O.G.R.E. – Open Gestures Recognition Engine

José Miguel Salles Dias[1] Pedro Nande[1] Nuno Barata[1] André Correia[1]

[1] *ADETTI/ISCTE, Associação para o Desenvolvimento das Telecomunicações e Técnicas de Informática, Edifício ISCTE, 1600-082 Lisboa, Portugal, www.adetti.pt*
{Miguel.Dias, Pedro.Nande, Nuno.Barata, Andre.Correia} @iscte.pt

Abstract

In this paper, we describe a hand gesture recognition engine based on Computer Vision (CV), as a computing platform to support gesture interaction between humans and computers. Presenting a simple approach to recognizing gestures through image processing techniques and a single video camera, we address the problem of generic hand gestures recognition, especially of spelled Sign Language hand poses, introducing a preliminary study to its kinetic component. In our methodology, the system initially removes the background of captured images, eliminating irrelevant pixel information. The human hand is then detected, segmented and its contours localized. From these contours significant metrics are derived, allowing a search in a pre-defined hand poses' library, where each pose is previously converted into a set of metric values. We discuss several algorithmic options to support our methodology and present experimental results, regarding the recognition of Portuguese Sign Language signs. We discuss future directions of our work.

1. Introduction

When a new computing technology is deployed there is always the concern of how it will interact with the final user, including users with special needs. The ideal form of interaction would be to have a natural way of accessing the functionalities of that technology, without the need for specific peripheral devices. This would mean to communicate with a computer as we interact with other humans. There has been a slow evolution in this subject but some new solutions have shown relevant improvements and have demonstrated that it is possible to use different modalities, such as voice, gestures or other natural means as multiple and complementary ways to communicate with the computer. With our work we are aiming at developing an open source gestures recognition engine that would offer the possibility to trigger user-specified actions, activated by different hand gestures. The system requires a single video camera that captures the user's hand motion and recognizes a set of pre-defined hand poses or other types of gestures. Currently, only single hand gestures are supported.

In this paper we define four different types of hand gestures: "static hand poses", "simple hand path gestures", "staged hand path gestures" and "free hand tracking". A "static hand pose" is a gesture represented by a single hand pose with a spatial position that doesn't vary with time. As an example, it could well be a symbol of a sign language alphabet. A "simple hand path" is a gesture that describes a generic trajectory that can be compared to the sketching of a primitive shape. The analysis of this type of gestures consists in the evaluation of some hand's feature coordinates, such as the centre of mass of the hand and compare the generated sketch with a primitive shape. The "staged hand path" is a hybrid gesture composed by "static hand poses" and "hand paths". Finally, "free tracking" consists in evaluating the position of the hand while the user moves it freely. These types of hand gestures were developed as low level building blocks to be used in a higher layer of dynamic hand gesture recognition for a given human-computer interaction task. This is an approach to the introduction of new types of hand gestures, as these can ultimately be decomposed in such four different types. Any hand gesture that is recognized by the system can trigger actions in the application level, thus providing human-computer interaction.

In synthesis, our system recognizes a set of hand gestures based on CV that can be used in any type of final application that may require this type of human-computer interaction (HCI) modality. Envisaged applications are the ones dedicated to persons with hearing impairments, cognitive impairments, persons with mobility constraints (e.g., with a severe disease and lying on a bed in a Hospital), persons with problems in verbal communicating, or persons more oriented to natural forms of human-computer interaction, that disregard the dominant WIMP (Windows, Icons, Menu and Pointing) user interaction metaphor. Our engine has also potential applications in multimodal human-computer interaction, such as in the coordination of gestural and verbal constituents in deictic utterances, like in [17].

The paper is organized as follows: in section 2. Background, we provide a background in the issues of gesture data acquisition and recognition. In section 3, Gestures and Actions Definitions, we classify the type of recognised gestures and triggered actions. We then present our System Architecture in section 4. Section 5, Engine Modules, describes each of our implemented engine modules. Section 6, Results details the results achieved with various algorithmic alternatives in recognising shapes, in the context of the Portuguese Sign Language static hand pose symbols. Finally in section 7, Conclusions and Future Results, conclusions of the research done so far and future directions are given.

