ISSN 1870-4069

Hands Skin Segmentation and Tracking for Interaction with an Augmented Reality Entity

Miguel Sanchez-Brito, Carlos F. Garcia-Hernandez

National Institute of Electricity and Clean Energies, Cuernavaca, Morelos, Mexico

miguel.sanchez@ineel.mx,cfgarcia@ineel.mx

Abstract. Time invested in education has been growing in recent years and it is a subject of interest not only for schools, but also for companies, due to the time and money saved because of a correct understanding of the development of an activity. To contribute to learning, different techniques for the development of learning objects have been investigated. Novel techniques involve the use of augmented reality. In this research paper, the implantation of two innovative areas belonging to the computer science to support the learning process is presented: augmented reality and image processing. Through augmented reality we project virtual entities to the user computer screen and through image processing techniques, we perform the detection of an object that allows the interaction with the virtual entity without the need to use any special equipment.

Keywords: Augmented reality, image processing, virtual reality, learning object, object segmentation.

1 Introduction

Some of the most recent computational techniques used to support the learning process are virtual reality (VR) and augmented reality (AR), however, the difference between them is not clear in many cases.

In [1], the authors provide the following VR definition: "VR is high-end user-computer interface that involves real time simulation and interactions through multiple sensorial channels. These sensorial modalities are, visual, auditory, tactile, smell and taste". In the definition, real time means that the computer can detect a user's input and modify the virtual world instantaneously. In [2] is specified that the human brain can process between 10 and 12 images per second and detect them as a simple image and with a higher number of images will produce the sensation of visual continuity, so in VR, computer must be able to process more than 12 images per second.

In [3], AR is described as a reality in which virtual content is seamlessly integrated with displays of real world scenes, the formal definition proposed by the authors is: "AR is a combination of technologies that enable real time mixing of computer generated content with live video displays".

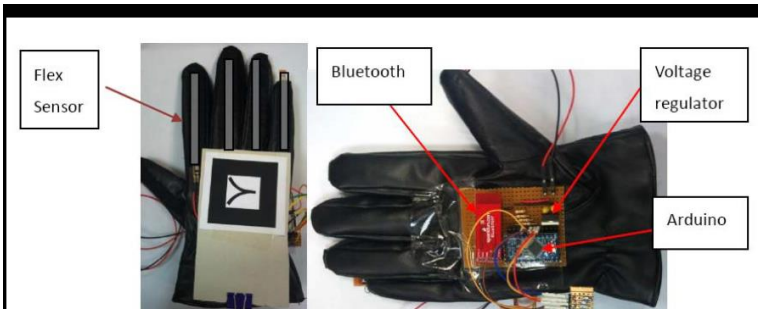

Fig. 1. Glove developed for the rehabilitation process.

Fig. 2. AR projection based on Feature Points.

Both, VR and AR consider the use of virtual entities, however, the most important difference is provided in [4]: "VR technologies completely immerse a user inside a synthetic environment. In contrast, AR is taking digital or computer generated information and overlaying them over in real time environment".

2 Related Work

In [5], the authors propose the use of glove with many sensors adapted to it, to interact with virtual entities using AR. The proposed methodology aims to support the rehabilitation process. The system is able to get a measure of any finger movement which is not relevant to the method proposed in this research work. Despite of being a good option to the rehabilitation process, the acquisition of the glove is crucial to interact with virtual entities to perform the rehabilitation process.

3 Digital Entities Modeling

To develop both, virtual environments and virtual entities, VR uses different software tools to create, first, a mesh model of the entity and then create the environment, once created the environment, the entity is exported into it. Finally, when the environment and the entities are mixed, the project is produced and exported to a device (lenses usually), in [6] this process is detailed.

In AR, the entity modeling process is a little different, the entity could be developed through software tools too, however, usually the environment is the real one recorded through a camera or any other scanner. To introduce the entity in the real world, often AR uses pattern recognition algorithms to project the virtual entity in the pattern usually

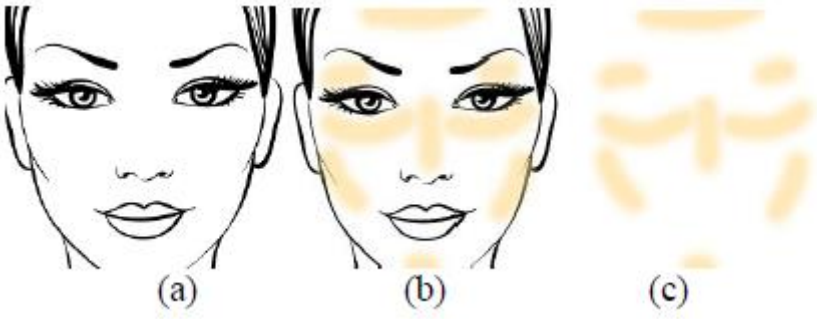

Fig. 3. Region detection to virtual entity projection.

printed in a sheet of paper, in [7, 8, 9], the use of this way of AR is detailed. Additionally, in [10], another way to combine the real world and virtual entities is exposed: training a machine learning methodology to recognize a specific object in the real world and project the virtual entity in it.

