Engineering Applications of Artificial Intelligence 35 (2014) 54–70

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Real time hand detection in a complex background

Ekaterini Stergiopoulou, Kyriakos Sgouropoulos, Nikos Nikolaou, Nikos Papamarkos*, Nikos Mitianoudis

Image Processing and Multimedia Laboratory, Department of Electrical & Computer Engineering, Democritus University of Thrace, 67100 Xanthi, Greece

ARTICLE INFO

Article history:
Received 17 October 2013
Received in revised form
12 February 2014
Accepted 13 June 2014
Available online 8 July 2014

Keywords:
Region based hand detection
Complex background
Hybrid motion detection
Skin color detection
Color reduction

ABSTRACT

Hand gesture recognition has gained the interest of many researchers in recent years, as it has become one of the most popular Human Computer Interfaces. The first step in most vision-based gesture recognition systems is the hand region detection and segmentation. This segmentation can be a particularly challenging task when it comes to complex backgrounds and varying illumination. In such environments, most hand detection techniques fail to obtain the exact region of the hand shape, especially in cases of dynamic gestures. Meeting these requirements becomes even more difficult, due to real-time operation demand. To overcome these problems, in this paper, we propose a new method for real-time hand detection in a complex background. We employ a combination of existing techniques, based on motion detection and introduce a novel skin color classifier to improve segmentation accuracy. Motion detection is based on image differencing and background subtraction. Skin color detection is accomplished via a color classification technique that employs online color training, so that the system can dynamically adapt to the variety of lighting conditions and the user's skin color as well as possible. Morphological features of the detected hand in previous frames are employed to estimate the probability of a pixel belonging to the hand section in the current frame. Finally, the derived motion, color and morphological information are combined to detect the hand region. Experimental results show significant improvement in hand region detection, compared to existing methods with an average accuracy of 98.75%.

1. Introduction

The increasing spread of intelligent computing in everyday life has introduced a growing need for more intuitive and efficient ways of interaction between human and computers. Hand gestures are an appealing alternative to traditional currently-used devices (keyboard, mouse), since they form an extensive part of natural human communication (Ebert et al., 2012). Vision-based gesture recognition provides the potential for creating a new, easier and more powerful human – computer interface, because this task does not require any special hardware that might hinder user's comfort. It is also a non-intrusive information processing tool with many capabilities and a low-cost method, since only a web camera is required. Vision-based hand gesture recognition has a range of applications, such as sign language interpretation and learning, teleconferencing, distance learning, robotics, games, selection and object manipulation in virtual environments (Wachs et al., 2011).

Hand gesture recognition systems commonly consist of three main stages: (i) hand detection, (ii) hand feature extraction, and (iii) gesture recognition. Two major difficulties are usually encountered in these systems:

- Uncontrolled environments: An ideal hand gesture recognition system should operate regardless of the background complexity or the variety of lighting conditions. Nevertheless, the task of locating a rigid object in a complex background remains challenging in computer vision (Erol et al., 2007).
- Processing speed: The systems should be able to perform real-time gesture recognition. If the system performance is slow, it will be unacceptable for commercial applications. Simple and computationally efficient features are of great interest to machine vision (Wachs et al., 2011).

A common technique to cope with these difficulties is to apply restrictions on the user or the environment (Chua et al., 2002; Flasiński and Myśliński 2010; Ge et al., 2008; Lee 2008; Shimada et al., 2001; Ueda et al., 2003; Vatavu et al., 2009; Wilkowski 2009; Yang et al., 2009; Zhao and Chen 2009). Commonly encountered assumptions are that the background is plain, the

[*] Corresponding author: Tel.: +30 25410 79585; fax: +30 25410 79569.
E-mail address: papamark@ee.duth.gr (N. Papamarkos).
URL: http://www.papamark.gr/ (N. Papamarkos).

http://dx.doi.org/10.1016/j.engappai.2014.06.006
0952-1976/© 2014 Elsevier Ltd. All rights reserved.

hand is the only skin-colored object in the observed scene and that the lighting conditions are specific. The same assumptions were used in our earlier work (Stergiopoulou and Papamarkos 2009) on a hand-gesture recognition system.

In order to address the problem of vision-based hand detection in a complex background, most approaches in the literature employ visual features such as color, motion information, shape or a combination of these (Zabulis et al., 2009). Skin color can be estimated using off-line training data or face-pixel colors. In some cases, the skin color model can be adapted using the detected hand pixels from previous frames. Motion detection is achieved mainly by image differencing or/and background subtraction algorithms. The most representative methods for hand detection can be divided in two main categories: (i) methods that estimate a region containing the hand, and (ii) methods that extract the exact shape of the hand.

