Hand Gesture Interaction Using Color-Based Method For Tabletop Interfaces

Abdelkader BELLARBI, Samir BENBELKACEM, Nadia ZENATI-HENDA, Mahmoud BELHOCINE
Centre de Développement des Technologies Avancées, CDTA, Algiers, Algeria
{abellarbi, sbenbelkacem, nzenati, mbelhocine}@cdta.dz

Abstract—In this paper, a hand gesture recognition method based on color marker detection is presented. In this case, we have used four types of colored markers (red, blue, yellow and green) mounted on the two hands. With this posture the user can perform different gestures as zoom, move, draw, and write on a virtual keyboard. This implemented system provides more flexible, natural and intuitive interaction possibilities, and also offers an economic and practical way of interaction.

Keywords: Computer vision, augmented reality, interactive table, hand gesture recognition.

I. INTRODUCTION

Augmented reality (AR) techniques have been applied to many application areas; however, there is still research that needs to be conducted on the best way to interact with AR content. As Buchmann et al. [1] affirms, AR interaction techniques need to be as intuitive as possible to be accepted by end users and may also need to be customized for different application needs. Since hands are our main means of interaction with objects in real life, it would be natural for AR interfaces to allow free hand interaction with virtual objects. This would not only enable natural and intuitive interaction with virtual objects, but it would also help to ease the transition when interacting with real and virtual objects at the same time.

Interactive tabletop systems aim to provide a large and natural interface for supporting direct manipulation of visual content for human-computer interactions. In this paper, we present a system that tracks the 2D position of the user’s hands using color markers on a table surface, allowing the user drawing, writing and manipulating virtual objects over this surface.

In creating an AR interface, that allows users to manipulate 3D virtual objects over a tabletop surface, there are a number of problems that need to be overcome [2]. From a technical point of view, it is necessary to consider tracking, registration accuracy and robustness of the system. From a usability viewpoint we need to create a natural and intuitive interface and address the problem of virtual objects occluding real objects.

Our implementation is based on a computer vision tracking system that processes the video stream of a single camera. The tracking of hands by a webcam not only provides more flexible, natural and intuitive interaction possibilities, but also offers an economic and practical way of interaction.

Vision-based hand tracking is an important problem in the field of human-computer interaction. Song et al. [3] list many constraints of some vision-based techniques used to track hands: methods based on color segmentation [4, 5] need users to wear colored gloves for efficient detection; methods based on background image subtraction require a fixed camera for efficient segmentation [6, 7]; contour based methods [8] work only on restricted backgrounds; infrared segmentation based methods [9, 10] require expensive infrared cameras; correlation-based methods [11, 12] require an explicit setup stage before the tracking starts; and the blob-model based methods [13] imposes restrictions on the maximum speed of hand movements.

The approach we’ve chosen uses color-based markers, to speed up the process of visual recognition by the machine, those markers are attached to the hand fingertips to tracking the hands position.

The paper is organized as follows. In section 2, we present some previously proposed interaction approaches to AR tabletop applications. In section 3 we give an overview of our system and explain the hand tracking approach. system implementation and some results are presented in the section 4. Finally, conclusions are drawn and future plans discussed.

II. RELATED WORK

Interaction with AR content is a very active area of research. In the past, researchers have explored a variety of interaction methods to AR tabletop applications. We present some of them in this section.

The most traditional approach for interacting with AR content is the use of vision tracked fiducial markers as interaction devices [14, 15]. This approach enables the use of hand-held tangible interfaces for inspecting and manipulating the augmentations [16,
For example, a fiducial marker can be attached to a paddle like hand-held device enabling users to select, move and rotate virtual objects [18, 19]. In the “VOMAR” [2] application, the paddle is the main interaction device, its position and orientation is tracked using the ARToolkit. The paddle allows the user to make gestures to interact with the virtual objects.

Some researchers have also placed ARToolkit markers directly over the user’s hands or fingertips in order to detect its pose and allow simple manipulations of the virtual objects [1, 20]. But this method leads to low marker detection because the markers are too small.

Lee et al. [15] suggest an occlusion based interaction approach, in which visual occlusion of physical markers are used to provide intuitive two dimensional interaction in tangible AR environments. Instead of markers, some approaches [22, 23, 24] use trackable objects like, for example, small bricks as an interaction device to the tabletop Augmented Reality. These objects normally are optically or magnetically tracked.

Many researchers have studied and used glove-based devices to measure hand location [25]. In general, glove-based devices can measure hand position with high accuracy and speed, but they aren’t suitable for some applications because the cables connected to them restrict hand motion [26]. Some systems use retro-reflective spheres that can be tracked by infrared cameras [24, 27, 28]. Dorfmüller-Ulhaas et al. [29] propose a system to track the user’s index finger via retro-reflective markers and use its tip as a cursor. The select command is triggered by bending the finger to indicate a grab or grab-and-hold (drag) operation.

The “SociaDesk” [30] project, uses a combination of marker tracking and infrared hand tracking to enable the user interaction with software-based tools at a projection table. The “Perceptive Workbench” system [31] uses an infrared light source mounted on the ceiling. When the user stands in front of the workbench and extends an arm over the surface, the hand casts a shadow on the desk’s surface, which can be easily distinguished by a set of cameras. Sato et al. (the Augmented Desk Interface) [10] make use of infrared camera images for reliable detection of user’s hands and uses template matching strategy for finding fingertips. This method allows the user to simultaneously manipulate both physical and electronically projected objects on a desk with natural hand gestures.