2. Background

In this section we will present an overview of the background in the domain of gesture recognition.

There are two different approaches when acquiring data for gesture recognition: one based on sensory devices and the other, a vision-based approach. The first requires research in some areas of Engineering, like Electronics, Mechanics, etc. and the second is based on Image Processing and Computer Vision research. There are also hybrid solutions that combine both fields, aiming towards a pragmatic solution.

2.1. Approach Based on Sensory Devices

This approach is focused in Mechanical, Electronics or Electrical Engineering research and suggests the use of physical devices that can measure the variations that occur while performing a gesture, such as position, linear acceleration, and orientation changing, performed by the hand. The gesture analysis is essentially mechanical and electromagnetic, consisting in the evaluation of these physical parameters. Due to the specificity, complexity and cost associated to this type of electronic/electromagnetic devices, the systems based on this approach are usually less scalable and expensive. Nevertheless, there are academic developments of these solutions [1, 2] and commercial applications with some acceptance in the market.

2.2. Approach Based on Computer Vision

Gesture recognition systems based on computer vision are subject of widely diffused research works. Although theoretical approaches tend to perspective user gestures in the larger and general problem of human motion, several results have focused on practicality and usability. Therefore, there are specific gesture-recognition systems of great interest. In [3, 4, 5] we find gesture recognition systems applied to useful, Sign Language interpretation. Other systems, like in [6], are more directed for human-computer interface control.

Recognition techniques, either image or vector based, all tend to acquire hand contour as a starting point for gesture perception and then its motion, being somewhat influenced by the human visual system process. Background scenario complexity is not frequently addressed, requiring gestures to be executed against a homogenous prepared scene. Related works from the computer vision domain, like advanced background subtraction [7], can be useful for gesture recognition.

Alternative approaches are based on mathematical descriptions of the whole captured image itself, as in [8], where Haar-like features are used to recognize trained objects, with an interesting potential for hand pose detection.

3. Gestures and Actions Definitions

3.1. Actions

In our systemic approach, based on the proposed O.G.R.E – Open Gestures Recognition Engine, actions are contextualized hierarchies of application dependant functionalities (requiring hand gesture recognition for Human-Computer Interaction - HCI or other purpose), which feed the gestures engine and provide a guide for available recognition algorithms selection.

Actions are defined in XML by the following notation:

```
<Action name="myFirstAction">
   <Start gesture_name=”myFirstGesture”></Start>
   <End gesture_name=”mySecondGesture”></End>
   <Actions>
         <Action name=”mySecondAction”></Action>
         <Action name=”myThirdAction”></Action>
         <Action name=”myForthAction”></Action>
   </Actions>
```

```
</Action>
```

Where `<Start>` and `<End>` are the gestures which trigger and terminate the action, and `<Actions>` are an optional list of child actions which can be accessed through the current context. The top hierarchy node name from which all actions derive is the reserved word `root`.

For the purpose of our system, we have also identified, as mentioned, four types of hand gestures, suitable for simple forward recognition, namely, Static Poses, Simple Paths, Staged Paths and Free Tracking.

3.2. Static Poses

Static poses are rigid hand postures which do not depend on the movement of the hand. They are characterized by a silhouette stored in a user-specific database.

A static pose is defined in XML by the following notation:

```
<StaticPose name="A" filename="A.bmp">
</StaticPose>
```

3.3. Simple Paths

Simple paths are sketched trajectories representing a given hand gesture kinetic motion, that correspond to simple primitive shapes, such as triangle, circle, square, or ellipse.

Simple paths are defined in XML by the following notation:

```
<SimplePath name="Circle"></SimplePath>
```

3.4. Staged Paths

Staged paths are vector based trajectories, similar to polylines, which depend on localized hand postures to define the control points that divide the gesture into several line segments.

Locally defined coordinate tolerance allows execution with error resilience robustness.

