate the retrieval performance of the proposed method, a large number of experiments were carried on a set of images recorded by a member of our group and a a database of images created using Python and Poser, see [1]. I have used descriptors from MPEG-7, formally known as Multimedia Content Description Interface and includes standardized tools (descriptors, description schemes, and language) enabling structural, detailed descriptions of audiovisual information. Two MPEG-7 visual descriptors are used in the experiments. The Color Layout descriptor (CLD) provides information about the spatial color distribution within images. After an image is divided in 64 blocks, this descriptor is extracted from each of the blocks based on Discrete Cosine Transform.We can evaluate the distance between two CLD vectors using the formula with luminance and 2 chrominance channels information:

$$S_{CLD}(Q,I) = \sqrt{\sum_i w_{yi}(Y_{Qi}-Y_{Ii})^2} + \sqrt{\sum_i w_{Cbi}(Cb_{Qi}-Cb_{Ii})^2} + \sqrt{\sum_i w_{Cri}(Cr_{Qi}-Cr_{Ii})^2}$$

where w_i represents the weight associated with coefficient i. There are 12 coefficients extracted for the color layout descriptor (6 for Y, and 3 each for Cb and Cr). There is a more detailed description in the MPEG-7 ISO schema files, [8]. The region based shape descriptor belongs to the broad class of shape analysis techniques based on moments . It uses a complex 2D Angular Radial Transformation (ART) , defined on a unit disk in polar coordinates. The ART coefficients were recorded from each segmented image after selecting (cropping) the body part of interest (face or hands). From each shape, as set of ART coefficients, F_{nm}, is extracted, using the following formula:

$$F_{nm} = \Big\langle V_{nm}(\rho,\theta) - f(\rho,\theta) \Big\rangle$$
$$= \int_0^{2\pi}\int_0^1 V^*_{nm}(\rho,\theta)f(\rho,\theta)\rho d\rho d\theta$$

where $f(\rho,\theta)$ is an image intensity function in polar coordinates and $V_{nm}(\rho,\theta)$ is the ART basis function of order n and m. The basis functions are separable along the angular and radial directions, and are defined as follows:

$$V_{nm}(\rho,\theta) = \frac{1}{2\pi} exp(jm\theta) R_n(\theta), \tag{1}$$

$$R_n(\theta) = \begin{cases} 1 \quad if \quad n = 0, \\ 2cos(\pi n \rho) \quad \text{if} \quad n \neq 0 \end{cases} \tag{2}$$

The default region-based shape descriptor has 140 bits. It uses 35 coefficients (n=10, m=10) quantized to 4 bits per coefficient. I have used in the object description all the 35 resulted coefficients. The region based shape descriptor expresses pixel distribution within a 2D object region; it can describe complex objects consisting of multiple disconnected regions as well as simple objects with or without holes (Figure 2). Some important features of this descriptor are:

- ○ It gives a compact and efficient way of describing properties of multiple disjoint regions simultaneously,
- ○ It can cope with errors in segmentation where an object is split into disconnected subregions, provided that the information which subregions contribute to the object is available and used during the descriptor extraction,
- ○ The descriptor is robust to segmentation noise.

The information from CLD and RS (the region shape) descriptors is stored in XML Metadata Interchange format, which is further processed by the classifier.The feature space for shape retrieval and classification consisted of up to 47 coefficients (12 from CLD and 35 from ART descriptors) and they were passed further on to the classifier after the selection scheme based on Fuzzy C-Means.

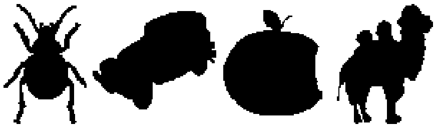

Figure 2: Example of shapes where region based shape is applicable.

2.3 Video stream segmentation and RFrames selection

This initial phase is supposed to select the images from our simple video recordings that are to be used for shape understanding. A simple algorithm, was chosen, (also [9]): for example, if a logical video segment is v_{10-100}, and the Rframe set from the whole video is {v1, v40, v75, v120...}, then {v40, v75} can be used to visually represent the segment.