For hand region detection, Dadgostar et al. (2009) employ a thresholding skin detector in the hue color space, whose thresholds are constantly recalculated using the moving skin pixels of the scene. Moving skin pixels are detected by image differencing between consecutive frames. This method performs well as long as non-skin objects appear in the scene, whose hue color component falls into the skin detector range. Wilson and Salgian (2008) implement a gesture recognition method that uses a background subtraction technique. However, they assume that the background would be static with no illumination variation. Face and hand region detection is achieved by applying a Bayes classifier and Gaussian mixture models for the skin and non-skin classes. Alon et al. (2005) detect the face and then use the mean and the covariance of the face skin pixels colors in the normalized RGB color space to compute the skin likelihood image. If there is significant motion between the previous and the current frame, a motion mask, produced by image differencing, is applied to the skin likelihood image so as to estimate the hand likelihood image. They use sub-windows to extract the hand region based on the sum of their pixels likelihood. Their method implies that user's face is present in the scene, ideally illuminated, in order to construct the skin likelihood model. Guo et al. (2012) use an object detector based on weak classifiers, hard-thresholding skin color segmentation in HSV color space and background cancelation for hand region detection under complex background. Their method reduces the training time of their detector by using a new set of pixel-based hierarchical-features. The proposed window-based features exploit the fact that the hand is centered in each of the training images. Background cancelation is based on a pixel-wise background model trained over a period of time. In general, background cancellation improves hand detection. However, the hard-thresholding skin color classifier in the HSV color space cannot deal effectively with false positive detections.

For hand shape extraction, Alexander et al. (2009) use a two-frame difference to identify areas containing motion. Also, they perform hand detection by using a corner detection algorithm and geometric features of the hand. Unfortunately, the authors do not provide experimental results to evaluate their method and the presented techniques have not been examined in detail. Zhao et al. 2008 implement real-time gesture segmentation based on dual-complexion and an adaptive complex background model. Initially, they build the complexion model by means of a Gaussian distribution in the YCbCr color space and then apply it on the input image. The results are refined via a thresholding skin detection technique in the normalized RGB color space. The authors propose a background adapting modeling technique also based on the Gaussian distribution, which is able to adapt environment changes, but fails when the hand overlaps with skin color background. Chen et al. 2003 detect the moving region by image differencing. The result is refined by comparing the non-moving objects with the sample skin color. Then, edge detection is applied in order to separate the arm from the hand. The segmented hand is the output of the bitwise logical "AND" operation on the motion, skin and edge detection results. They also use a background subtraction technique, with a continuously updated background. Their method could lead to false positives, when a large object, like a sleeve, moves with the hand, since they do not use any pre-trained color model, but only color samples from moving objects. The HSV color space is used by Dardas and Georganas (2011) for thresholding classification of skin and non-skin regions. The contours of the skin detected regions are then compared to the contours of static gestures templates to decide whether the detected region contains a hand or it is a false positive. However, the authors do not provide any details on the actual contour comparison algorithm, and on the impact of the number of different gesture templates on the detection rate. Donoser and Bischof (2008) combined a skin color likelihood algorithm with an interest region detector for real-time hand tracking. They analyze color cues to calculate a skin color probability value for every pixel in the frame. A detector that estimates the high probability-connected regions, which display low probability values along their boundaries, is then applied to extract the hand region. The proposed technique can be applied only when the hand is the only object that is similar to the color model. Mao et al. (2009) combine the object detector, proposed by Viola and Jones (2001), with a skin color filtering technique to detect and track the hand in complex background. Skin detection is applied to remove background pixels, followed by a real-time object detector. The hand detector introduced by this technique improves the standard object detector of Viola and Jones against complex background, but still fails with skin color background. Okkonen et al. (2007) combine background subtraction with histogram-based color segmentation for a robust skin area segmentation algorithm. The main disadvantage of this method is that it cannot adapt to changes in the background since the background image is composed as an average of the first N images in the video sequence.

In this paper we propose a new method for real-time hand detection in a complex background. The main motivation behind the proposed technique is to address the problem of uncontrolled environments without using restrictions along with low computational cost and inexpensive hardware. These difficulties can be amended by means of a robust and more efficient hand detection technique. The proposed real-time hand detection method takes advantage of motion, skin color and morphology information, in order to increase effectiveness and robustness. The aim of our paper is the implementation of an effective and real-time hand detection system which operates in a complex background and under various illumination conditions. In addition, this system could be exploited further for dynamic gesture recognition. The second objective of the proposed technique is the precise extraction of the hand shape, i.e. the palm and the raised fingers should be well detected by the system, aiming further at static gesture recognition.