Staged paths are normalized for comparison purposes and are defined in XML, by the following notation:

```
<StagedPath name="my StagedPath">
   <StaticPose name="T" x="0" y="0
      tolerance="0.4"></StaticPose>
   <StaticPose name="Y" x="0.5" y="1"
      tolerance="0.4"></StaticPose>
   <StaticPose name="1" x="1" y="0"
      tolerance="0.4"></StaticPose>
</StagedPath>
```

3.5. Free Tracking

Free tracking is a gesture category of arbitrary hand movement, useful for HCI tasks and that can be subjected to post processing whenever hand coordinates need additional analysis.

4. System Architecture

Our O.G.R.E – Open Gestures Recognition Engine modular system architecture and process flow, is depicted in the next figure.

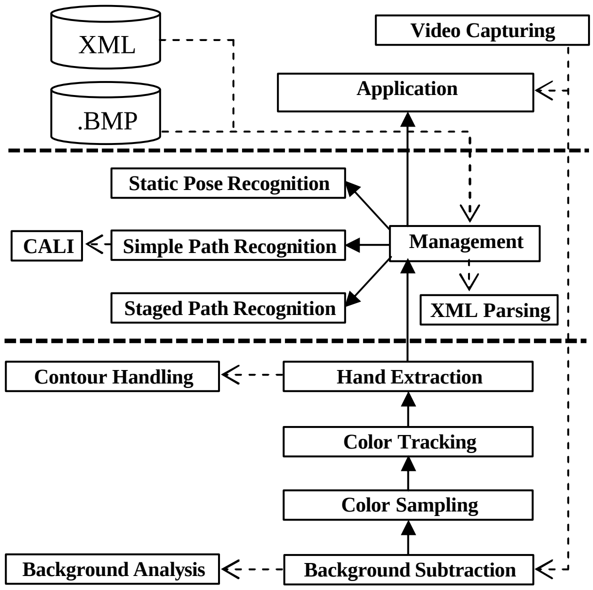

Figure 1 OGRE system architecture.

5. Engine Modules

5.1. Video Capturing

The video capturing module provides streaming media, both to the core engine and to the application layer. The raw images captured, in this way, can be processed separately by the engine and the application, if needed. This is an embedded module, which can be replaced by an outer video source, if advantages are found in such philosophy.

5.2. Background Subtraction

Background subtraction [7] is applied prior to any subsequent processing. It consists of a calibration period during which maximum and minimum per-pixel values in the YCrCb domain are stored and updated. After this initial period, foreground classification occurs, based on simple comparison between actual frame pixels' YCrCb values and the stored background model, since it is assumed that variations of actual frame pixels' YCrCb values below the stored minimum or above the stored maximum, classify these as foreground pixels (see Figure 2). This is based on a more widely used technique where the background pixels' average and standard deviation are first determined, and then compared with actual pixels' values in the RGB color space. However, tests have demonstrated that the standard deviation is an insufficient metric for aiding in adequate foreground classification, thus causing less quality results than the first method. Also, the YCrCb color space provides a better medium for this kind of analysis than the RGB model, due to its separate dependency between chrominances and luminance, which helps in segmenting contour transition areas.

5.3. Background Analysis

This module is responsible for background deterioration detection. It has been observed that the

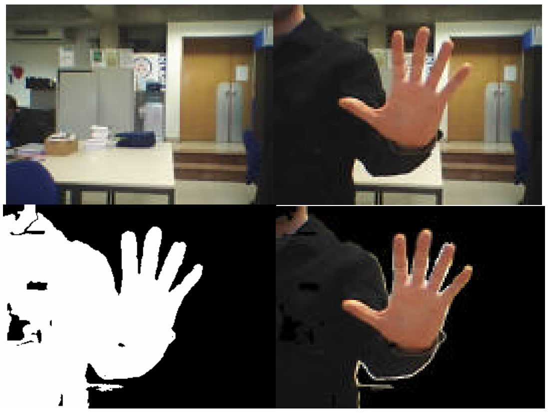

Figure 2 Subtracting the background. Top Left: the original background. Top Right: The user in foreground. Bottom Left: resulting background subtraction mask. Bottom Right: Subtracted background.

background subtraction algorithm used is not resilient to environmental changes, such as light fading, scene decorative objects replacement and camera positioning instability. Therefore, for long functioning periods, it is necessary to robustly adapt to new scene conditions without user intervention. The algorithm used for detecting such changes is based on a timed observation of the background subtraction mask. In normal conditions, this mask is a binary image composed of both black and white pixels, representing background and foreground regions respectively. Foreground regions, as typically observed during user interaction, are a reduced number of large connected white areas (about one or two spots of a quarter image area). When deteriorated, this mask will contain