In order to have a clear understanding of how gesture's RFrames are selected a simple figure (3) shows a relative smooth transition from neutral phase (where signer keep hands down) to the active phase (the region in green from the middle) and again in the neutral position.In this selection process, the images are represented in the Principal Components (PC) space, [4]. The representation in PC-space has previously revealed us very interesting aspects of motion's dynamics, [11],[18]. Since we are not dealing with crisp transitions (ie an image may belong to 2 subsets), I considered mandatory to use Fuzzy Clustering.

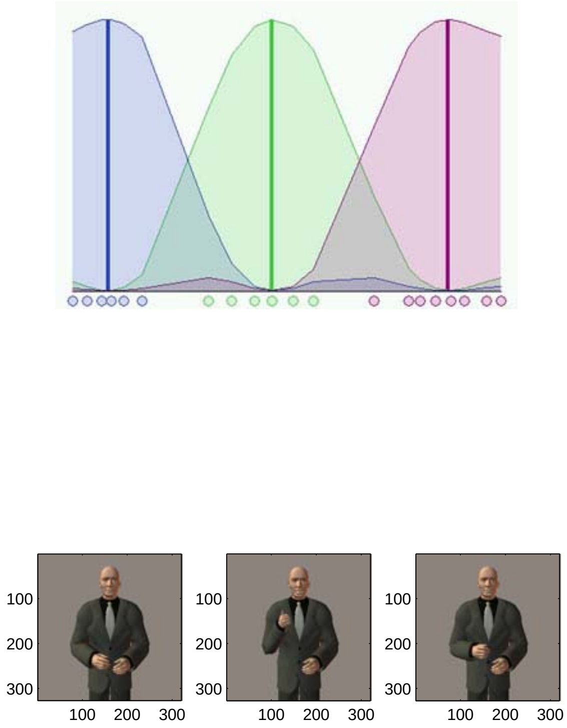

Figure 3: The figure shows how the images in a videostream can be clustered in 3 simple regions, therefore Fuzzy C-Means is applicable

The basics of the algorithm, called Fuzzy C-Means (FCM) were introduced by Dunn [5] and improved by Bezdek, [6] a classic fuzzy clustering algorithms. The objective function for FCM is given by:

Table 1:

```
Fix number of the clusters C;
Fix the fuzzifier m;
Do {
    Update membership using equation (4)
    Update center using equation (5)
} Until (center stabilize)
```

$$J = \sum_{i=1}^{C} \sum_{j=1}^{N} (u_{ij})^m d^2(x_j, c_i) \qquad (3)$$

where $u_{ij} \in [0,1]$ for all *i* and obeys the constraints defined in the equation below:

$$\sum_{i=1}^{C} u_{ik} = 1, 1 \leq k \leq n$$

$$0 < \sum_{k=1}^{n} u_{ik} < n, 1 \leq i \leq c$$

where *C* is the number of clusters (3 as explained above) and *d* is the distance norm (like Euclidean, Manhattan aso). The minimization of the objective function with respect to membership values leads to the following:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} (\frac{d^2(x_j,c_i)}{d^2(x_j,c_k)})^{\frac{1}{m-1}}} \qquad (4)$$

And the minimization of the objective function with respect to the center of each cluster will gives us:

$$c_i = \frac{\sum_{j=1}^{N} (u_{ij})^m x_j}{\sum_{j=1}^{N} (u_{ij})^m} \qquad (5)$$

In the equation above $m \in [1, \infty]$ is the fuzzifier. Therefore, in the end the algorithm looks like in the Table 1.

2.4 Short introduction to SVM

In the current framework of Machine Learning and data understanding we considered the approach derived from statistical learning theory of SVM (support vector machines).

