Non-rigid Shape Recognition for Sign Language Understanding

LIVIU VLADUTU
School of Computing
Dublin City University
Glasnevin, Dublin 9, DCU
IRELAND
lvladutu@computing.dcu.ie http://www.computing.dcu.ie

Abstract: - The recognition of human activities from video sequences is currently one of the most active areas of research because of its many applications in video surveillance, multimedia communications, medical diagnosis, forensic research and sign language recognition. The work described in this paper describes a new method designed to precisely identify human gestures for Sign Language 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 Human-Computer interaction (HCI) projects in general and Irish Sign Language (ISL) understanding in particular. 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' gestures, recorded in video-streams, where only an individual gesture is executed, not linked to the consecutive signs (as in a normal sign-language conversation).

Vision-based gesture recognition techniques have received a lot of attention in the recent years, and controlling and understanding gestures are the focus of current research in vision-based interfaces (VBI), and there are a lot of possible applications:

- television control, [021] ;
- music synthesis,[23];
- robot control [27];
- surgeon's computerized aid, [22];
- video surveillance ;
- forensic research;
- hand-gestures interfaces for smart phones;
- smart interfaces,[25];
- home automation & medical monitoring, like Gesture Pendant, [28] etc.

1.1 Main steps

The classical steps of this type of human-computer interaction (HCI) system that were implemented by members of the team (see also [2]) 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 or .mexw32 programs in Matlab 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 point of view of the human motion characteristics. It has the following main steps:

1) Video-stream analysis of the whole video-stream (corresponding to a gesture) using Principal Component Analysis (based on Singular Value Decomposition) in order to extract the representative frames (using Fuzzy-Logic);
2) Feature extraction using novel features (Colour Layout Descriptor and the Region shape one developed by MPEG-7 group);
3) shape classification using Computational Intelligence methods (based on Support Vector Machine classifiers).

The detection of subunits and the skin segmentation are hampered by a series of inevitable natural factors, some of them being enumerated below:

- Different skin colour of the signers;
- differences in illumination conditions;
- different anatomic characteristics of the humans;
- differences in distances and angles of camera location from the signer;
- different temporal execution of the gestures (influenced by signer's mood or temperament);
- differences between the ways a man / woman executes the gestures.

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 [1][6] is based on the assumption that skin colour is quite different from colours of other objects and it's distribution might form a cluster in some specific colour-spaces (RGB, YCbCr). 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 the work described in [10], influenced by new insights of the machine learning and statistical learning theory[11],[18].
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 handy-cam, but also a proprietary database of synthetic images (using virtual signers) created using Poser (http://www.e-frontier.com) and the Python programming language.

2 The proposed approach

The skin model

The current work was based on a large experience in skin-model detection ([1][6] 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 for instance:
http://www.youtube.com/watch?v=exbGdHpFiW0,
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”, [12].

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 database of images created using Python and Poser. 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. Moving Picture Experts Group is the committee that also developed the Emmy Award winning standards known as MPEG-1 and MPEG-2, and the MPEG-4 standard. An extended description is also at:
http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm

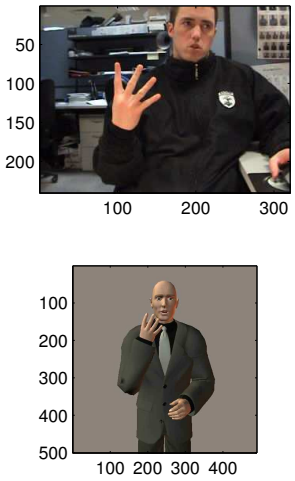


Figure 1: Example of an image from real signer (above) and the equivalent one from a Poser-generated video (below)

Two MPEG-7 visual descriptors are used in the experiments. The Colour Layout descriptor (CLD) provides information about the spatial colour 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:

$$S_{CLD}(Q,I)=\sqrt{\sum_i w_{y_i}(Y_{Q_i}-Y_{I_i})^2}+\sqrt{\sum_i w_{Cb_i}(Cb_{Q_i}-Cb_{I_i})^2}+\sqrt{\sum_i w_{Cr_i}(Cr_{Q_i}-Cr_{I_i})^2}$$

where w_i represents the weight associated with coefficient *i*. There are 12 coefficients extracted for the colour 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, see [7] and: http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-7_schema_files/. 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:

$$F_{nm}=<V_{nm}(\rho,\theta)-f_{nm}(\rho,\theta)>$$
$$=\int_0^{2\pi}\int_0^1 V_{nm}^*(\rho,\theta)f_{nm}(\rho,\theta)\rho d\rho d\theta$$

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:

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

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

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 sub regions, provided that the information which sub regions contribute to the object is available and used during the descriptor extraction,
- The descriptor is robust to segmentation noise.

