# Use of a webcam for movement detection.

W.R.S. Silva, N.C. Pereti and J.G. Rogeri
*Instituto de Ciências Exatas e Tecnologias, JK, Universidade Paulista (UNIP), Sao Jose do Rio Preto – SP, Brazil.*

A.S. Pereira and N. Marranghello
*Department of Computer Science, Sao Paulo State University (UNESP), Sao Jose do Rio Preto – SP, Brazil.*

A. F. Araújo and João Manuel R. S. Tavares
*Faculdade de Engenharia, Universidade do Porto (FEUP) / Instituto de Engenharia Mecânica e Gestão Industrial (INEGI), Porto, Portugal.*

ABSTRACT: Accessibility evolving as a major concern nowadays, the suppression the utilization of a mouse while the user interacts via an interface becomes relevant as well. The work described herein aims at creating a method to control a computer through a webcam, which recognizes the movements of the user hands and moves the cursor accordingly. The proposed method turns out to be reliable and presents a low computational cost being capable of fulfilling the requirements of several applications.

## INTRODUCTION

It is a fact that the computer has contributed and continues to contribute to the evolution of society as a whole. Using a computer can make life easier for people in several ways, from day to day problem solving to people interaction via communication channels over the internet.

"Nowadays it is known that Communication and Information Technologies are increasingly becoming important instruments in our culture, and its use is a real mechanism of inclusion and interaction around the world […]" [4]. Communication and Information Technologies (CIT) are more conspicuous when used for the development of applications aimed at the social inclusion of handicapped people in our contemporary society. In such a case CIT can be used as supportive technologies using the computer itself to reach a predefined goal. It can be used an interface to input user's digital images that may be captured by a camera. These images can be immediately processed so that the user can interact with the computer by having his/her body movements interpreted on the fly.

From the point of view of the computational evolution it is possible to emphasize the development of software using this kind of technology for the conception of systems. The use of such techniques for Human-Machine Interaction not only contribute for the development of new supportive technologies for handicapped people, as give light to the production of a new class of interfaces capable of revolutionizing future virtual environments.

The work presented in this paper aims at carrying out the detection and recognition of hand movements for use in Human-Computer interfaces. The main techniques utilized are digital image processing as well as object recognition via *Haar* classifiers.

## DETECTION USING VIOLA-JONES' ALGORITHM

Detection is the initial phase of several design approaches. Several techniques can be utilized for such a task. However, Viola-Jones' object detector has recently been the most widely used one. The type of algorithm used by Viola-Jones [2] method for detection consists in applying rectangular features in images, finding a set of features that represent the desired object. It has made possible to precisely detect objects with great accuracy rate, low false positive rate and low computational cost. The algorithm has three parts, as described in the following paragraphs.

The first part consists in representing image characteristics based on Haar filters. This is done by using the complete original image. The main characteristics of the Haar filters as used in the original Viola-Jones' detector are presented in Figure 1. These characteristics are equivalent to the difference in intensity between the sub-regions.

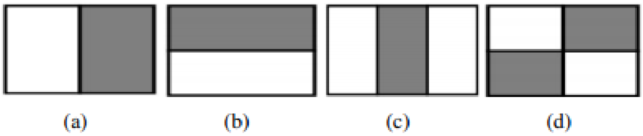

Figure 1 – Basic characteristics of the *Haar* filter. (a) Horizontal division; (b) Vertical division; (c) Two vertical divisions; (d) Horizontal and vertical divisions.

The second phase consists in generating a Boosting-based classifier to select the most important characteristics, decreasing the number of features generated.

The third phase consists in combining the classifiers in pipeline to ensure a good performance level as well as fine processing speed.

In Viola-Jones' detector representation of the training data in the feature space is obtained from the complete original image I (x, y), defined by:

$$I(x,y) = \sum_{x' \leq x, y' \leq y} i(x',y') \qquad (1)$$

where i(x, y) is an image of size L × C, $1 \leq x, x' \leq L$ and $1 \leq y, y' \leq C$.

With this approach four accesses to the image are enough to compute the sum on any chosen rectangular region.