Suppose we are given a set of examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)$, $\mathbf{x}_i \in X, y_i \in \{\pm 1\}$ and we assume that the two classes of the classification problem are *linearly separable*.

Theorem 1 *Let the l training set vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l \in X$ (X is the dot product space) belong to a sphere $S_R(\boldsymbol{a})$, of diameter D, and center at $\boldsymbol{a}$, i.e. $S_R(\boldsymbol{a}) = \{\boldsymbol{x} \in X : \|\boldsymbol{x} - \boldsymbol{a}\| < \frac{D}{2}\}, \boldsymbol{a} \in X$. Also, let $f_{\boldsymbol{w},b} = \mathrm{sgn}\big((\boldsymbol{w} \cdot \boldsymbol{x}) + b\big)$ be canonical hyperplane decision functions, defined on these points. Then the set of Δ-margin optimal separating hyperplanes has the VC-dimension h bounded by the inequality*

$$h \leq \min([D^2/\Delta^2], n) + 1 \qquad (6)$$

where $[x]$ denotes the integer part of x.

In this case, we can find an optimal weight vector $\mathbf{w}_0$ such that $\|\mathbf{w}_0\|^2$ is minimum (in order to maximize the margin $\Delta = \frac{2}{\|\mathbf{w}_0\|}$ of Theorem 1) and $y_i \cdot (\mathbf{w}_0 \cdot \mathbf{x}_i + b) \geq 1,\ i = 1, \ldots, l$.

The support vectors are those training examples that satisfy the equality, i.e. $y_i \cdot (\mathbf{w}_0 \cdot \mathbf{x}_i + b) = 1$. They define two hyperplanes. The one hyperplane goes through the support vectors of one class and the other through the support vectors of the other class. The distance between the two hyperplanes is maximized when the norm of the weight vector $\|\mathbf{w}_0\|$ is minimum. This minimization can proceed by maximizing the following function with respect to the variables α_i (Lagrange multipliers) [20]:

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \cdot \alpha_j \cdot \big(\mathbf{x}_i \cdot \mathbf{x}_j\big) \cdot y_i \cdot y_j \qquad (7)$$

subject to the constraint: $0 \leq \alpha_i$. If $\alpha_i > 0$ then $\mathbf{x}_i$ corresponds to a support vector. The classification of an unknown vector **x** is obtained by computing

$$F(\mathbf{x}) = sgn\{\mathbf{w}_0 \cdot \mathbf{x} + b\},\ where\ \mathbf{w}_0 = \sum_{i=1}^{l} \alpha_i \cdot \mathbf{y}_i \cdot \mathbf{x}_i \qquad (8)$$

and the sum accounts only $N_s \leq l$ nonzero support vectors (i.e. training set vectors $\mathbf{x}_i$ whose α_i are nonzero). Clearly, after the training, the classification can be accomplished efficiently by taking the dot product of the optimum weight vector $\mathbf{w}_0$ with the input vector **x**.

$$minimize\ \frac{1}{2}\mathbf{w} \cdot \mathbf{w} + C \cdot \sum_{i=1}^{l} \xi_i \qquad (9)$$

The case that the data is not linearly separable is handled by introducing slack variables $(\xi_1, \xi_2, \ldots, \xi_l)$ with $\xi_i \geq 0$ [21] such that, $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, i = 1, \ldots, l$. The introduction of the variables ξ_i, allows misclassified points, which have their corresponding $\xi_i > 1$. Thus, $\sum_{i=1}^{l} \xi_i$ is an upper bound on the number of training errors. The corresponding generalization of the concept of optimal separating hyperplane is obtained by the solution of the optimization problem given by equation (9) above, subject to:

$$y_i \cdot (\mathbf{w} \cdot \xi_i + b) \geq 1 - \xi_i \; and \; \xi_i \geq 0, i = 1, \ldots, l \quad (10)$$

The control of the learning capacity is achieved by the minimization of the first term of (9) while the purpose of the second term is to punish for misclassification errors. The parameter C is a kind of regularization parameter, that controls the tradeoff between learning capacity and training set errors. Clearly, a large C corresponds to assigning a higher penalty to errors.