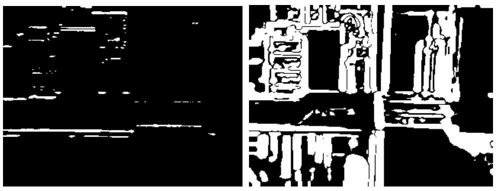

Figure 3 Background subtraction mask noise due to lighting changes (left) and camera positioning variation (right).

several smaller regions spread throughout expected locations (the edges of contrasting background elements, such as closets, doors, tables, etc), as depicted in Figure 3.

These spots can therefore be classified as noise, as they are unwanted, disruptive elements for both background and color segmentation. Their detection is done by finding small separated regions of 'white' value. For noise classification purposes, two measures are considered: 1). the maximum area of a spot, bellow which it is seen as noisy, and 2). the minimum noise coverage area, relative to the image's area, above which it is considered large enough to cause interference. A second type of noise is also observed when environmental conditions vary drastically (turning lights on or off, camera's field of view occlusion by large passing objects, etc). In all these cases there are no small spots visible, as the entire mask is constituted by a large white stain. This is also disruptive and therefore, considered noise. Its detection consists on simply establishing a maximum area for white value regions occupation, above which such elements are taken in consideration as noise.

Also, during calibration, as changes in background are decisively included in its statistical model, an in-depth observation is required. This is used to detect unwanted movements which can compromise a correct calibration. To do so, the model is analyzed in each consecutive update, checking for environment abnormalities. The background subtraction algorithm used is such that, at any moment during calibration, two image matrices exist: one containing per pixel minimum values sampled and the other its maximum values. In a stable environment, the difference between these matrices should be a homogenous grayscale image, representing small variations distributed equally in the entire image. In the presence of unwanted movement, a given area will show a peak in this differences matrix, if the interfering object (or other agent) is different enough from the surrounding scene. After thresholding the matrix, we obtain a binary image similar to the one analyzed in the search for noise. Thus, finding disturbances in the calibration process is a search for white value pixels covering an area large enough to be considered obtrusive.

In both algorithms, noise detection triggers a positive alarm. During normal interaction a time window of such alarms is analyzed. If these positive alarms appear in a considerable number, recalibration will occur automatically. During calibration, no time window is analyzed, as a unique positive alarm is sufficient to destroy the entire statistical model.

5.4. Color Tracking

Hue values obtained in the previous color sampling phase, feed the CAMSHIFT algorithm (Continuously Adaptive Mean Shift [9]), which is then applied to the current captured image. The CAMSHIFT algorithm computes a histogram back projection binary image, with reduced resolution, in respect to the initial image luminance, since the algorithm works in the sub-sampled chrominance domain, representing areas of a specified tonality (the hand tonality, in our case, which is a parameter of the system). It also reduces the hand pose searching area to the largest connected component representing the user's hand hue. Hand tracking is therefore guaranteed in the following frames.

5.5. Hand Extraction

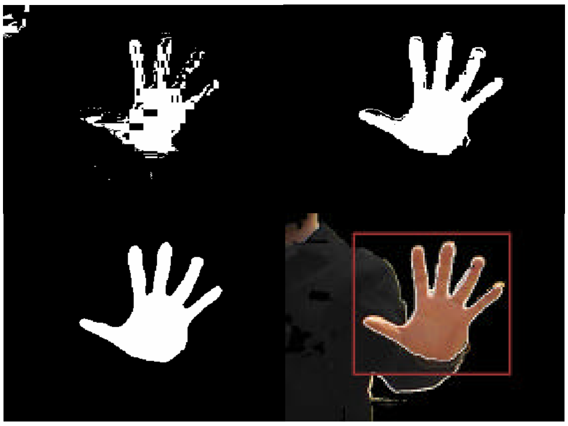

Figure 4 Extracting hand contour. Top left, the CAMSHIFT algorithm histogram back projection result. Top right: Result after YCrCb re-sampling. Bottom Left: result after morphological smoothing. Bottom Right: result after contour vectorisation and mixture with the foreground image.