The novelties introduced in this paper are summarized as follows: Firstly, we introduce a modification to the motion detection algorithm of Collins et al. (2000). The proposed modification addresses the problem of misdetection when the moving object has the same color as the background, a common situation in hand detection applications. We also propose the combination of online and offline training of the Skin Color Map (Bayes) classifier, which has been used by other researchers only as a pre-trained classifier. Additionally, an algorithm which defines morphology weights of hand pixels is proposed. Finally, our technique employs a color reduction algorithm to define arbitrary shaped areas of similar color in which the derived motion, color and morphological information is combined. The proposed region-based approach differs significantly from other methods mentioned earlier, as we combine geographical proximity criteria (e.g. simple window-based region) with important color homogeneity criteria.

2. Proposed method

2.1. Overview

In this section an overview of the proposed method is presented. The method consists of four main stages: (i) motion detection, (ii) skin color detection, (iii) morphological descriptors extraction, and (iv) combination of extracted information in each region of interest. The flowchart of the entire method is depicted in Fig. 1.

For motion detection, a hybrid technique is used. Specifically, image differencing of three consecutive frames, which detects sudden movements, are considered in order to define the motion Region of Interest (mROI). Consequently, a background subtraction step is applied on the mROI, in order to track the hand even if it stops moving temporarily. This algorithm uses two different reference models:

(i) a background model, and
(ii) a skin-colored background model.

The background model describes the observed scene without the moving hand and is updated in order to adapt to possible changes. The skin-colored background model is derived from the background model and depicts skin-colored objects in the scene. This model is used in order to cope with detection errors when the hand overlaps with other skin-colored objects.

Skin detection is based on a color classification technique and more specifically on a modified version of the Skin Probability Map (SPM) (Jones and Rehg, 2002) in the HSV color space. The modification of the SPM technique involves the incorporation of an online color probability map training step. Online training renders the overall technique adaptive to the user's individual skin color, to the background colors and to the illumination conditions.

The morphology descriptor stage is a feedback stage, since it uses the final detected hand of the previous frame. The detected hand's morphology is described in terms of weight factors. A weight factor is equal to the minimum distance (horizontal and vertical) of a hand pixel to the hand contour and estimates the probability of this pixel to be part of the hand in the current frame.

Finally, in the last stage we combine the information extracted from the previous stages. The combination is accomplished in a region-based approach, in order to take into consideration the information that is provided not just from a single pixel or a neighborhood window, but from an arbitrarily shaped area with similar color. This is achieved by over-segmenting the input frame using color reduction. In particular, the graph theoretical clustering algorithm (Matas and Kittler, 1995) is applied and the input frame is divided into uniformly colored regions. Each one of these regions is rated in order to specify its possibility to be part of the hand region. This ratio relies on the results of the motion and skin color detection stages, the skin-colored background model and the morphology weights. The output is a grayscale image that via Otsu binarization (Otsu, 1979) results to the final hand region. Each stage is explained more thoroughly in the subsequent sections.

Finally, it should be noted that the aim of the proposed method is to detect a hand moving in real-time, not necessarily continuously, in front of a non-uniform background, under a variety of illumination conditions. The hand should be the largest moving object in the scene; the camera should be steady and the background fairly static. The presence of the user's face in the scene is not a problem, since it is relatively static and thus is treated as a part of the background.

2.2. Motion detection

2.2.1. State of the art

Motion detection is an interesting research field in computer vision. Segmenting video sequences into moving and background regions, i.e. detecting moving blobs in a sequence, can have many interesting implications for recognition, classification, and activity analysis tasks, since only moving regions need to be processed (Collins et al., 2000). Therefore, it is a structural component of many applications, such as surveillance, where one or more subjects are being tracked over time and possibly monitored for special actions, control applications (game interface, virtual environments) and motion capture analysis (Moeslund et al., 2006). A successful motion detection technique has to overcome challenges, including changing illumination conditions, shadows and noise. Motion detection is used in the preprocessing stage in order to define the hand region.