The machine learning algorithm is K nearest neighbor, and the data collection of the object where the virtual entity is going to be projected was obtained by Scale-invariant feature transform (SIFT) feature detector and descriptor, Figure 2.

Despite being a good strategy, the use of feature detectors and descriptors not only in AR, but also in all the computer vision field, still presents a great area of investigation, because of that, its effectiveness in an environment with uncontrolled conditions depends on expensive computational techniques like Random sample consensus (RANSAC). In [9], this problem is exposed and to try to face it, authors propose a methodology based on artificial neural networks called DeepAR, they compare their technique against Oriented FAST and rotated BRIEF (ORB) feature detector and descriptor. According to their results, their algorithm got a better performance than ORB; however, they mentioned that they needed in some occasions more than 200 iterations.

Authors in [10] present a little bit different way to detect the region where the virtual entity is going to be placed: Face Detectors, see Figure 3.

They implement the Qualcomm Snapdragon (Software Development Kit) SDK to detect a face. The SDK allows to detect some specific regions into the face: eyes, eyebrows, nose, chin and ears. Once detected the face, an image developed using Adobe Photoshop and which contains the makeup for the regions mentioned of the face is projected over the face.

4 Entity Interactions

Once developed and displayed the digital entity, it is necessary to establish a means of control to interact with them.

In the case of VR, the interaction with the digital environment is performed by sensors located in any specific body part, with which is possible to get a measure and

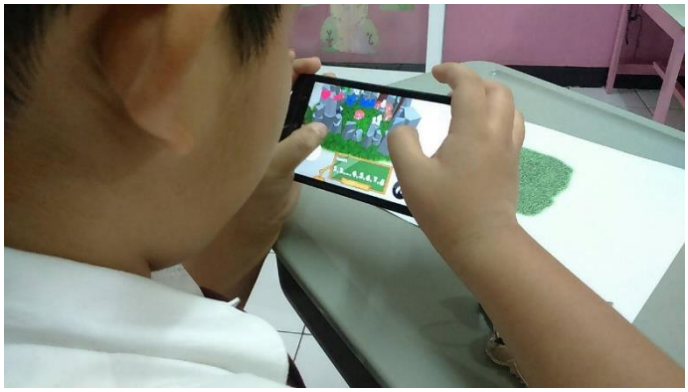

Fig. 4. Virtual entities projection through mobile device.

extrapolate it to the virtual environment. In nowadays is commonly interact in the virtual environment through buttons located on VR lenses.

A common way to interact with virtual entities is through mobile devices (smartphones principally). Research in [11, 12] promote this methodology. In [11], authors use SIFT to detect features in the image and the bag of words technique to recognize a specific object recorded by the cell phone. To solve the problem of processing capacity, they divide their system in a client-server architecture; the client part (cell phone) detects features and project the digital entities, after that, the system send the image recorded to the server through a conventional Wi-Fi connection. In the server part, the object recognition process is developed by the bag of words technique. In [12] an Arithmetic Learning game based in AR is presented. This research provides a client-server architecture, they use libraries, toolkits and SDKs to develop both, pattern recognition and entities modeling, however, they do not specify which ones, Figure 4.

Many techniques to project a virtual entity in the real world have been proposed, many techniques to interact with virtual entities too, however, almost all of them are based on the use of a hardware (gloves or touch screen of mobile device). To avoid the use of sensors or a mobile device, we propose the use of image processing techniques to recognize a specific object or body part recorded by a webcam and use it to interact with the digital object projected (pointer). With that methodology, the user will not need any other device additional to his/her computer with a webcam, however, the process of recognition and interaction with the digital object must be developed in real time (at least the software must be able to process more than 12 frames per second).