A common approach these days is to use touch sensitive surfaces to detect where the user hands are touching the table [25, 32], but Song et al. [3] arguments that barehanded interfaces enjoy higher flexibility and more natural interaction than tangible interfaces. Furthermore, tabletop touch interfaces requires special hardware that is still expensive and not readily accessible to everyone.

Lee et al. [33] proposed a hand tracking approach without any marker or glove, where it is possible to detect the five fingertips of the hand and calculate the 6DOF camera pose relative to the hand. This method allows the users to interact with the virtual models with the same functionality as provided by the fiducial paddle. In this case, there is no need to use a specially marked object, users can select and rotate the model with their own bare hand; But this proposed approach based on hand color distribution, so this method makes the tracking unstable (conflict between hand skin color and background colors).

Unlike Lee et al. approach, our approach is to place colored markers directly over the hands fingertips in order to detect its pose and allow simple manipulations of the virtual objects.

### III. PROPOSED METHOD

To allow users to interact with virtual content using natural hand gestures, different steps has to be done. The video has to be cut in different images. In these images, the hand has to be detected and tracked. Only after these steps, the effective gesture recognition can start. In this section, we show the necessary steps used in this project.

#### 3.1. Hand Detection and Tracking

Our proposed approach consists to place colored markers directly over the hands fingertips (Figure 1). So, to detect the hand’s positions, it’s enough to detect the placed markers positions.

![Color markers](image1.png)

In order to detect the colored markers, a color segmentation method is implemented using the Hue Saturation Value (HSV). A big advantage of this color space is that it is often more natural to think in hue and saturation. This color space is less sensitive to shadow and uneven lighting [34]. The articles [35] and [36] did also used this color space. A graphical representation is given in figure 2.
First the image has to be converted to the HSV format. From this converted image four new images are created in the four colors of the finger markers (red, blue, yellow and green) (see figure 3). To segment the image in different color images there are three different thresholds used, one in each component of HSV (hue, saturation and value).

![Figure 2: HSV color space.](image)

In this project we use the both parts (HGR & HPR). Like this, the user will be able to point on something for instance, and to be tracked (for example when drawing).

To be able to detect all the different gestures over the time, they have to be segmented. Of course there exist different methodologies to split these gestures. The simplest method is to have a kind of initialization posture [37]. In this kind, the starting and the ending posture would be the same. The problem of this strategy is, that it will not be very natural for the user.

Another possibility is to have a posture which delimits different gestures [37]. the third method is called velocity, this means that when the hands are not moving, it is considered as the end of the gesture.

Last but not least there is the method called acceleration. Normally when someone stops a gesture, they often accelerate afterwards to start a new one [37].

In our case, to determine which gesture is being expressed, a combination between colors of the detected markers and their positions is performed. The pictures below demonstrate some defined gestures (Figures 4, 5 & 6).

![Figure 3: Colors segmentations, (a & b): captured images, (c & e): colors filtered form the image a. (d & f): colors filtered form the image b.](image)

Once the markers were been found, they have to be followed in order to detect which movement the user did. The hands are therefore followed from frame to frame.

3.2. Hand Gesture Recognition

Hand gesture recognition can be divided into two parts; Hand Posture Recognition (HPR) and Hand Gesture Recognition (HGR). HPR is static which means that it is used for a single image. HGR is dynamic. The hand will be followed over a sequence of images [37].

![Figure 4: Gesture to click](image)

To move an object the user clicks down, moves the object and releases using left hand (figure 5).

![Figure 5: Gesture to move](image)
When the user wish resize an object he can use both hands and execute two click gestures at the same time as shown in figure 6.

![Image of gesture to zoom (resize)](image)

Figure 6: Gesture to zoom (resize)

IV. IMPLEMENTATION AND RESULTS

4.1. System Design

We have designed and implemented our vision-based projected tabletop interface using a video projector and a web-camera. The projector and the camera are mounted on support with an overlapping view of the tabletop (Figure 7).

In this setup, the camera can see the user hands during the interaction. As shown in Figure 7, the camera is overlooking the projected content on the tabletop and the hands as well.

Users need no other devices for interaction, hence the system provides direct, natural and intuitive interactions between the user and the computer.

![Image of hardware setup](image)

Figure 7: Hardware Setup

4.2. Results

This section presents some examples of the implemented gestures for the prototype. It is clear that for the future, other gestures should be included (see figures 8, 9, 10, & 11).

![Image of example of clicking gesture](image)

Figure 8: Example of clicking gesture

![Image of example of moving and resizing gestures](image)

Figure 9: Example of moving and resizing gestures

![Image of example of drawing](image)

Figure 10: Example of drawing

![Image of example of writing on virtual keyboard](image)

Figure 11: Example of writing on virtual keyboard
V. CONCLUSION

In this paper, tracking approach based on color marker recognition is proposed. This approach is used to recognize the hand gesture. In this case, we have used four types of colored markers (red, blue, yellow and green) mounted on the two hands. With this posture the user can perform different gestures as zoom, move, draw, and write on a virtual keyboard. This way allows the user to be more flexible when he performs a set of tasks. Also, the used approach reduces errors due to bad detection and the brightness variation and also offers an economic and practical way of interaction.

REFERENCES