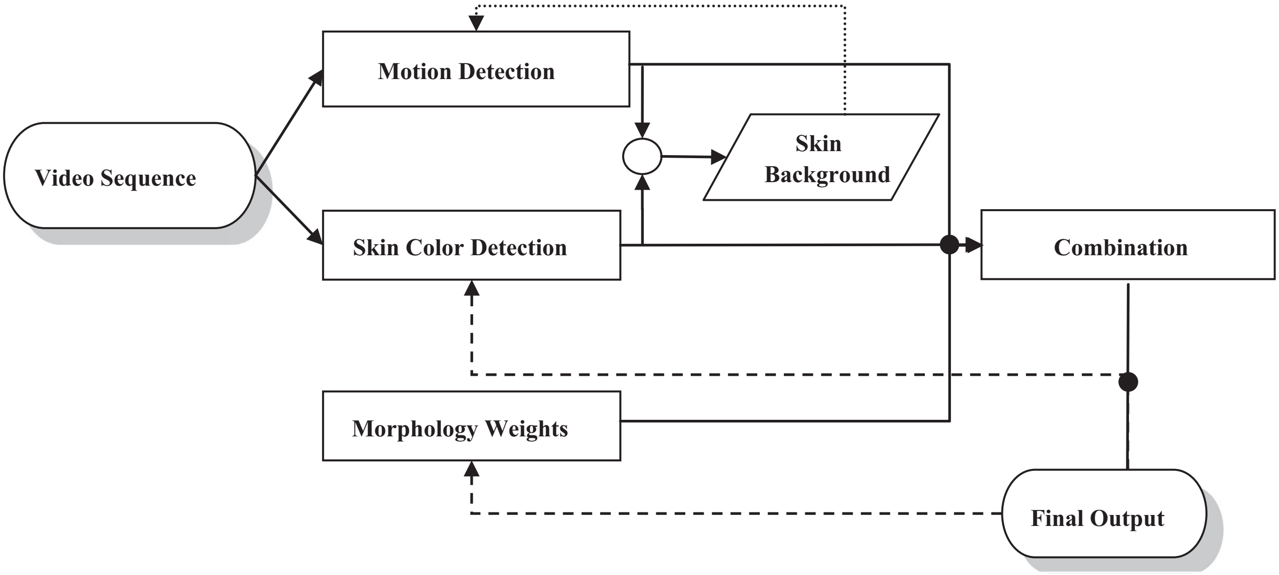

Fig. 1. Flowchart of the proposed method.

Assuming that the camera is steady and the background is fairly static, the main criteria for selecting a motion detection algorithm are its low computational cost and effectiveness. The most popular and computationally efficient approaches are image differencing (Dadgostar et al., 2009; Alon et al., 2005; Chen et al., 2003) (temporal differencing) and background subtraction (Wilson and Salgian 2008; Zhao et al., 2008). In the present paper, a hybrid motion detection technique has been applied, based on a modified version of the Collins et al. (2000) algorithm. They designed and implemented a system for automated video surveillance, which is able to track and classify moving objects into semantic and activity categories (e.g. human running, car moving). They developed a method for moving object detection, which is a combination of background subtraction and image differencing. We choose to use the Collins et al. algorithm in our approach, since it seems to give robust results under various illumination conditions.

Image differencing is applied by simply subtracting the current image of a video sequence from the previous image in a pixel-by-pixel basis using intensity values. Background subtraction attempts to detect moving regions by subtracting the current image of a video sequence from a reference background image. One may encounter many variants of the two aforementioned techniques. These differ in terms of the background model type and the background update procedure. Both algorithms have several shortcomings and therefore applying each one individually is not very efficient.

As mentioned by Collins et al., image differencing fails to extract the whole hand region, since it usually generates “holes” inside the detected region, due to similar intensity values. Similarly, background subtraction is extremely sensitive to dynamic scenes, due to lighting and extraneous events. In addition, it usually fails to handle situations, when a stationary object in the scene starts to move. “Holes” are usually created, where the newly exposed background differs from the “known” background model. In order to overcome these problems and create a robust motion detection technique, a combination of these algorithms for hand region detection is proposed here. This approach includes an image differencing scheme that adapts quickly to scene changes and detects sudden movements. This technique is used to determine the motion region, followed by an adaptive background subtraction scheme that is able to extract the entire moving object in a compact manner.

2.2.2. Image differencing

According to Collins et al., the previous image differencing algorithm is improved by using three consecutive frames instead of two. The three frame differencing algorithm suggests that a pixel is adequately moving, if its intensity has changed significantly between both the current image and the last frame, and the current image and the next-to-last frame. If $I_n(x,y)$ is the intensity of a pixel with coordinates (x,y) at frame n, then this pixel is moving if

$$\left|I_n(x,y)-I_{n-1}(x,y)\right| \geq T^{id} \text{ AND } \left|I_n(x,y)-I_{n-2}(x,y)\right| \geq T^{id} \tag{1}$$

where T^{id} is a threshold describing significant changes in intensity. Its value is set to 8.

The bounding box of the largest object in the outcome image is considered to be the motion Region of Interest (mROI). Fig. 2 shows the result of the image differencing technique. The red rectangle depicts the mROI.

