NATURAL HUMAN-MACHINE INTERFACE USING AN INTERACTIVE VIRTUAL BLACKBOARD

Nicola Conci, Student Member, IEEE, Paolo Ceresato, and Francesco G. B. De Natale, Senior Member, IEEE

DIT- University of Trento (Italy)

ABSTRACT

Input peripherals such as mouse, tablet or touchscreen, significantly contributed to ease the attitude of humans towards computing machines. They reduce the need of a keyboard and make the interaction with the computer faster and more instinctive, in particular for unskilled users. Next step would be the complete removal of any tangible device, towards the concept of "disappearing computer". In this paper we propose an interactive virtual blackboard, based on a video processing and gesture recognition engine, which enables the user interacting almost seamlessly with the system giving commands, writing, and manipulating objects on a projected visual interface.

Index Terms— Human Machine Interaction, Multimodal Interfaces, Hand Gesture Recognition

1. INTRODUCTION

Human-machine interface (HMI) has always been a crucial aspect while developing computer applications. A good HMI should be versatile, fast, inexpensive, easy-to-use, ready-learned. Recently, the development of the e-society implied a massive deployment of advanced multimedia technologies in every application field, involving all user categories including elderly, children, disabled, and persons with very heterogeneous technical skills. They may benefit of smart environments that facilitate communicating, relating with other people, managing everyday activities (such as operating house appliances and living space functions). Nevertheless, many people remain skeptical, due to time and effort needed to understand system functionalities. Even simple concepts such as "drag&drop" or "double-click" may cause trouble to a novice, in particular if he is old, or affected by some physical or cognitive impairment. User acceptance is becoming the primary criterion for the success of new immersive technologies: technology should adapt to humans, and not viceversa. Systems that turn out to be too complicated or troublesome will be rapidly dismissed. These considerations are leading to a new generation of HMIs, characterized by important features such as being natural, unobtrusive, and invisible. The interaction between user and system requires a virtual communication channel that translates a user's "idea" into a command [1]. Typically, this communication involves one or more physical devices, encompassing a wide range of functionalities, from simple text typing to the manipulation of complex 3D scenes for games or immersive tele-presence. Narrowing the attention to input devices, we can roughly classify them into two main groups, according to their discrete (e.g., keyboards, buttons) or continuous (e.g., mouse, trackball) nature. The most likely evolution of discrete devices is towards language technologies, where typing a phrase or pushing a button is replaced by vocal interaction. On the other side, continuous devices can be readily used as a basis to handle 2D/3D worlds, through motion and gesture recognition. Focusing on the latter, several tools have been proposed so far to make such interaction more immediate and effortless. Touchscreens are a widespread example: the complexity of the underlying system is hidden and makes it possible for a user to point options as he could do in real life. The major limitations of touchscreens are costs, size, need of a physical location, and intrinsic limitation to 2D. More advanced devices proposed for virtual reality [2] include gloves [3][4], or other wearable tools such as mechanical sensors/actuators and micro-cameras [5][6]. They can handle 3D worlds, being natural and realistic, and can provide in some implementations tactile sensations. Unfortunately, their cost is usually very high, and user acceptance limited, thus making them more suitable for professional applications (e.g., a flight simulator or remote surgery equipment).

The most natural way to remove technological barriers would be adopting the hands themselves as "input devices". This requires the capability of understanding human patterns without the need of contact sensors, but relying on external devices able to acquire the patterns and to translate them into inputs. This can be achieved by using a video camera that grabs user's gesture, and a processing system that tracks the important features and classifies the relevant behaviors. Studies on hands, gestures and movement allowed developing models of the human body, making it possible to face the problem from a mathematical viewpoint [7]. Nevertheless, these approaches may be excessively complex

and sophisticated for typical application scenarios. In most cases, pattern recognition methodologies can solve the problem with lower hardware and computation requirements. In the following, we will consider these aspects by making reference as a proof-of-concept to a smart living environment, were a user can perform everyday actions (close a window, control oven temperature, tune radio, place a phone call, take a note) through an intelligent system that translates the user needs into practical actions.

2. VIB: AN INTERACTIVE VIRTUAL BLACKBOARD

As a proof of the above concepts, we developed a natural hand-motion based interface called VIB: interactive virtual blackboard. VIB was tested it in the context of a smart living space for aging people available at our Department. The proposed system removes any physical connection between user and domotic system, making use of the hands as instruments to efficiently handle objects and functions of common use. The interface is connected, through a central processing unit, to actuators taking charge of user commands. VIB is based on low-cost hardware architecture, easily deployable in almost any kind of environment, independently of the characteristics and variations of the surrounding conditions. The visual interface is projected through a beamer on a flat surface (table, wall) regardless of the size and perspective, and the acquisition of user's gesture is achieved by a simple VGA-resolution webcam. In our tests the HMI consists of a panel that allows controlling external sensors, appliances, home shutters, door lock, phone, etc. Furthermore, it implements some simple games. Using VIB, a user (e.g., a disabled lying in bed) can easily answer a phone call, open the door to a visitor after having seen him by a video-camera, close a window in another room without the need of standing up, simply moving his hands. Fig. 1 shows a simplified block diagram of VIB.