The resulting hand contour image is still inadequate for static pose recognition. In order to extract a realistic hand silhouette, further processing is needed. The CAMSHIFT algorithm identifies the largest connected component‘s bounding box in the histogram back projection image, but it often covers an insufficient area of it's whole. A recursive algorithm is applied to determine the real bounding box, simply enlarging its sides by small amounts and checking if it is already large enough to cover the object's area. The resulting binary image is then used as a mask for YCrCb color space re-sampling, applied to the initial image with the background removed. With this, we intent to obtain richer information in the image domain, more suitable for edge extraction. Luminance and chrominances are sampled at different resolutions in order to achieve the best possible contour detail. With this process, we are able to determine the average of YCrCb pixel values of the hand region, and perform a better classification of pixels, belonging or not to the hand contour and interior. A smoothing morphological operation is then applied with an adequate structural filtering element for noise reduction. This element's dimensions can be of 5x5, 7x7, 9x9 or 11x11, depending in the estimated silhouette dimension.

5.6. Contour Handling

The extracted hand contour is converted to the vector form and, if necessary, a polygonal approximation sensitive to finger curvature is applied. This approximation is based on the best fit ellipse mathematical approach, as to obtain a measure of the curvature of a given set of points, proportional to the ellipse eccentricity. This module offers contour handling operations as to achieve an appropriate hand silhouette representation.

5.7. XML Parsing

This module is responsible for configuration, gestures and actions definitions parsing, as defined in a XML file.

5.8. Management

This module is the engine's core "intelligence". It analyses specific action context and redirects gesture recognition into the adequate system module: Static Pose Recognition, Simple Path Recognition or Staged Path Recognition.

5.9. Static Hand Pose Recognition

In order to recognize static poses, several widely known algorithms for shape analysis were studied. We can divide these algorithms into two categories, Image Based Analysis and Contour Based Analysis. We have studied the following:

Image Based Analysis:

Template Matching [9]: Based in the convolution between two images at several scales in order to find a given template.

Discrete Cosine Transform (DCT) Analysis: [4, 10, 11] A scale and rotation independent domain transformation in the frequency domain.

Contour Based Analysis:

Hu Moments [9]: These are a set of shape characteristic invariant metrics that can be useful for shape classification.

Pair-wise Geometrical Histogram (PGH) [12]: This method, computes the Histogram of distances and angles between the contour polygon's edges, which provides us with a unique contour signature.

Simple Shape Descriptors (SSD) [13, 14, 15, 16]: Combined simple geometrical metrics, which helps describing shapes.

PGH-SSD Hybrid: This method corresponds to the authors effort in combining PGH and SSD advantages.

For static Hand Pose Recognition, the extracted hand silhouette is compared against a library of silhouettes templates or a library of silhouettes signatures (depending in the method) using one of the above algorithms. In each case, proper formats for hand contour are used, since either we're dealing with image-based or vector-based contour analysis.

5.10. Simple Hand Path Recognition

The Simple Hand Paths recognition method uses C$_{AL}$I [14], a software library for calligraphic interfaces

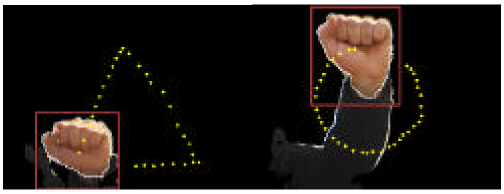

Figure 5 Sketching a triangle and a circle.

based on fuzzy logic. It has been used to recognize sketched shapes, including expected sketched shapes, on a different application context (hand sketching), but has an important application on our system as well. In fact, a moving hand describes a trajectory, whose projection on the image plane, can be compared to a drawn simple shape (such as triangle, circle, square or ellipse, see Figure 5). Therefore, an adequate

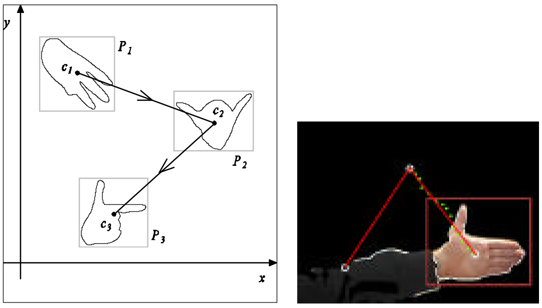

Figure 6 Staged path examples.

subset of C$_{AL}$I's default recognizable shapes is selected in order to adapt to the present system characteristics, that is, the ability to recognize simple trajectories or paths.