2.2.3. Background subtraction

The background subtraction algorithm proposed by Collins et al. (2000) is also applied to the mROI. Let $\overrightarrow{B}_n(x,y)$ be the RGB color vector of a pixel of the background model with coordinates (x,y) at time n, and $\overrightarrow{CI}_n(x,y)$ the RGB color vector of the respective pixel of the current image at time n. A pixel is moving, if the Euclidean distance of these color vectors is greater than a threshold.

$$\|\overrightarrow{B}_n(x,y)-\overrightarrow{CI}_n(x,y)\| \geq T_n^{bs}(x,y) \tag{2}$$

The background model $\overrightarrow{B}_n$ and the difference threshold $T_n^{bs}(x,y)$ are determined by the following relations:

$$\overrightarrow{B}_{n+1}(x,y)=\begin{cases}\overrightarrow{B}_n(x,y), & \text{if}(x,y)\in mROI\\ a\overrightarrow{B}_n(x,y)+(1-a)\overrightarrow{CI}_n(x,y), & \text{otherwise}\end{cases} \tag{3}$$

$$T_{n+1}^{bs}(x,y)=\left(\begin{array}{ll}T_{n+1}^{bs}(x,y), & \text{if}(x,y)\in mROI\\ aT_{n+1}^{bs}(x,y)+5(1-a)\left|\overrightarrow{B}_n(x,y)-\overrightarrow{CI}_n(x,y)\right|, & \text{otherwise}\end{array}\right. \tag{4}$$

where a is a time constant that determines the adaptation speed. The recommended value of a is 0.7 and the recommended value of the initial threshold (T_0^{bs}) is 30. The initial background model ($\overrightarrow{B}_0$) is considered to be the first frame of the video sequence.

The background subtraction algorithm fails to extract the entire hand when the latter moves over background objects of a similar color, i.e. skin color (Fig. 4(b)). To address this, we have made the following amendments to Collins et al. (2000) technique. Based on the background model $\overrightarrow{B}_n$, a skin-colored background model is created $\overrightarrow{sB}_n$ (Fig. 3(b), for details see Section 2.6), where the non-white pixels are possibly skin colored. The background subtraction rule is enriched and thus becomes:

$$\begin{cases}\|\overrightarrow{B}_n(x,y)-\overrightarrow{CI}_n(x,y)\| \geq T_n^{bs}, & \text{if } \overrightarrow{sB}_n(x,y)=white \text{ (Not skin colored background)}\\ \|\overrightarrow{B}_n(x,y)-\overrightarrow{CI}_n(x,y)\| \geq T_n^{bs}/3 & \text{if } \overrightarrow{sB}(x,y)\neq white \text{ (Skin colored background)}\end{cases} \tag{5}$$

As it can been seen in Fig. 4(c), the problem of false negatives, which occurs when the hand and background objects of similar color overlap in space, is reduced in the proposed modification.

Fig. 5 shows the background model and the background subtraction outcome. As stated before, the background subtraction algorithm is applied only on the mROI (red rectangular).

2.3. Skin detection

2.3.1. State of the art

The majority of current hand-detection methods use skin color information as a primary cue. This choice is based on the fact that color is a computationally efficient, easy to understand and highly robust feature, invariant to morphologic variations and geometric changes of the hand, such as rotation, scaling, or translation. Nevertheless, skin color can be very sensitive to illumination conditions (indoor, outdoor, highlights, shadows, non-white lights), ethnicity variance, and dependency on camera characteristics. Research efforts have focused on overcoming these inherent problems. Studies have attempted to decide on a suitable color space to represent skin as well as a proper classification method.

More specifically, it has been proposed to represent skin color in a color space that separates luminance from chrominance components, in order to remove the luminance component (Chai and Bouzerdoum, 2000; Lee and Yoo, 2002; Yang and Ahuja, 1998; Yoo and Oh, 1999). This technique has been reported to improve the separability between skin and non-skin classes, increase similarity among different skin tones and eliminates the effect of varying lighting conditions. However, the effectiveness of this tactic has been supported only theoretically and no sound results

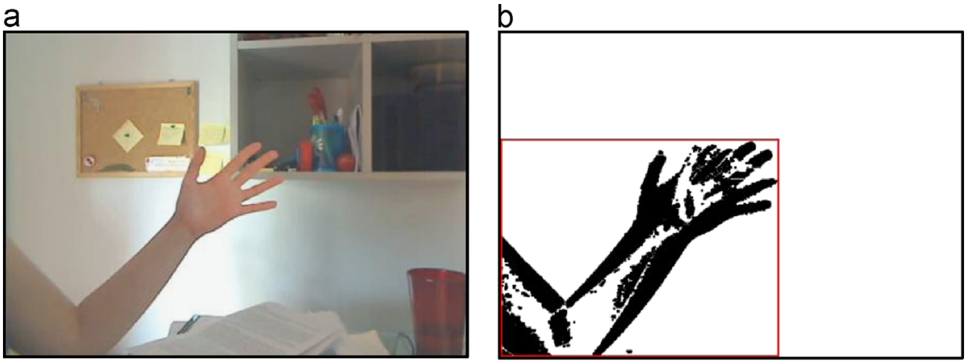

Fig. 2. Image differencing: (a) input video frame and (b) result.

