**Chapter 4**

**Hand Gesture Detection**

**4.1 Introduction**

Object detection play very important role in my own data set. To analyze and extract relevant data about and object of interest from an image, one need to first get that object in the image. Hand posture detection refers to finding the place and size of hand within a sequence of images. Now days it is very popular problem in computer vision problem and has many numerous applications, such as gesture recognition, sign language recognition, computer graphics games, and human computer interaction (HCI).

Skin color [Bretzn02, Mckenna04, Imagawa00] is an important property to detect hand and tracking. However, Color image have different problem of removing other objects with similar color such as face and human arm. To solve this problem, we introduce a new method in [Dardas11] to detect hand postures only using face detection and subtraction, skin detection, and hand postures contours detection and comparison algorithm. The face was detection removed because the skin detection will detect the face and the face’s contours are very similar to the fist hand gesture contours. Interest of area has been captured and other same skin color has been removed from interest of area. After removing unwanted area of the face, I detected the skin area using the hue, saturation, value (HSV) color model since it has real-time performance and it is strong against alternations, scaling and lighting conditions. Then, the interested area of contours was compared with all the existing hand posture template contours to eliminate unwanted interest of area like objects existing in the image.

**4.2 Hand Detection Approaches**

There are different approaches for hand detection have been introduced in the literature that employ different visual features and, in many cases, their combination. These features are motion, skin color, shape, and 3D models of hands. Hand detection methods were discussed in [Zabulis09] and will be discussed later in this chapter.

**4.2.1 Color**

Skin color detection has been used in many hand gesture recognition projects. A main objective towards giving a model of skin color is the choice of the color space to be utilized. Different color spaces have been introduced such as RGB, normalized RGB, HSV, YCrCb, and YUV**.** Color spaces that efficiently divide the chromaticity from the luminance parts of color are typically regarded as preferable as I did in my approach by removing the Value (V) section in HSV model. This is because of employing chromaticity-dependent mechanisms of color only.Generally**,** hull color detection can be disordered by background objects that have a hull color distribution like human skin color.Some project has been done on this problem by using background subtraction [Rehg94b, Gavrila96].On the contrary, it was expected that unwanted background subtraction normally depends on the camera system that does not move with respect to a fix background.Another solution [Utsumi98, Blake99] has applied the dynamic modification of background compensation techniques.

**4.2.2 Shape Hand**

Shape property have been working to discover in frames**.** More details can be acquired by reducing the contours of objects in the frame.If the contour is perfectly detected, it provides good presentation of the hand gesture which is indirectly related to viewpoint, skin color, and lighting. Typically, contour extraction based on edge recognition uses many edges fitting to the hand image area but also to distinct background objects.Accordingly, sophisticated post-processing techniques are needed to develop the presentation of this method such as our approach in [Dardas11] by combining skin color detection with contours detection and comparison after face subtraction. A second method that has been used in fingertip finding is pattern matching. Patterns can be images of fingertips [Crowley95] or fingers [Rehg95] or generic 3D cylindrical models [Davis94]. These pattern matching methods can be upgraded by using extra image features such as contours [Rehg94b].

**4.2.3 Learning Detectors from Pixel Values**

Currently, this technique that use a machine learning method named boosting have showed remarkably strong results in face recognition and good results in hand recognition [Chen07]. In [Viola01],an object recognition method was proposed in which a weak classifier may be a simple finder that uses basic image block differences efficiently calculated using an integral image. **On the other hand, this technique may provide an unnecessary number of weak classifiers. The AdaBoost method, discussed in section 2.3.9, has a drawback because it does not consider the elimination of chosen weak classifiers that no longer participate in the detection procedure. Besides, there are some issues to detect the hand using the Viola-Jones method [Viola01, Viola04] related to rotation and cluttered background as it will be discussed in Chapter 4.**

**4.2.4 3D Model-Based Detection**

**One major benefit of a 3D model-Based approach is that it can allow for view- independent detection. The utilized 3D models should have sufficient degrees of freedom to adapt to the dimensions of the hand that exist in an image. Different models use different image features to build feature-model correspondences. Point and line features are utilized in kinematic hand models for recovering angles created at the joints of the hand [Wu01, Shimada98, Wu99b]. Hand postures are then evaluated based on the relations between the 3D model and the observed image features.**

**4.3 My Approach for Hand Detection**

I propose an integrated system for detection, segmentation and tracking of the hand in a gesture recognition system using a single webcam.Some other methods that use color gloves **[Aran06, Tokatl05],** my method can detect the plain hand posture by integrating two useful features: skin color detection and contour matching.my proposed hand posture finding algorithm has real-time performance and is strong against rotations, scaling, a cluttered background, and lighting conditions. **Section 4.4.2 shows the robustness of our proposed hand posture detection algorithm based on comparison with other approaches.** Detecting the human hand in a cluttered background will boost the performance of hand gesture recognition systems used in Chapter 5 and 6 in terms of accuracy and speed. In this method, the speed and result of recognition will be the same for any frame size taken from a webcam such as 640×480, 320×240 or 160×120 and the system will be also robust against cluttered background because I process the detected hand posture area only.The small image size that holds the detected hand posture area only must be similar with the training images size of training stage as I will discuss the training and testing recognition systems stages in Chapter 5.