A set of characteristics, given by the difference between the sum of two rectangular regions of pixels is easily obtained from the complete original image. This kind of characteristics is similar to the inner product with Haar wavelets being thus known as Haar-like features. In the original Viola-Jones approach the four characteristics displayed in Figure 1 have been used. In this case the value of a given characteristic is given by the difference between the sum of the pixels of the white region and the sum of the pixels of the black region.

In Figure 2 there is an extended set of characteristics used in more recent versions of Viola-Jones' algorithm. This extended set includes a new feature as well as rotated versions of those used for the original algorithm. Besides, the four rectangles feature that appears in the original algorithm is not used in this newer version.

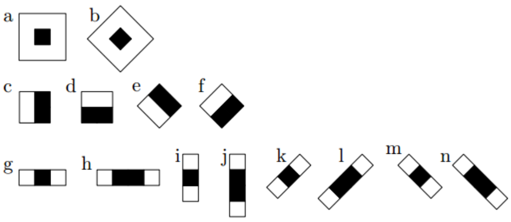

Figure 2 – Extended set of characteristics

Viola-Jones' training ser is composed by N samples of the kind ($x_n$, $y_n$), where $x_n$ is an image of dimension 24 ×24 and $y_n = \{0, 1\}$ is the label of the class. In this case $y_n = 1$ corresponds to the image of the object found, and $y_n = 0$ corresponds to an image not found.

Nevertheless, a 24 x 24 pixels window is used to obtain features by forming approximately 180 thousand feature combinations. This size of a feature space is too large. Any classifier would take too much time to analyze all this many feature combinations. As a solution to this problem the classifier is designed for the single feature that is found to perform best to separate positive from negative examples. Thus, an optimum feature separation threshold is determined by the simple classifier for the classification function.

For each AdaBoost iteration a set of weak classifiers $h_j$ is adjusted to minimize the classification error. Each such classifier corresponds to a $f_j(x_n)$ feature, where j = 1, …, J, and J is the total number of features. Given a threshold $\theta_j$ and a parity $p_j$, the classification rule is given by:

$$h(x) \begin{cases} 1, se\ p_j f_j(x_n) < pj\,\theta_j \\ 0, caso\ contrátio \end{cases} \qquad (2)$$

where the parity $p_j$ indicates the direction of the inequality.

From a practical point of view only one feature is not enough to detect objects with a low error rate. The features are selected by an algorithm that considers a synthetic set of features. An example of the outcome of such an algorithm can be seen in Figure 3 where it can be observed that two features were selected, i. e., one rectangle in the region of the eyes and the nose, and another rectangle on the forefront in the region between the eyes. In the first case the algorithm detected a difference in brightness between the nose and the eye area. In the second case a similarity in brightness between the eyes and the difference to the upper region of the nose had been detected.

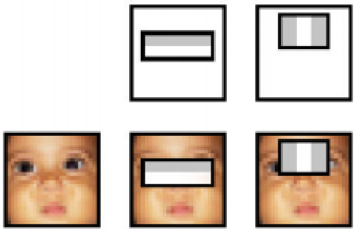

Figure 3 – Characteristics selected by *AdaBoost*.

According to Viola and Jones [3] it can be estimated that this operation would require approximately 60 microprocessor instructions. The following classifiers produce the same work but use a larger number of features. As the sub-window finishes its pass through each classifier it is applied to another more complex one, with a higher computational cost.

For a fast detection the first classifier has to delete most of the sub-windows of the image with a false positive, as it only performs the computation of two features.

## SYSTEMS BASED ON BOOSTING CLASSIFIERS PIPELINE

The classifier pipeline based on Boosting or AdaBoost is another class of methods used for object recognition. The technique is used for selecting a set of characteristics previously extracted from images. The most usual extraction approaches use either Gabor wavelets or Haar wavelets. Techniques using the latter one are often variations of Viola-Jones' detector [2].

## PROPOSED METHOD

Software using CIT has been welcome both by target (handicapped) people and by others that see in it an innovative, facilitating alternative technology.