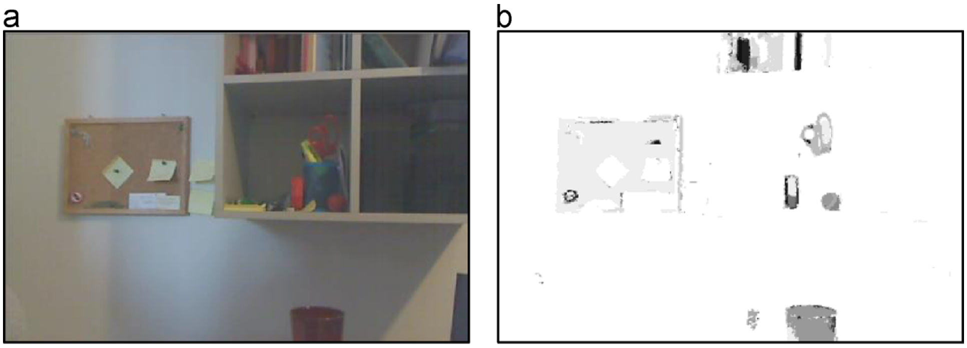

Fig. 3. (a) Background model and (b) skin background model.

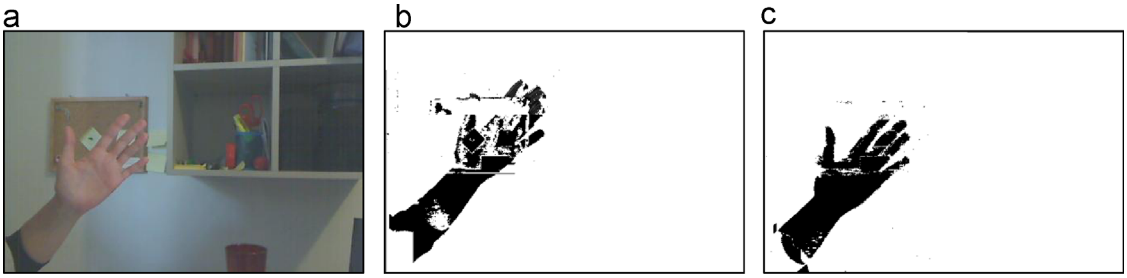

Fig. 4. Background subtraction algorithm modification: (a) input video frame; (b) background subtraction without taking account of the skin-colored objects of the background and (c) background subtraction after taking account of the skin-colored objects of the background.

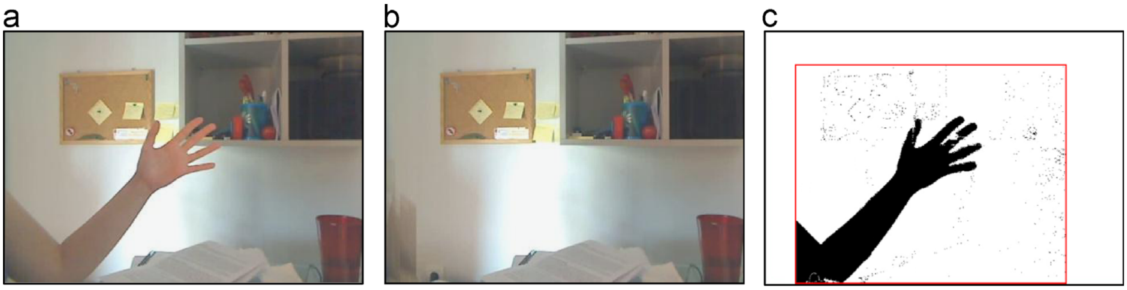

Fig. 5. Background subtraction: (a) Input video frame; (b) Background model and (c) Result.

have ever been presented (Shin et al., 2002; Xu and Zhu, 2006). In contrast, recent papers using a variety of metrics and large image datasets support the view that the presence of the luminance component significantly improves skin detection performance (Shin et al., 2002; Xu and Zhu, 2006; Phung et al., 2005; Schmugge et al., 2007). It has also been stressed that there is no optimal color space that can perform well in all metrics and that the suitability of a color space is affected by the chosen skin classifier. Possible skin classifiers are unfortunately numerous, as it has been clearly stated in the thorough review by Kakumanu et al. (2007). Thus, any decision on the best combination of color space and classification technique can be very challenging.

The choice of the skin color classifier, which is the cornerstone of our proposed technique, is based on robustness, effectiveness and low computational cost. One of the most famous classifiers is the Skin Probability Map (SPM), also known as histogram-based Bayes classification technique. It is fast and, more importantly, its performance rates are slightly higher than those of Gauss Mixture Models (GMM) or explicit thresholding techniques (Phung et al., 2005; Schmugge et al., 2007; Kakumanu et al., 2007). It has been used widely for skin detection (Jones and Rehg, 2002; Chai and Bouzerdoum, 2000; Brand and Mason, 2000; Gomez and Morales, 2002; Schwerdt and Crowley, 2000; Sigal et al., 2004). The main disadvantage of the SPM is the large training dataset that is required for generalization. In the present paper, this is tackled by using a large number of training videos. Experimental results show that the perceptual color spaces perform better with the SPM (Phung et al., 2005; Schmugge et al., 2007). Consequently, the proposed skin color segmentation technique is essentially a SPM classifier that operates in the HSV color space with the presence of the luminance component.