To detect the hand gesture in the image, a four phases system was designed according to my approach and as shown in Figure 3.1.First, we will open camera which have 50 squares box to capture hand gesture. Second Put your hand in those boxes and make sure your hand covers all the squares box. Third, the skin color locus for the image was removed for the user’s skin color after face deletion. Then last step, the hand gesture was spotted by removing false positive skin pixels and identifying hand gesture and other real skin color regions using contours matching with the loaded hand gesture patterns contours**.** Skin Recognition Area Loading Hand Postures Patterns Contours Face Detection and Subtraction Capturing Images from Webcam or Video file Templates Contours Comparison with Skin Area Figure 3.1: Hand posture detection stages

**4.3.1 Skin Detection**

Skin detection is a useful approach for many computer vision applications such as face recognition, tracking and facial expression abstraction, or hand tracking and gesture recognition.There are recognised procedures for skin color modeling and recognition that will allow to differentiate between skin and non-skin pixels based on their color**.** To get suitable distinction between skin and non-skin areas, a color transformation is needed to separate luminance from chrominance [Zhu04].The input images normally are in Color format (RBG), which has the drawback of having components dependent on the lighting situations.The misunderstanding between skin and non-skin pixels can be decreased using color space transformation**.** There are different approaches to detection skin color components in other color spaces, such as HSV, YCbCr, TSL or YIQ to provide more better result in parameter recovery under changes in lighting condition. Researches have shown that skin colors of individuals cluster closely in the color space for all people from different societies, for example, color appearances in human faces and hands vary more in intensity than in chrominance [Jun08, Kelly08, Yang98].Thus, take away the intensity V of the original color space and working in the chromatic color space (H,S) provides invariance against illumination situations. In [Zhu04], it had been well-known that removal the Value (V) component and only using the Hue and Saturation components, can still permit for the detection 96.83% of the skin pixels.In my application, I use the hue, saturation, value (HSV) color model since it has shown to be one of the most adapted to skin-color detection [Zarit99].It is also well-matched with the human color perception. In addition, it has real-time execution and it is more robust in cases of rotations, scaling, cluttered background, and changes in lighting condition.So, my projected hand gesture detection algorithm is real-time and robust against the mentioned previous changes.The other skin like objects existing in the image are removed by contour comparison with the loaded hand postures prototype contours**.** The HSV color space is gained by a non-linear transformation of the essential RGB color space.The conversion between RGB and HSV was described in [Ford98].Hue (H) is a section that characterises pure color such as pure yellow, orange or red, whereas saturation (S) provides a measure of the degree to which a pure color diluted by white light [Gonzal04]. Value (V) attempts to represent brightness along the grey axis such as white to black, but since brightness is subjective, it is thus difficult to measure [Gonzal04]. According to [Foley96] and Figure 3.7, Hue is estimated in HSV color space by a position with Red starting at 0, Green at 120 and Blue at 240 degrees.The black mark in the diagram at the lower left on the screen determines the hue angle.Saturation is a ratio that ranges between 0.0 along the middle line of the cone (the V axis) to 1 on the edge of the cone.Value ranges string from 0.0 (dark) to 1.0 (bright).

**According to [Jun08],** the HSV model can be resulting from non-linear transformation from an RGB model according to the following calculations. H=3.1, S = 3.2, V = 3.3 θ = arcos (3.4) As per a classification point of view, skin-color detection divided into two class problem: skin-pixel vs non-skin-pixel classification.Currentlythere are different known classification approaches exits such as thresholding, Gaussian classifier, and multilayer perceptron **[Nallap07, Greens01, Phung01].** In my research, I used a thresholding technique that allows get good result for higher computation speed when compared with other techniques, given our real-time requirements. This thresholding classification is used to find the values between two components H and S in the HSV model as I removed the Value (V) component. Usually a pixel can be observed as being a skin-pixel when the following threshold values are synchronised satisfied: 0° < H < 20° and 75° < S < 190°.

**4.3.2 Contour Comparisons**

Once the skin color has been detected, the contours of the detected skin color are recovered and then compared them with the contours of the hand gesture patterns.Once skin color contours are recognized as belonging to the hand gesture contour patterns, that area will be identified as a region of interest (ROI) which will then be used for tracking the hand movements and saving the hand posture in JPEG format in small images as shown in Figure 3.7.After that stored images will further be used to extract the features needed to recognize the hand postures in the testing stage as discussed in Chapters 5.

**If there are two hand gesture in the image, our system will alternate in detecting one of the two hands for every frame captured because the OpenCV function cvBoundingRect will enclose one rectangle only around the detected hand, which has the largest matching contours with the loaded hand posture templates contours. The single rectangle will enclose the detected hand posture for one frame and may enclose the other hand posture for the next frame if it has a larger matching contour.**

American sing language has different gesture for each alphabet and numbers. I captured the images for each sign. In which consist unwanted noise and removed the backgrounds from each of the images using background-subtraction techniques.