Finally, the case of nonlinear Support Vector Machines should be considered. The input data in this case are mapped into a high dimensional feature space through some nonlinear mapping Φ chosen a priori [20]. The optimal separating hyperplane is then constructed in this space. As shown at 5.5 in [13] the corresponding optimization problem is obtained from (7) by substituting **x** by its mapping $\mathbf{z} = \Phi(\mathbf{x})$ in the feature space, i.e. is the maximization of $W(\alpha)$:

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \cdot \alpha_j \cdot (\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)) \cdot y_i \cdot y_j$$

subject to:

$$\sum_{i=1}^{l} y_i \cdot \alpha_i = 0$$
$$\forall i : 0 \leq \alpha_i \leq C$$

2.5 The proposed classifier

The number of training examples is denoted by *l*.In our case *l* was always 280 (10 frames for each of the 28 one-handed gestures of ISL). α is a vector of *l* variables, where each component α_i corresponds to a training example $(\mathbf{x}_i, y_i)$. $\mathbf{x}_i$ represents the features vector which is formed by either 35 (only the Region-shape coefficients) or 47 coefficients corresponding to both the Region-shape (RS) and the CLD descriptors. A fast Windows implementation (.dll's) for the extraction of the video descriptors was chosen, ([17]) that can be included in our final real-time Sign-Language understanding system. Although there are many available implementations in several programming languages (like Matlab, C++, Java, Lisp aso), I have used a Java version ([15]) of an implementation of the Support Vector Machine called mySVM developed by Stefan Rüping. It is based on the optimization algorithm of SVM^{light} as described in [16]. mySVM can be used for pattern recognition, regression and distribution estimation. In order to cope with the relatively small number of examples, a cross-validation (see, [19] with a factor of 25 was chosen. Several types of kernels were tested (neural, polynomial, Anova, epanechnikov, gaussian-combination, multiquadric, based on radial-basis functions) but, our experience [10] is once more confirmed, that (most probably) the SVM-classifiers based on polynomial kernels are the best for classification problems. Therefore, all the results expressed in the table 2 correspond to the polynomial-kernel supervised learning.

Table 2: Generalization Performance of mySVM classifier with polynomial kernels.

Current Experiment	Performance Vector	Description of the experiment
RS1	6.03%	Region-shape only
RS1 & CLD1	6.39%	Region-shape and CLD coeffs real images
RS2	7.64 %	Region-shape coeffs for real and virtual signers

3 Experimental results

The experience acquired in the group has shown that there are many factors that can influence the quality of image understanding, like: the differences between signers clothing, between the lighting sources, between the skin of the humans or due to the motion blur. Therefore the first step was to compare the classification performance of our algorithm for two classes of input data, only 35 coefficients (of the RS-descriptor), or 47 coefficients (of the RS and CLD descriptors). The performance vector (overall) resulted from the confusion matrix of the results is presented in the table 2, and it shows that by adding the 12-extra coefficients corresponding to CLD, the classification error is only slightly increased (by 0.35%). That will allow to gather more information in our training and testing database (more real and virtual signers) and to quantify the differences enumerated above in only few coefficients at virtually no expenses. The second step of our experiment used the same number of images but, for the same letter/ sign expressed were used ten static images and ten images extracted from the Poser-generated video-stream- corresponding to the same sign, like in the Figure 1. The performance is given in the third line of the table.

4 Conclusions

The results explained in the previous sections show that a limited of gestures (executed by a human or a robot...) can be learned and understood by combining the shape recognition (hand shapes playing the role of letters in an alphabet) detailed in the current work with an understanding of the gesture dynamics represented in the feature space (like PCA). In this latter approach, the images are represented in the PCA-space and the gestures are represented in a nonlinear manifold, see [18]. The fast procedure exposed- it takes approximately 10 msec (average classification time), and approx. 1 second for the VDE-feature extraction on a PC (2.4 GHz) it's considered to be a good choice for other researchers in the field.