The first step is the acquisition phase. Since a standard input peripheral (keyboard, pointing device) will be unacceptable in this application context, we focus on possible alternatives by considering smart interfaces, inspired by the natural behavior of the user in real-world actions. Details about the implementation of a prototype interface with such characteristics are also provided. The choice of the capturing device must be done according to the idea of spreading the installation in homes, hospitals, schools, thus maintaining the resulting costs low. For this reason, special care was given to ensure good performance even in the presence of low-cost cameras. The camera is supposed to be fixed, and illumination slowly varying. Real-time constraints imposed a careful design of the processing system. To this purpose, unnecessary information is first removed. In particular, a background suppression procedure is performed in the HSV color space, where the scene can be modeled almost discarding illumination variations. Background suppression simplifies the hand detection and tracking, focusing the attention on areas corresponding to human skin color.

Fig.2 shows the necessary processing steps: initialization of the interface by putting the hand in a pre-defined area (a); extraction of the hue component (b); identification of the hand as the object to be tracked (c); the hand-tracking algorithm is run (d). In this way the impact of surrounding noise is reduced, as well as spurious skin-color areas (background, other parts of human body). The classification process is periodically repeated to detect hand posture and to decide operation modes. Next sections provide some insight about the two main processing blocks within VIB: hand recognition and hand tracking.

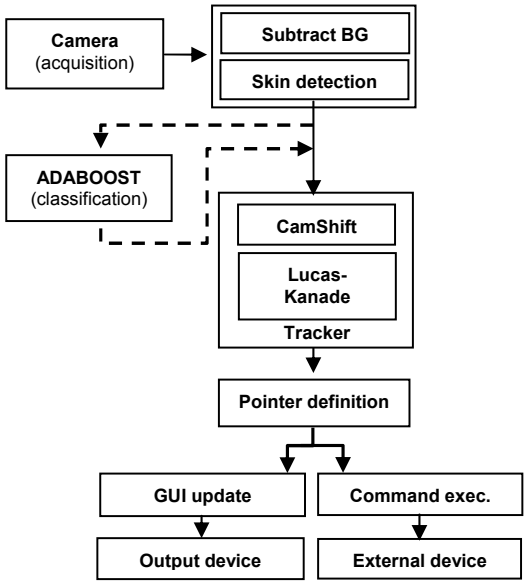

Fig.1. VIB block diagram

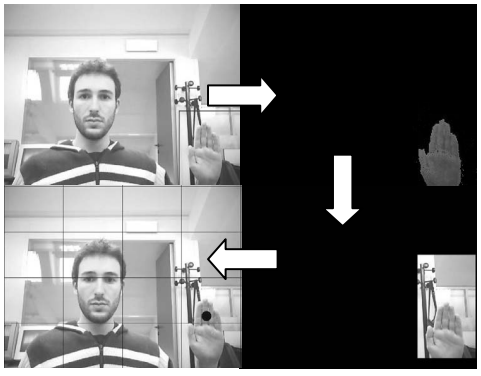

Fig.2. Initialization: starting position; hue image of initialization area; focus-of-attention; right hand detected, initialized.

3. HANDS RECOGNITION AND TRACKING

After the initialization, an Adaboost [8] classifier is responsible of locating hand position and classifying gesture (open, close, pointing, etc.). AdaBoost guarantees real-time operation, by employing a *cascade* of "weak" classifiers able to globally construct a strong learning algorithm. Each

classifier is tweaked to correctly detect the instances that were misclassified by the previous one, basing the analysis on an image representation called *integral image*. The integral image can be computed from the original one by using a reduced number of operations per pixel by calculating the Haar-like features, consisting of two or three joined "black" and "white" rectangles as shown in [9]. The integral image can be computed in a fast way by considering that the value of the pixel at location *x,y*, is the sum of the pixels above and to the left as in (1).

$$P(x,y) = \sum_{x' \le x, y' \le y} i(x',y') \tag{1}$$

where *P(x,y)* is the integral image and *i(x,y)* is the original image. Then, with reference to Fig. 3:

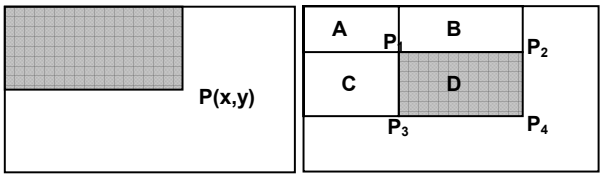

Fig. 3. Integral image computation

$$\begin{aligned} P_1 &= A, \\ P_2 &= A + B, \\ P_3 &= A + C, \\ P_4 &= A + B + C + D \\ D &= P_1 + P_4 - P_2 - P_3 \end{aligned} \tag{2}$$

The Haar-like features improve the classification speed and at the same time reduce/increase the in-class/out-of-class variability if compared with raw pixel analysis. The weak learning algorithm is designed to select the single rectangle feature which best separates the positive and negative examples determining the optimal threshold classification function. In order to detect the hand, the image is scanned with a sub-window containing a Haar-like feature as shown in Fig. 4, where each stage of the classifier cascade was trained to eliminate 50% of the "non-hand" patterns while falsely eliminating only 0.1% of the "hand" patterns. The complete hand detection cascade uses 18 classifiers. Accordingly, we expect a false alarm rate around $0.5^{18} \approx 3.8\text{x}10^{-06}$ and a hit rate $0.995^{18} \approx 0.91$. The target was to mimic three typical actions: moving a cursor (positioning), clicking (selecting a function or drag&dropping), and tracing a curve (writing, drawing, etc.). For this reason it was necessary to recognize three different hand postures: open, close and finger pointing. The classifier has been trained to detect each posture and in particular 5000 positive and 30000 negative patterns have been selected for the "open" hand (moving) while the "close" (drag&drop) and "pointing" (write) actions have been trained with 2500 positive and 5000 negative samples. The following step was to achieve a robust tracking of the hand motion along the video sequence. To this purpose, a *Continuously Adaptive Mean SHIFT (CAMSHIFT)* algorithm [10] has been used.

CAMSHIFT is based on the mean shift algorithm, a robust non-parametric iterative technique for climbing density gradient to find the mode of probability distributions. Unlike the mean shift algorithm, which is designed for static distribution, CAMSHIFT is designed for dynamically changing distributions. This feature makes the system particularly attractive to track moving objects in video sequences, where size and location change over time. In this way, it is possible to dynamically adjust the search window size. CAMSHIFT is based on colors, thus requiring the availability of the color histogram of the object to track the desired objects along the video sequences. As already mentioned, the color model was built on the basis of the hue component in the HSV domain. After having selected the search window size and its initial position, the spatial mean value is computed (4). The search window is then centered in that value. The centroid and first-order moment for *x* and *y* are calculated. The process is repeated until convergence. The quality of the tracker has been improved by applying the Lucas-Kanade feature [11], which is an iterative implementation of the Lucas-Kanade optical flow able to provide a good local tracking accuracy. The best solution to the tracking problem has therefore been obtained by applying the pyramidal implementation of the classical Lucas–Kanade algorithm, which relies on the residual pixel displacement vector that minimizes the image matching error [12]. Multiresolution tracking allows for relatively large displacements between images. According to these considerations, the basic principle is that a good feature is one that can be well tracked, in such a way that tracking should not be separated from the feature extraction process. If a feature is lost in a subsequent frame, the user can optionally ask the procedure to look for a new one, in order to keep the number of features almost constant. As an example, a good feature consists in a textured patch with high intensity variation both in *x* and *y* directions, such as the representation of a corner.

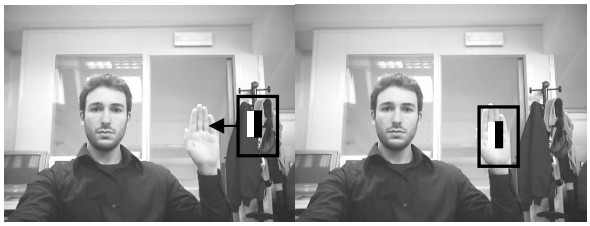

Fig.4. Image scanning and Haar-like feature matching.

4. USE CASE AND TRIALS

VIB was extensively tested and validated in a realistic environment, consisting of a smart living space. The target was to demonstrate the possibility of achieving a natural interface for an unskilled user of a domotic system. The interface allows accessing a number of services (TV, phone,

appliances) and controlling the relevant parameter (tuning, switching, opening-closing, moving) by using different hand postures and behaviors. Fig. 5 shows the result of the classification of the three hand postures defined in section 3: the dots represent the conventional hand locations in the three cases, while the gray points superimposed to the hand mark the texture patches used for hand tracking.