In order to be more robust to skin color variety, camera characteristics and illumination conditions, the SPM is enriched with an online training scheme for the skin and non-skin color maps. The outcome is a combination of both the offline and online probability maps. A detailed analysis of the proposed technique follows.

2.3.2. Skin probability map

The SPM classification technique is divided in two parts: the training procedure (offline and online) and the actual classification procedure.

2.3.2.1. Offline training. During the offline training procedure, the offline skin and non-skin color models in the HSV color space are constructed. Using skin and non-skin-colored video sequences as training data, two different histograms are calculated with 256 bins per color component. Next, the histograms bin counts are converted into probability distributions. The probability P^{off} that a given hsv color triplet belongs to the offline skin and non-skin class (also called class conditional probability) is defined as:

$$\begin{aligned} &P^{off}(hsv/skin) = (s[hsv]/Cs) \\ &P^{off}(hsv/nonskin) = (ns[hsv]/Cns) \end{aligned} \tag{6}$$

where $s[hsv]$ is the number of pixels that belong to the bin associated with the hsv color triple of the skin histogram, $ns[hsv]$ is the number of pixels that belong to the bin associated with the hsv color triple of the non-skin histogram, Cs and Cns are the total number of pixels contained in the skin and non-skin histograms, respectively.

The offline training data consist of video sequences captured by a QuickCam OrbitSphere AF of 640 × 480 pixel resolution. Fifteen (15) videos have been used for the calculation of the non-skin color histogram (26.852 frames) and contain indoor scenes with diverse complex background and lighting conditions. Thirty (30) videos have been used for the construction of the skin color histogram (55.176 frames). These videos display moving hands of different skin colors, under various lighting conditions, in front of a uniform blue background. Example frames are shown in Fig. 6(a). The background is plain, so as to achieve automatic extraction of the hand region by means of a simple algorithm which consists of only two steps: (i) grayscale conversion of the input frame, (ii) binarization through the application of the Otsu algorithm (Otsu 1979). Otsu's algorithm calculates the global optimum threshold of a bimodal histogram and it is used to convert a grayscale image to a binary form, so that the combined spread (intra-class variance) of the two classes is minimal. As shown in Fig. 6(b), the hand-class pixels are depicted in black color. The initial color values of these pixels are used during the training procedure for the calculation of the skin color probability maps. The calculation of the offline skin color histogram is achieved by using a limited number of hand region samples.

2.3.2.2. Online training. The online training procedure leads to the construction of the online skin and non-skin color models in the HSV color space. The skin and non-skin color histograms are calculated with 256 bins per color component and are converted into probability distributions. The probability P^{on} that a given hsv color triplet belongs to the online skin and non-skin class is defined as:

$$\begin{aligned} &P^{on}(hsv/skin) = \sum_n P_n^{on}(hsv/skin) = \sum_n (s_n[hsv]/Cs_n) \\ &P^{on}(hsv/nonskin) = \sum_n P_n^{on}(hsv/nonskin) = \sum_n (ns_n[hsv]/Cns_n) \end{aligned} \tag{7}$$

where n the frame.

The training data used for the calculation of the online skin and non-skin color models are the pixels of the detected hand and the rest image pixels respectively, as shown in Fig. 7. The online training lasts for a limited period, called the 'Online Training Period', in order to reduce the computational burden.

2.3.2.3. Skin color classifier. During the classification process, the class conditional probabilities P of skin and non-skin color models used are a combination both of the offline and online color maps. Specifically:

$$\begin{aligned} &P(hsv/skin) = (1-lr)P^{off}(hsv/skin) + lrP^{on}(hsv/skin) \\ &P(hsv/nonskin) = (1-lr)P^{off}(hsv/nonskin) + lrP^{on}(hsv/nonskin) \end{aligned} \tag{8}$$

where $0 \le lr \le 1$ is a learning rate that defines how fast the combination outcome adapts to the online training. Its value is calculated by the following equation:

$$lr = \frac{\text{Number of Processed Frames}}{\text{Online Training Period}} \tag{9}$$

This increases the impact of the online training to the final result as time progresses, i.e. as more frames are processed. In our current work, since the test video sequences' mean length is approximately 700 frames, the Online Training Period is set to 350 frames (around half the mean sequence length).