5 Application Development

In nowadays, many software tools are available to develop the virtual entities, either in third dimension or two dimensions. In this research project, we opted for the use of two dimensional images. To provide a high quality, the images were developed using Photoshop of Adobe. The size of the images is delimited by the webcam resolution, in

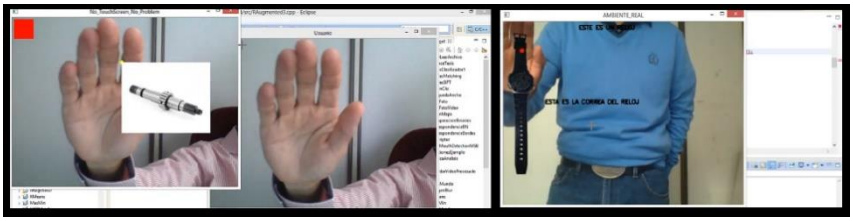

Fig. 5. JPG and PNG Image format comparison.

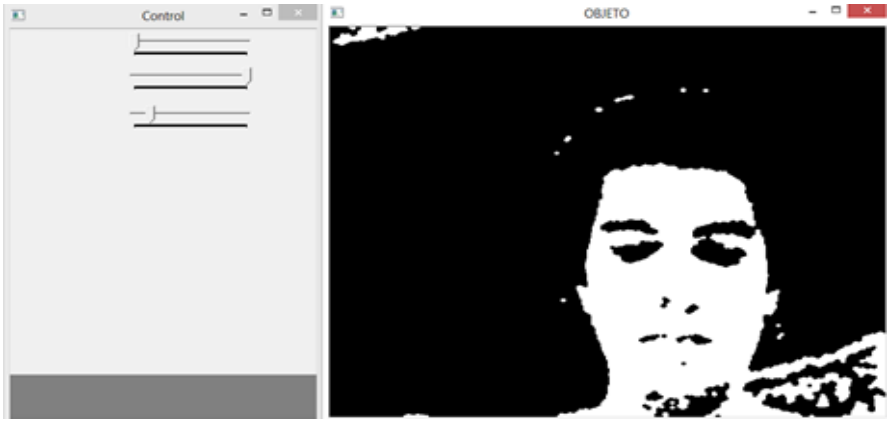

Fig. 6. Skin segmentation.

our case the webcam resolution is 640x480 pixels, so the images created through Photoshop must be smaller than this size.

After some experiments, we chose the Portable Network Graphics (png) format for images. The principal reason to select this format is because of its transparency, the use of a Joint Photographic Experts Group (jpeg) or a Windows bitmap (bmp) format will produce a solid surface around the image (image background), while it is projected to the video of the real environment, Figure 5.

5.1 Color Model

Many color models have been proposed until nowadays, some of them are: red, green, blue (RGB), Hue, Saturation, Lightness (HSL or HSI) or Hue, Saturation, Value (HSV), among others. The models contemplate different parameters to conform all the variety of colors, for example, from the RGB model uses colors Red, Green and Blue to conform all the colors, from black (0,0,0) to white (255,255,255).

5.2 Digital Entities Projection

Detect a specific surface to project the entity is not the main objective of this work, but rather the projection of the virtual entity and set a specific interaction zone on the object to interact with it. The main consideration when the entity it is going to be projected is

Fig. 7. Virtual entity projection and interactive zones.

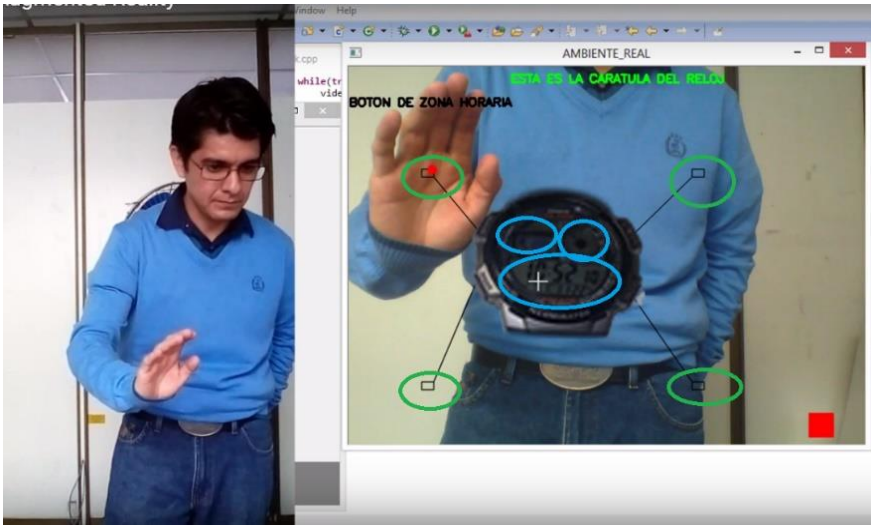

Fig. 8. Update interactive zones.