Also the classification with this descriptors outperforms the results obtained with other descriptors, like SIFT, described in [30] and [32]. The information from CLD and RS (the region shape) descriptors is stored in XML (Extensible Markup Language) a 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 selectionSubsection

This initial phase is supposed to select the images from our simple video recordings that are to be used for shape understanding. First of all, the stream of images is represented in the PCA-space, like in the figure 3 below:

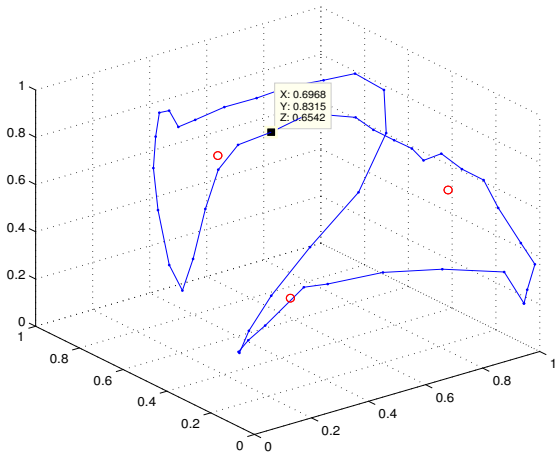
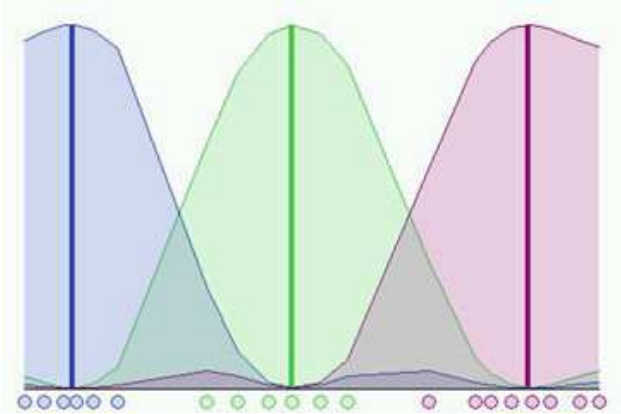


Figure 3: A representation of a video-gesture 'a' in the 3-dimensional PCA-space

In the figure above, there are 3 clusters and their centers (represented as red circles) and we selected the frames (RFrames) from the middle-cluster (the one from the left hand of the picture). The manifold is a closed one, since, the signer's hands are starting and ending in the same position, as emphasized in the Figure 4.

A simple algorithm, was chosen, (see also [8]): for example, if a logical video segment is v_{10-100}, and the Rframe set from the whole video is $\{v_1, v_{40}, v_{75}, v_{120}, \ldots\}$, then $\{v_{40}, v_{75}\}$ 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 4 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.



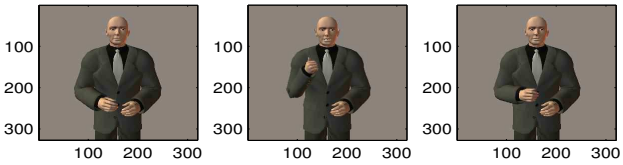


Figure 4: The figure shows how the images in a video stream can be clustered in 3 simple regions, therefore Fuzzy C-Means is applicable

In this selection process, the images from the all gesture's video stream, are represented in the Principal Components (PC) space,[3]. The representation in PC-space has previously revealed us very interesting aspects of motion's dynamics, [10][16].

Since we are not dealing with crisp transitions (i.e. an image may belong to 2 subsets), I considered mandatory to use Fuzzy Clustering.

The basics of the algorithm, called Fuzzy C-Means (FCM) were introduced by Dunn [4] and improved

by Bezdek, [5] a classic fuzzy clustering algorithms.

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

Table 1

The objective function for FCM is given by the function *J*:

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

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

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

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

where *C* is the number of clusters (3 as explained above) and *d* is the distance norm (like Euclidean, Manhattan etc). I have used as validity measure for fuzzy clustering, the Xie-Beni index [20], which for $C \in [2,10]$, recommends $m \in [1.5, 2.5]$. We have *C=3,* so the choice *m=2* for the fuzzifier gave the best results.
The minimization of the objective function *J* with respect to membership values leads to the following:

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

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 (m=2 in this case).
Therefore, in the end the algorithm looks like in the pseudo-code description depicted in Table 1.