Acknowledgements:
The research was supported by the Science Foundation of Ireland (SFI) and the author is also thankful to Dr. Sara Morrisey for the database of synthetic video streams and to Lecturer Alistair Sutherland (both from Dublin City University)- for being a good group leader.

References:

[1] Poser official site by e-Frontier America Inc : http://www.e-frontier.com/.

[2] Junwei Han, George Awad, Alistair Sutherland, and Hai Wu, Automatic Skin Segmentation for Gesture Recognition Combining Region and Support Vector Machine Active Learning, *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 237–242.

[3] George M. Awad, A Framework for Sign Language Recognition using Support Vector Machines and Active Learning for Skin Segmentation and Boosted Temporal Sub-units, PhD Thesis, Dublin City University, 2007.

[4] I. T. Jolliffe, Principal Component Analysis, Springer Verlag, 2002.

[5] J. C. Dunn, *A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters*, Cybernetics and Systems: An International Journal, 1087-6553, Volume 3, Issue 3, 1973, pp. 32 - 57.

[6] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981

[7] J. Kovac, P. Peer and F. Solina, Human Skin Colour Clustering for Face Detection, *Proc. of EUROCON 2003*, Finland, pp. 144–148.

[8] http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-7_schema_files/

[9] Anupam Joshi, Sansanee Auephanwiriyakul, and Raghu Krishnapuram, On Fuzzy Clustering and Content Based Access to Networked Video Databases, *Proceedings of the Workshop on Research Issues in Database Engineering*, Page: 42–47 ,1998.

[10] S. Papadimitriou, L. Vladutu, S. Mavroudi and A. Bezerianos, *Ischemia Detection with a Self-Organizing Map Supplemented by Supervised learning*, IEEE Transactions on Neural Networks 12 (2001), 503–515.

[11] Wu Hai and Alistair Sutherland, Irish Sign Language Recognition Using Hierarchical PCA, *Irish Machine Vision and Image Processing Conference* (IMVIP 2001), National University of Ireland, Maynooth, 5–7 September 2001.

[12] G. Awad, J. Han and A. Sutherland, Hand and face tracking demo: http://www.youtube.com/watch?v=exbGdHpFiW0

[13] Vladimir Vapnik, *The Nature of Statistical Learning Theory*, 2nd Edition, Springer –Verlag, Berlin–Heidelberg–New York–Tokyo 2000.

[14] ”The Standard Dictionary of Irish Sign Language”, by microBooks Ltd. on DVD,2006 by NAD, www.deafhear.ie/documents/pdf/1951.pdf.

[15] Open-Source Data Mining with the Java Software RapidMiner, http://rapid-i.com/content/blogcategory/38/69/

[16] Joachims, Thorsten, Making large-Scale SVM Learning Practical, In *Advances in Kernel Methods - Support Vector Learning*, chapter 11, MIT Press, 1999.

[17] Giorgos Tolias, Visual Descriptor Applications, Semantic Multimedia Analysis Group- NTUA, Greece: http://image.ntua.gr/smag/tools/vde.

[18] L. Vladutu, A. Sutherland, Gesture Analysis of Deaf People Language using nonlinear analysis of manifolds, , July 2007, SFI Conference, Dublin, Ireland.

[19] Kohavi, Ron, ”A study of cross-validation and bootstrap for accuracy estimation and model selection”, Proceedings of the 14th International Joint Conference on Artificial Intelligence 2 (12): 1137-1143,(Morgan Kaufmann, San Mateo),1995.

[20] V. N. Vapnik, Statistical learning theory, Wiley-Interscience, 1998.

[21] C. Cortes and V. Vapnik, *Support vector networks*, Machine Learning 20:3,1995, pp. 273–297.