the opacity. Independently of the color model selected, is important to consider the opacity as another different color channel. For example, if we choose the RGB model, a fourth color channel must be considered "opacity (O)", redefining the model as RGB-O

5.3 Digital Entities Interactions

To interact with a digital entity, we have defined some specific areas on it; these are going to change depending on the image displayed on the user screen. When the cursor

is over the interactive zone, a bigger image of the area is going to be displayed together with a resume of it; we decided to use a body part (hand) as a cursor.

Before to applying an algorithm to recognize the object selected as a cursor, we must segment it from image recorded by the webcam, in order to delimit the region of the hand we must be able to detect the skin region. Due scene luminosity, it is difficult define some specific values to segment skin color and make easy hand detection. As a way to help on skin color detection independently of scene luminosity, a control panel is provided to establish the best color channels combination according of the luminosity at that moment, Figure 6.

Once segmented the object selected, we are able to apply an algorithm to recognize it. At present, many algorithms to develop the object recognition task has been proposed. On this work, we have selected Hu invariant moments [12]. To obtain the Hu invariant moments of the object selected, it is necessary binarize the object image. Once segmented the image we are able to apply the Hu equation (1).

Consider f(x,y) as the intensity of pixel (x,y) in a region. The moment of order (p+q) for the region is defined as:

$$m_{pq} = \sum_{x} \sum_{y} x^p \, y^q \, f(x, y). \tag{1}$$

6 Model Implementation

The algorithm proposed was implemented in a computer with the following characteristics:

- Processor: Intel Core I3, 2.40GHz, 64 Bits,
- RAM: 4.00 GB.

6.1 Performance

To evaluate the algorithm performance the following test was proposed. We defined two types of interactions:

- Displaying information about the actual model displayed on the screen. That means, define zones that show a summary of it, Figure 7.
- Displaying a model about the selected zone. That means, when the user overlaps the cursor in that zone, the actual model must be replaced for a model of the selected zone.

At the beginning of the test, the principal entity selected to interact with, is displayed. The specific interaction zones for actual image are stablished.

Zones under green ellipses display information about that specific part, while zone under blue one, update the principal entity with the zone selected and the new interaction zones are calculated, Figure 7 and Figure 8. The behavior on this level are similar that the mentioned before: Zones under green ellipses display information about

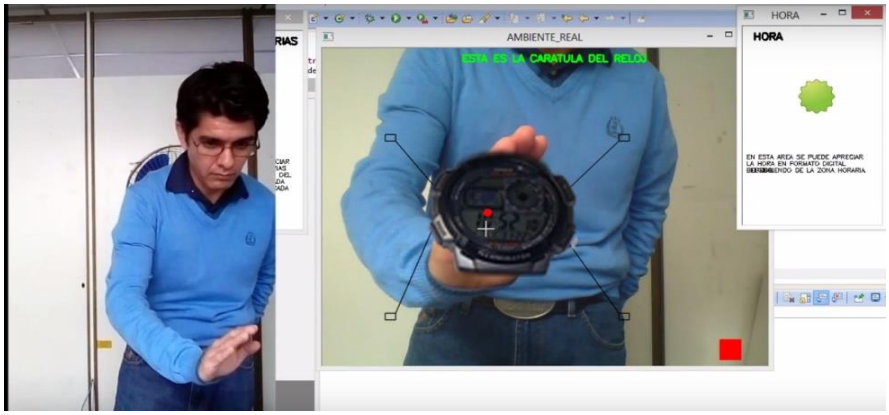

Fig. 9. Summary about hour zone.