5.11. Staged Hand Path Recognition

The Staged Hand Paths recognition method is based on vector analysis and localized hand pose identification. The user hand describes a trajectory which consists of a limited point set (dependant on the video capture rate and the gesturing speed) representing segments of high curvature. This alone is inappropriate for direct vector analysis as defined in XML. Therefore, significant stages, or control points, must be identified in the drawn curve. As we are searching for vectors, we can subdivide the point set into a subgroup of line segments separated by zones of high curvature. The Douglas-Peucker [9] approximation algorithm is applied with an adequate accuracy parameter in order to retrieve the geometry of these lines. Once acquired, hand poses performed in its vertices are verified. If both geometry and hand pose syntax correspond to a known gesture, as defined in the XML syntax description of the Staged Path, then a positive match is found and the corresponding action is triggered (Figure 6). However, this demands very accurate gesturing as to obtain a given gesture's proper geometry.

A different approach based on gesturing timing was also implemented. It only accepts intermediate trajectory information, after the correct hand pose at a certain location is correctly executed and is expected, thus being more insensitive to movement.

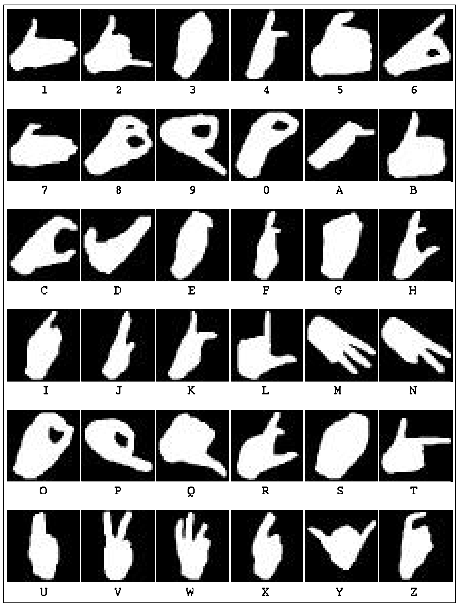

Figure 7 Portuguese Sign Language Symbols.

6. Results

Static hand pose recognition algorithms were tested for all the Portuguese Sign Language (PSL) 36 symbols (Figure 7). Each hand pose was tested ten times against a user library (that is, the user performing the gestures was the same user whose hand poses have been stored in the library) and, against another person's library (the user performing the tests was not the same one, whose hand poses images are in the library).

Table 1 Testing Static Hand Poses for Static hand pose recognition algorithms for all the Portuguese Sign Language (PSL) symbols.

Algorithms	Avg. Recognition Rate (%)	
	User Lib.	Generic Lib.
Template Matching	53.6	38.3
DCT	50.6	42.5
Hu Moments	25.8	17.8
PGH	58.1	30.3
SSD	5.8	2.8
PGH-SSD Hybrid	21.9	13.3

The results described in Table 1, show that Template Matching, DCT and PGH are, significantly, the best methods studied for robust shape recognition. Hu Moments are weak classifiers as well as SSD alone. An attempt to improve PGH effectiveness was taken by creating a hybrid algorithm based in its integration with Simple Shape Descriptors. These can't robustly recognize individual symbols, but are useful when excluding some possibilities from the standard set of PSL. PGH would then be able to precisely pinpoint the performed pose from this shortened set. However, at the current stage, we weren't able to effectively combine both algorithms own advantages, thus resulting, so far, on a poor hybrid method for shape recognition.

Although the average recognition rate for the best algorithm (PGH) is less than 60% when compared to similar state of the art systems having recognition rates up to 90%, we must consider that all possible symbols were included in the testing session. This means that highly correlated symbols have interfered in the process, thus causing false pose interpretations. When considered individually, we have identified a smaller set of Sign Language symbols which have a high standalone recognition rate (Table 2).