2.4 Short introduction to Support-Vector Machines based learning

Machine learning in general is a scientific discipline that is concerned with the design and development of algorithms that allow computers to learn based on data, such as from sensor data (data derived from images in our case) or databases. In the current framework of Machine Learning and data understanding I have considered the approach derived from statistical learning theory of SVM (support vector machines). Suppose we are given a set of examples $(x_1, y_1), (x_2, y_2), \ldots (x_l, y_l), x_i \in X, \; y_i \in \{\pm 1\}$ and we assume that the two classes of the classification problem are *linearly separable*.

Theorem 1: *Let the "l" training set vectors* $x_1, x_2, \ldots, x_l \in X$ *(*X *is the dot product space) belong to a sphere* Sr(***a***)*, of diameter* D*, and center at* ***a****, i.e. :*

$$SR(a) = \left\{x \in X, \|\mathbf{x} - \mathbf{a}\| < \frac{D}{2}\right\}, \mathbf{a} \in X$$

Also, let $f_{w,b} = \mathrm{sgn}((\mathbf{w} \cdot \mathbf{x}) + b)$ *be canonical hyper-plane decision functions, defined on these points. Then, the set of* Δ *-margin optimal separating hyper-planes has the VC-dimension* h *bounded by the inequality:*

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

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

In this case, we can find an optimal weight vector w_0 such that $\|w_0\|^2$ is minimum (in order to maximize the margin $\Delta = \frac{2}{\|\mathbf{w}_0\|}$ of Theorem 1 above and

$$y_i \cdot (\mathbf{w}_0 \cdot \mathbf{x}_i + b) \geq 1,\ i = 1,...,l$$

The support vectors are those training examples situated on the boundary (between the 2 classes) that satisfy the equality:

$$yi \cdot (\mathbf{w}_0 \cdot \mathbf{x}_i + b) = 1,\ i = 1,...,l$$

They define two hyper-planes. The one hyper-plane goes through the support vectors of one class and the other through the support vectors of the other class.

The distance between the two hyper-planes is maximized when the norm of the weight vector $\|w_0\|$ is minimum.

This minimization can proceed by maximizing the following function with respect to the variables α_i (Lagrange multipliers) [18]:

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

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:

$$F(\mathbf{x}) = \mathrm{sgn}\{\mathbf{w}_0 \cdot \mathbf{x} + b\}, \text{ where:}$$

$$w_0 = \sum_{i=1}^{l} \alpha_i \mathbf{y}_i \mathbf{x}_i \quad (8)$$

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**.

The case that the data is not linearly separable is handled by introducing slack variables $\xi_1, \xi_2, ..., \xi_l$ with $\xi_i \geq 0$ (see also [19]) such as:

$$yi(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i,\ i = 1,2,...,l$$

The idea can be expressed in a formal way as: the goal is to:

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

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 hyper-plane is obtained by the solution of the optimization problem given by equation (9) above, subject to:

$$yi(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i,\ i = 1,2,...,l$$
$$and \quad \xi_i > 0 \quad (10)$$

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 trade-off 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 [18].

The optimal separating hyper-plane is then constructed in this space. As shown in Chapter 5 in [11] 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:

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

2.4 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, ([15]) 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 a.s.o.), I have used a Java version ([13]) of an implementation of the Support Vector Machine called mySVM developed by Stefan Rüping. It is based on the optimization algorithm of SVMlight as described in [14]. 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, [17]) 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 [9][24] 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 (the one with the smallest overall classification error for the 3 experiments described in Table 2) supervised learning. In the graph from Figure 4 below are described the classification errors (on ordinate) for 4 of the most commonly used kernels;

- Anova;
- polynomial;
- radial;
- dot.

The SVM with regular kernels as above are expressed by the following equations:

●linear SVM: $\psi(x, x_k) = \left\langle x_k^T, x \right\rangle$

●the polynomial SVM of degree *d:*

$$\psi(x, x_k) = (a\left\langle x_k^T \cdot x \right\rangle + 1)^d$$

●the RBF (Radial Basis-Function) SVM:

$$\psi(x, x_k) = \exp\{-\sigma \left\| x - x_k \right\|_2^2\}$$

●the Laplacian kernel:

$$\psi(x, x_k) = \left\langle x, x_k^T \right\rangle \exp\{-\sigma \left\| x - x_k \right\|_2^2\}$$

as expressed in an excellent new reference: see [35].