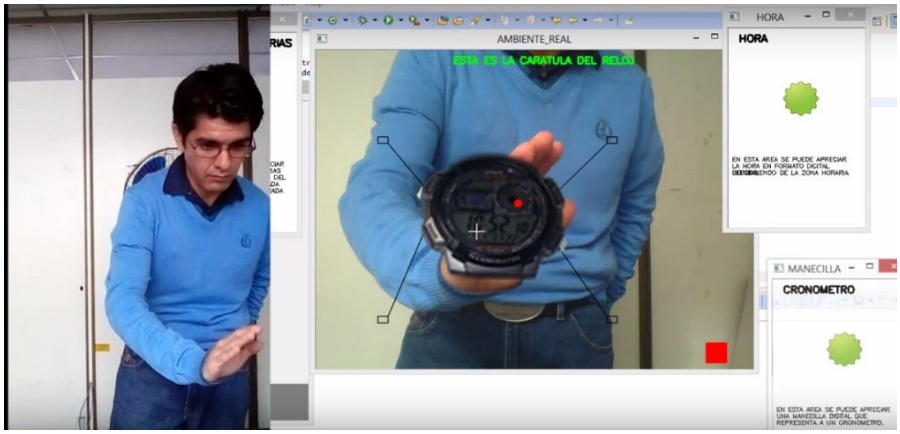

Fig. 10. Summary about chronometer zone.

Table 1. Activity time performance.

Activity	Time (milliseconds)
1 Recognize the skin color after establishing RGB values through control panel.	137
2 Display the virtual entity on screen	35
3 Display information about specific zone of the entity	18
4 Update the principal virtual entity	25
5 Open a window with information about a specific zone	19
6 Reestablish the principal virtual entity	32

that specific part, however, zones under blue ones, provide information in a new window, Figure 9 and Figure 10.

The red area works as a button to reestablish the principal virtual entity and close the windows opened.

After run 10 times this application, we provide the following summary of time performance for each activity, see Table 1.

7 Conclusions

In the present research work, a way to interact with virtual entities in augmented reality is proposed. The method is based on Hu invariant moments to recognize the object to be used as a cursor. Once recognized the cursor, algorithm is able to process more than 12 images per second, because of that, its performance is considered on real time according to [2]. The control panel to face scene luminosity variation is necessary to segment efficiently the cursor object.

8 Future Work

Because of present research work, we propose two principal future activities:

- The development of an algorithm able to recognize not only the hand as a cursor, but also the specific hand gesture to associate it with a specific action.
- To study the effects of the luminosity in objects to define a specific range of acceptable performance of the algorithm.

References

1. Burdea, G.C., Coiffet, P.: Virtual Reality Technology, Volume 1. John Whiley and Sons, Hoboken, New Jersey (2003)
2. Read, P., Meyer, M.-P.: Restoration of Motion Picture Film. Butterworth-Heinemann, Oxford (2000)
3. Mullen, T.: Prototyping Augmented Reality. John Whiley and Sons, Indianapolis, Indiana (2011)
4. Kipper, G., Rampolla, J.: Augmented Reality: An Emerging Technologies Guide to AR. Syngress, Waltham, Massachusetts (2013)
5. Zhang, D., Shen, Y.: An Affordable Augmented Reality based Rehabilitation System for Hand Motions. In: International Conference on Cyberworlds, Singapore (2010)
6. Parisi, T.: Learning Virtual Reality Developing Immersive Experiences and Applications for Desktop, Web, and Mobile (2016)
7. Gao, Y., Wang, H., Bian, X.: Marker Tracking for Video-Based Augmented Reality. In: Proceedings of the 2016 International Conference on Machine Learning and Cybernetics, pp. 928–932, IEEE (2016)
8. Mahadik, A., Katta, Y., Naik, R., Naikwade, N., Shaikh, N. F.: A Review of Augmented Reality and its Application in Context Aware Library System. In: International Conference on ICT in Business Industry & Government (ICTBIG), IEEE (2016)

9. Akgul, O., Penekli, H. I.: Applying Deep Learning in Augmented Reality Tracking. In: 2016 12th International Conference on Signal-Image Technology & Internet-Based Systems, pp. 47–54, IEEE (2016)
10. Oliveira, D., Guedes, P., Silva, M., Vieira e Silva, A., Teichrieb, V.: Interactive Makeup Tutorial Using Face Tracking and Augmented Reality on Mobile Devices, In: XVII Symposium on Virtual and Augmented Reality, pp. 220–226, IEEE (2016)
11. Ha, J., Cho, K., Rojas, F. A., Yang, H.: Real-Time Scalable Recognition and Tracking based on the Server-Client Model for Mobile Augmented Reality. In: XVII Symposium on Virtual and Augmented Reality, pp. 267–272, IEEE (2016)
12. Young, J., Kristanda, M. B., Hansun, S.: ARmatika: 3D Game for Arithmetic Learning with Augmented Reality Technology. In: International Conference on Informatics and Computing (ICIC) (2016)
13. Pajares Martin-Sanz, G.: Visión por computador/ Segunda Edición, RAMA, México (2008)