The proposed system employs a webcam to capture the movement of the hands of a user to be used for the control of a mouse cursor. The user can move his/her hands over the system interface that can recognize the state and position of the hand. By state it is meant that a closed hand moving over the interface indicates to the system that the user is navigating; and an open hand represents the activation of a mouse button (clicking).

Viola-Jones algorithm [3] and gray scale image manipulation have been used for hand detection. The technique neither uses video movements nor pixel colors to speed up image processing. The idea used is the one of processing window that extracts image features and applies such features to a decision tree, which iteratively will inform the (non) existence of the object. In Figure 4 a model of the proposed system is displayed.

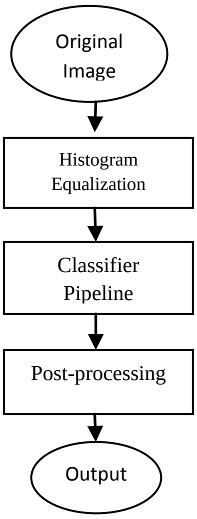

Figure 4 – Block diagram of the proposed system

For testing purposed the work has been implemented using C#, programming language object oriented with EMGU CV library, a wrapper library Open CV, which has methods to locate any objects in an image simply tell what the picture is, the size of the search window, the weights of the classifiers and the scale at which the object is sought in relation to the search window in which the classifiers were trained. The images used for detection are extracted from a real time video using a low cost webcam. Using the approach of Viola-Jones [2] to create the classifier, the mounting has been based in images with both variable (images with background objects) and infinite (images formed by hand and white background) background of the closed hand. Table 1 displays the amount of samples used by the classifier, an XML file containing several features, which are used to find the desired object.

Table 1. Image samples used by the classifier

| Type of image | Amount |
|---|---|
| Images with infinite background | 600 |
| Images with background variation | 400 |

In spite of the relatively small amount of samples used to set up the classifier (approximately 1000) the system can find the hand instantaneously. In Table 2 the detection rate with respect to the distance between the hand and the camera is presented.

Table 2. Detection with respect to distance

| Distance | Detection rate (%) |
|---|---|
| 30 cm | 100 |
| 60 cm | 95 |
| 75 cm | 85 |
| 80 cm | 0 |

It can be observed that, with increasing the distance of the object to be detected (hand) and the camera, the system can recognize incorrectly some areas like object located, generating a false positive. Figure 5 displays a hand grabbed by a system using the proposed method.

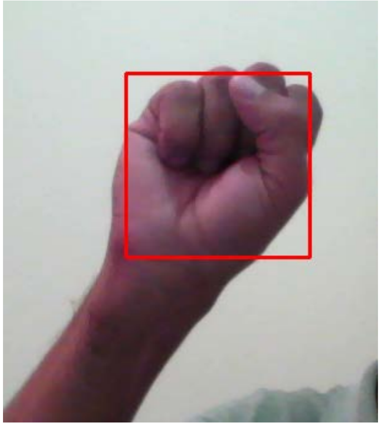

Figure 5 – Localized hand image.

## CONCLUSION

This work presented a system that allows people to use the computer without the need of a mouse device. This is mainly useful for handicapped people to interact with a computer system. The proposed interface is based on digital image processing. The algorithms implemented in this work are computationally light and mathematically accurate enough to produce results that are at the same time appropriately fast and exact. The implementation resulted in the object (hand) recognition that are successfully precise within a 1 second time frame. We also found out that the proposed method is quite sensitive to some intrinsic parameters of the classifier. We expect that by optimizing the code as well as the compilation process the parameter values can reach real time processing speeds, i. e., turn around times of about tenths of a second.

## REFERENCES

[1] FILHO, O. M.; NETO, H. V. Processamento Digital de Imagens. Rio de Janeiro: Brasport, 1999.

[2] VIOLA, P., JONES, M. “Rapid object detection using a boosted cascade of simple features”. In: Computer Vision and Pattern Recognition, 2001.
CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, v. 1, pp. I–511– I–518 vol.1, 2001.

[3] JONES, M.,VIOLA, P. (2003). Fast multi-view face detection. In MRL Technical Report TR2003-96, Cambridge.

[4] LÉVY, P. Cibercultura. São Paulo: Ed. 34, 1999.