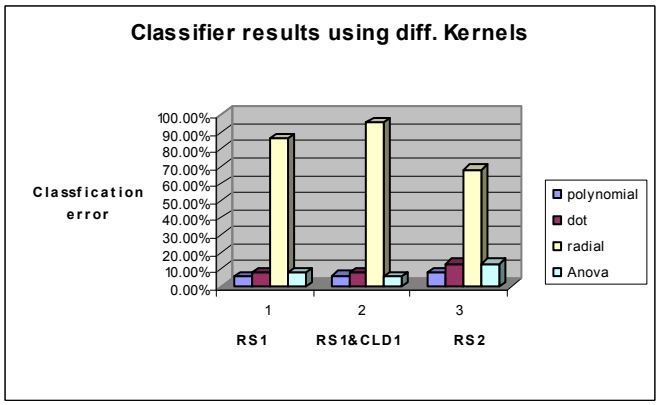


Figure 5: Classification performance of SVM with some of the most common kernels

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 2.

Implementations were built around Matlab (Mathworks ©) which is a powerful matrix-based software package having a lot of toolboxes, and which was used as a 'wrapper'.

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 coefficients real images
RS2	7.64 %	Region-shape coefficients for real and virtual signers

3 Conclusion

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. The fast procedure exposed- it takes approximately 10 milliseconds (average classification time), and approximately 1 second for the VDE-feature extraction on a PC (having dual-2.4 GHz processor) it's considered to be a good choice for other researchers in the related fields, which are enumerated in the Introduction section.

Future envisaged work involves:

- background modelling at the beginning of the video-frame analysis, [29];
- eventually tackling occlusion problems with the help of a-priori defined 3D models of the involved body parts (head and hands), using some of the available software environments, either:

● ITK/VTK (http://www.vtk.org/, [31] and http://www.itk.org/) ;

● using Computational Geometry Algorithms Library (http://www.cgal.org/) or related, see also [33], [36].

Acknowledgements:
The research was supported by the Science Foundation of Ireland –SFI (to whom I am deeply thankful) but the author is also thankful to:
- Lecturer Alistair Sutherland, Dr. George Awad, Dr. Junwei (Jeff) Han for the SST (Skin-segmentation and tracking) contribution;
- Dr. Sara Morrisey and Tommy Coogan for the database of synthetic video streams generated in Poser (all from Dublin City University) and, respectively, for the camera collected video-streams.

References:

[1] J. Han, G. Awad, A. Sutherland and H. Wu, Automatic Skin Segmentation for Gesture Recognition Combining Region and Support Machine Active Learning, *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 237-242.

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

[3] I. T. Jolliffe, *Principal Component Analysis,* Springer-Verlag, 2002.

[4] J.C. Dunn, A Fuzzy Relative to the ISODATA Process and It's Use in Detecting Compact Well-separated Clusters, *Cybernetics and Systems: An International Journal,* Vol. 3, Issue 3, 1973, pp. 32-57.

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

[6] J. Kovac, P. Peer and F. Solina, Human Skin Colour Clustering for Face Detection, *Proceedings of EUROCON 2003, Turku, Finland,* pp. 144-148.

[7] Shih-Fu Chang, T. Sikora and A. Puri, Overview of the MPEG-7 Standard, *IEEE Transactions on Systems and Circuits for Video Technology,* Volume 11, No. 6, June 2001, pp. 688-695.

[8] A. Joshi, S. Auephanwiriyakul and R. Krishnapuram, On Fuzzy Clustering and Content Based Access to Content Video Databases, *Proceedings of the Workshop on Research Issues in Databases Engineering,* 1998, pp. 42-47.

[9] S. Papadimitriou, S. Mavroudi, L. Vladutu and A. Bezerianos, Ischemia Detection with a Self-Organizing Map Supplemented by Supervised

Learning, *IEEE Trans. on Neural Networks,* Volume 12, Issue 3, pp. 503-515.

[10] W. Hai and A. 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.

[11] V. Vapnik, *The Nature of Statistical Learning Theory,* 2nd Edition, Springer Verlag, 2000.

[12] The National Association for Deaf People, Ireland, *The Standard Dictionary of Irish Sign Language, (CD/DVD),* by microBooks Ltd., 2006.