Table 2 Subsets of PSL symbols with average recognition rates higher than 80%, for the template matching, DCT and PGH algorithms.

Algorithms	Symbols
Template Matching	1,3,5,7,B,D,H,K,L,N,P,T,V,W,Y
DCT	1,2,5,7,9,A,B,D,L,N,P,T,Y
PGH	1,4,5,7,9,0,B,D,L,N,T,V,Y

When combined, and having selected an appropriate algorithm from the table above, the overall recognition rate can rise up to 90%.

A carefully chosen subset (or a user defined set of creative poses) can be used in specific control functionalities and Human-Computer Interaction for individuals with special needs, or in multimodal HCI. As seen in the next section, an application of this concept is already in operation in a Lisbon Museum.

The system can be also easily generalized to the recognition of other Signs Languages static hand poses, by just creating the appropriate hand pose library.

7. Conclusions and Future Work

In this paper, we have described the different architectural modules of a hand gesture recognition engine based on computer-vision. The system is configured with XML specifications that describe the type of gesture to be recognized in a given context: Static Hand Poses, Simple Hand Path recognition or Staged Hand Path recognition. The system was evaluated with an experiment were a user was issuing static hand poses of Portuguese Sign Language, to assess the robustness of various algorithmic alternatives to handle with the sub-problem of shape recognition, present in the hand pose understanding process. Our results have shown that Pair-wise Geometrical Histogram contour-based method, is the most effective in relation to the average symbol recognition rate metric, reaching the figure of 58.1% for the case of the own user library of symbols. If the test is only made with highly non-correlated symbols, that metric can rise up to 90%. We have also shown that our system can be generalized to the understanding of the static hand pose symbols of other Sign Languages and that is applicable in general Human-Computer Interface tasks, requiring the hand gesture recognition modality.

Although, one of our research aims is to pursue Portuguese Sign Language recognition, this initial experiment addresses only the recognition of the static symbols of this language. This is a clear simplification of the problem, since from previous work on American Sign Language - ASL recognition, it is clear that the facial expressions and signer's body are very important in the recognition process, as well as the recognition of both hands gestures. Another problem that has been encountered in ASL recognition is "co-articulation", which needs also to be tackled. In finger spelling, for example, the signer might not come to a clear precise formation of every symbol. To understand a particular sign language symbol, it is required to take into account the influence of the symbols that precede and appear after the current one. The recognition of a sign symbol on the basis of a static image of that symbol is still an initial approximation to the problem. We will need to develop, in our architecture, a higher layer for the recognition of a sequence of symbols, using lexical analysis, dictionaries and thesaurus.

As a natural continuation of our work, we aim at bi-manual gesture recognition and hand feature extraction for finger recognition and occlusion treatment.

Usability testing with people with special needs (hearing impairments, cognitive impairments, persons with mobility constraints, persons with problems in the verbal communicating, etc), or persons more oriented to natural forms of human-computer interaction, beyond the WIMP metaphor, would also be an enriching asset for this work and is planned in the short term. A usage scenario in a living room, were a user with hearing impairments, interacts with home appliances (controlling the Set-Top-Box, the TV Set, the DVD player, the home lighting, the window blinds, and activating macro commands, such as opening the lights and shutting down the blinds at nightfall) by means of selected static poses taken from the Portuguese Sign Language, with low correlation and that have shown more than 80% of recognition rates (namely the symbols corresponding to V, T, B, Y, W and 1), is currently in operation (see Figure 8). This work is being developed in collaboration with the Portuguese Foundation of the Communications, for the purpose of setting a demonstrator in the "House of the Future" test-bed of the Communications Museum in Lisbon, Portugal. Usability testing of this set-up is under way and will be reported in a near future, but initial tutorial sessions with the museum staff and the first visitors, has shown quite positive reactions.