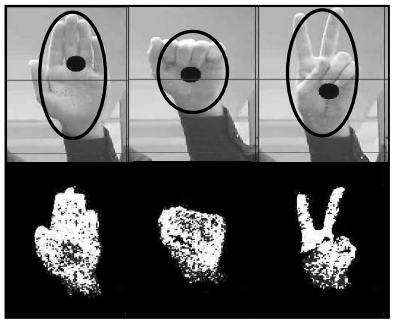

Fig. 5. Tracking of the chosen hand postures: open, close, pointing.

Fig. 6 shows two examples of interface operation: user tuning radio volume (left), and user employing a simple writing tool (right). In order to validate the performances of VIB, we considered the Fitt's Law [13], which is a common tool for establishing the Index of Performance (IP) of a pointing system. Results revealed a good capability of the system of following the hand movement, with an IP of 7.63, which is good if compared to traditional pointing devices such as a mouse, which IP is between 2.5 and 4 depending on the task to be done. A graphical representation of the linear regression is shown in Fig. 7, where each point represents the average time of 10 trials at the same difficult level. The line equation estimates the average time (MT) that is necessary to reach a certain target at a specific index of difficulty (ID), the flatter angular coefficient corresponding to better performance. The algorithm was tested on a P4 (2GHz – 512MB RAM) computer and a VGA camera. The processing has been performed at 10fps, which is a good compromise between precision and complexity.

Fig. 6. Volume adjustment and painting.

5. CONCLUSIONS

A new interface based on a virtual blackboard (VIB) has been presented. VIB demonstrate how current technologies may relief part of the problems connected to the need of interacting with complex technologies in everyday life. The main advantage of VIB is that it can be reproduced in almost any application scenario regardless of the environmental conditions, and does not require complex and costly equipment. It completely eliminates the need of physical connection between user and computer, thus increasing user acceptance. A more sophisticated prototype is being developed working in 3D by a multi-camera system.

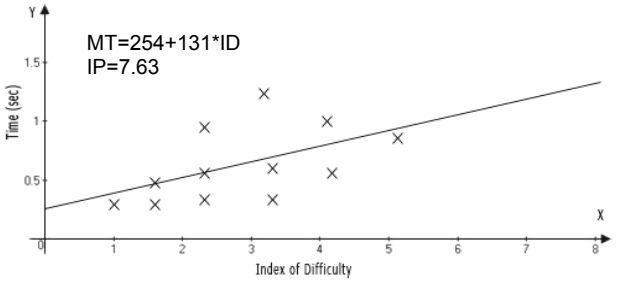

Fig. 7. Index of Performance by the Fitt's Law.

6. REFERENCES

[1] R. J. K. Jacob, "Human-Computer Interaction", *ACM Computing surveys*, March 1996, pp. 177-179

[2] D. J. Sturman, D. Zeltzer, *"A Survey of Glove-based Input"*, IEEE Computer Graphics and Applications, Jan. 1994, pp. 30-39

[3] R. Rosenberg, M. Slater, *"The chording glove: a glove-bsed Text input device"*, IEEE Trans. On Systems, Man, and Cybernetics – Part C: Applications and Reviews, May 1999, pp. 186-191

[4] N. Karlsson, B. Karlsson, P. Wide, *"A glove equipped with finger flexion sensors as a command generator used in a fuzzy control system"*, IEEE Trans. On Instrumentation and measurement, Oct. 1998, pp. 1330-1334

[5] A. Vardy, J. Robinson, Li-Te Cheng, *"The WristCam as input device"*, Wearable Computers, 1999

[6] Wong Tai Man, Sun Han Qiu, Wong Kin Hong, *"ThumbStick: A Novel Virtual Hand Gesture Interface"*, Proc. Of the IEEE International Workshop on Robots and human Interactive Communication, pp. 300-305

[7] B. Yi, F. C. Harris Jr., L. Wang, Y. Yan, *"Real-time natural hand gestures"*, IEEE Computing in science and engineering, May-June 2005, pp. 92-96

[8] P. Viola and M. J. Jones. *"Rapid Object Detection using a Boosted Cascade of Simple Features",* IEEE CVPR, 2001

[9] Rainer Lienhart and Jochen Maydt. *"An Extended Set of Haar-like Features for Rapid Object Detection",* IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002

[10] G. R. Bradski. "*Computer video face tracking for use in a perceptual user interface"*. Intel Technology Journal, Q2 1998.

[11] B. J. Yves, *"Pyramidal implementation of the lucas-kanade feature tracker"* Microsoft Research Labs, Tech. Rep., 1999.

[12] J. Shi and C. Tomasi. *"Good Features to track"*, IEEE Conference on Computer Vision and Pattern Recognition, pages 593-600, 1994.

[13] I. Scott MacKenzie, *"Movement Time Prediction in Human-Computer Interfaces"*, in Readings in Human-Computer Interaction (2nd edition), Los Altos CA, pp. 483-493, 1995.