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

[14] Joachims Thorsten, Making Learning Large-Scale SVM Learning Practical, *Advances in Kernel Methods, chapter 11,* MIT Press, 1999.

[15] G. Tolias, Visual Descriptors Applications, Semantic Multimedia Analysis Group, NTUA, Athens, Greece, http://image.ntua.gr/smag/tools/vde/

[16] L. Vladutu, A. Sutherland, Gesture analysis of deaf people language using nonlinear manifolds analysis, *SFI Conference,* Dublin, Ireland, July, 2007, presented by. A. Sutherland.

[17] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, *Proceedings of the14th International Joint Conference on Artificial Intelligence, 2(12),* Morgan Kaufmann, San Mateo, 1995, pp. 1137-1143.

[18] V. N. Vapnik, *Statistical Learning Theory,* Wiley-Interscience, 1998.

[19] C. Cortes and V. Vapnik, Support Vector Networks, *Machine Learning,* Volume 20, Number 3, September 1995, pp. 273-297.

[20] X. L. Xie, G. Beni, A Validity Measure for Fuzzy Clustering, *IEEE Trans. Pattern Analysis and Machine Intelligence,* Volume 13, Number 8, 1991, pp. 841-847.

[21] W. T. Freeman and C. Weisman, Television Control by Hand Gestures, *International Workshop on Automatic Face and Gesture Recognition*, IEEE Computer Society, Zurich, Switzerland, June 1995, pp. 179-183.

[22] C. Graetzel, S. Grange, T. Fong and C. Baur, A non-contact mouse for Surgeon-Computer Interaction, *Technology and Health Care, IOS Press,* Volume 12, Number 3, 2004, pp. 245-257.

[23] J. Carreira and P. Peixoto, A Vision Based Interface for Local Collaborative Music Synthesis, *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, FGR 2006, pp. 591-596.

[24] L. Vladutu, *Computational Intelligence Methods on Biomedical Signal Analysis,* VDM-Verlag Publishing House, 2009.

[25] J. Krumm, S. Shafer and A. Wilson, How a Smart Environment Can Use Perception, *Workshop on Sensing and Perception, (part of ACM UbiComp 2001),* September 2001.

[26] Y. Wu and T. Huang, Vision-Based Gesture recognition: A Review, *Lecture Notes in Computer Science,* Springer Verlag, Volume 1739, 1999, pp. 103-115.

[27] A. Corradini and H-M Gross, Camera-Based Gesture Recognition for Robot Control, *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, (IJCNN 2000),* Como, Italy, 2000, pp. IV 133-138.

[28] T. Starner, J. Auxier, D. Ashbrook and M. Gandy, The Gesture Pendant: A Self-Illuminating, Wearable, Infrared Computer Vision System for Home Automation Control and Medical Monitoring, *Proceedings of the4th IEEE International Symposium on Wearable Computers, ISWC 2000,* pp. 87-94.

[29] D. Gutchess, M. Trajkovics, E. Cohen-Solal, D. Lyons and A.K. Jain, A background model initialization algorithm for video surveillance, *Proceedings of the 8th IEEE International Conference on Computer Vision, ICCV 2001*, Volume 1, July 2001, pp. 733-740.

[30] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision,* Volume 60, Number 2, November 2004, pp. 91-110.

[31] B. Preim, D. Bartz, *Visualization in Medicine: Theory, Algorithms and Applications,* The Morgan Kaufmann Series in Computer Graphics, July 2007.

[32] D. G. Lowe, Object Recognition from Scale-Invariant Features, *IEEE 7th International Conference on Computer Vision, (ICCV '99),* Volume 2, 1999, pp. 1150-1156.

[33] A. Fabri, G-J. Giezeman, L. Kettner, S. Schirra, On the Design of CGAL, a computational geometry algorithms library, *Software- Practice and Experience,* Volume 30, Issue 11, pp. 1167-1202.

[34] M. Bober, MPEG-7 Visual Shape Descriptors, *IEEE Transactions on Circuits and Systems for Video Technology,* Volume 11, Number 6, June 2001, pp. 716-719.

[35] S. Amiri, D. von Rosen, S. Zwanzig, The SVM Approach for Box-Jenkins Models, *REVSTAT*

Statistical Journal, Volume 7, Number 1, April 2009, pp. 23-26.

[36] J. E. Goodman and J. O'Rourke, *Handbook of Discrete and Computational Geometry, 2nd Edition,* Chapman & Hall/ CRC, 2004.