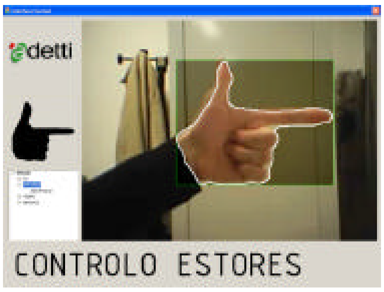

Figure 8 "House of the Future". The recognized static pose is presented to the user on the left, and the triggered action is indicated bellow (opening the window blinds).

8. References

[1] Waleed Kadous. *"Grasp: Recognition of australian sign language using instrumented gloves"*. Technical report, The University of New South Wales, Schools Of Electrical Engineering and Computer Science and Engineering, October 1995.

[2] Geoffrey E. Hinton S. Sidney Fels. *"Glove-talk 2 - a neural-network interface which maps gestures to parallel formant speech synthesizer controls"*. IEEE Transactions on Neural Networks, volume 9, January 1998.

[3] Santanu Chaudhurya Subhashis Banerjeeb Aditya Ramamoorthya, Namrata Vaswania. *"Recognition of dynamic hand gestures"*. The Journal of The Pattern Recognition Society. Department of Electrical Engineering and Department of Computer Science Engineering, IIT Delhi, 2000.

[4] Yoshinori Niwa Kazuhiko Yamamoto Jean-Christophe Terrillon, Arnaud Pilpré. *"Robust face detection and Japanese Sign Language hand posture recognition for human-computer interaction in an "intelligent" room"*. VI'2002. Office of Regional Intensive Research Project (HOIP), Softopia Japan Foundation, Faculty of Engineering, Gifu University, 2002.

[5] Yoshiaki Shirai Nobuhiko Tanibata, Nobutaka Shimada. *"Extraction of hand features for recognition of sign language words"*. VI'2002. Computer-Controlled Mechanical Systems, Graduate School of Engineering, Osaka University, 2002.

[6] Mario Aguilar Joshua R. New, Erion Hasanbelliu. *"Facilitating user interaction with complex systems via hand gesture recognition"*. ACMSE'03.Knowledge Systems Laboratory, Jacksonville State University, 2003.

[7] Larry S. Davis Thanarat Horprasert, David Harwood. *"A statistical approach for real-time robust background subtraction and shadow detection"*. Technical report, Computer Vision Laboratory University of Maryland, 1999.

[8] Jochen Maydt Rainer Lienhart. *"An extended set of Haar-like features for rapid object detection"*. IEEE ICIP'2002. Intel Labs, Intel Corporation, 2002.

[9] Gary R. Bradski. *"Intel open source computer vision library overview"*, 2002.

[10] J. S. Lim A. V. Oppenheim. *"The importance of phase in signals"*. Proceedings of the IEEE, 1981.

[11] P. D. Gianino J. L. Horner. *"Phase-only matched filtering"*. Applied Optics, 1984.

[12] Jörg Heuer André Kaup. *"Polygonal shape descriptors - an efficient solution for image retrieval and object localization"*. 34th Asilomar Conference on Signals, Systems and Computers. Siemens Corporate Technology, Information and Communications, Munich, Germany, 2000.

[13] Guojun Lu Dengsheng Zhang. *"A comparative study on shape retrieval using Fourier descriptors with different shape signatures"*. 2002. Fifth Asian Conference on Computer Vision (ACCV02). Gippsland School of Computing and Information Technology, Monash University, Australia, 2002.

[14] Manuel J. Fonseca Joaquim A. Jorge. *"A simple approach to recognize geometric shapes interactively"*. Technical report, Departamento de Engenharia Informática, IST/UTL, 1999.

[15] Ulrich Eckhardt Longin Jan Latecki, Rolf Lakämper. *"Shape descriptors for non-rigid shapes with a single closed contour"*. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2000. Dept. of Applied Mathematics University of Hamburg, Germany, 2000.

[16] Michael A. Wirth. *"Shape analysis and measurement"*. University of Guelph. CIS*6320 Image Processing Algorithms and Applications. Computing and Information Science Biocomputing Group, 2002.

[17] Wachsmuth I. et al, *"Deixis in Multimodal Human Computer Interaction: An Interdisciplinary Approach"*, University of Bielefeld, Germany, Gesture Workshoop 2003, April 2003, Genova, Italy.