Finally, the skin classifier is constructed using the Bayes maximum likelihood approach (Duda et al., 2002). According to this, a given frame pixel (x, y) can be classified as skin, if:

$$\frac{P(hsv_{(x,y)}/skin)}{P(hsv_{(x,y)}/nonskin)} \ge \Theta \tag{10}$$

where Θ is a threshold, which is serves as a trade-off between true positives and false positives.

Due to the sparse distribution of skin points in HSV color space, the number of bins in the histograms can be reduced. This results into more compact histograms. According to Kakumanu et al. (2007), the number of bins giving the best performance varies with the color space representation and the size of the training dataset. In the proposed technique, 16 bins are used, because they tend to form more compact objects. Fig. 8 shows an example of the SPM classification.

2.4. Calculation of morphology weights

This algorithm uses as input the detected hand of the previous frame. On the basis of its morphology, it defines weight factors for each one of the hand pixels, so as to describe their possibility of being part of the hand in the current frame. The main idea is that the pixels of the center of the palm are more likely to belong to the hand in the current frame, as opposed to the pixels of the fingers or the pixels close to the contour. Thus, for each black pixel of the final detection output, the minimum horizontal and vertical distance from the white background is calculated. The higher the distance, the higher the possibility to belong to the hand is, thus the higher the weight. Fig. 9(c) depicts the morphology weighted

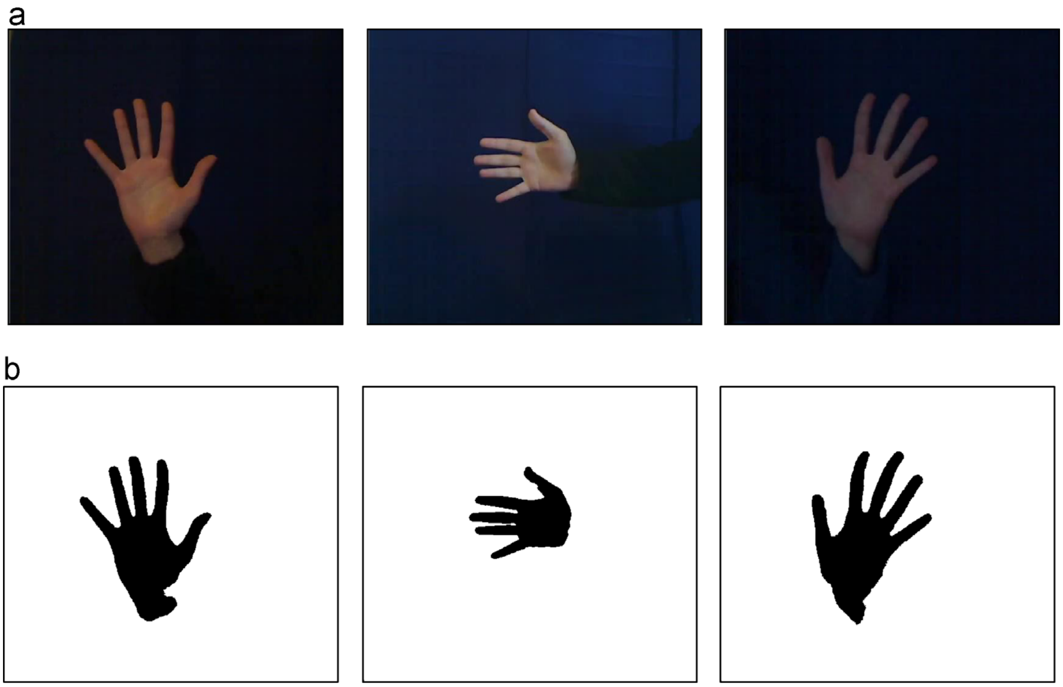

Fig. 6. Offline training of skin color maps: (a) input video frames and (b) skin color images, after the grayscale conversion and the application of the Otsu algorithm, used as training data.

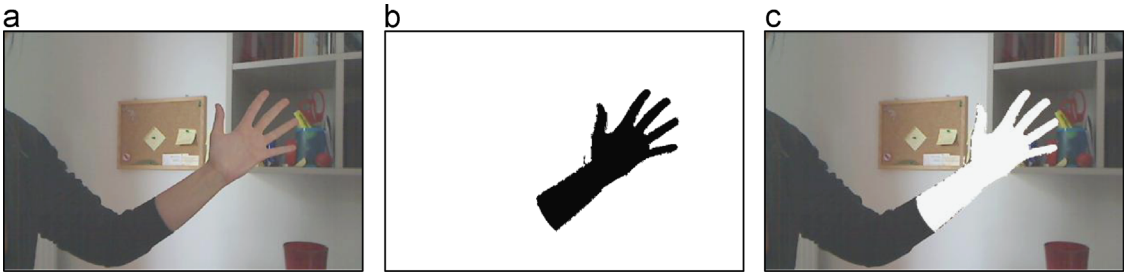

Fig. 7. Online training of skin and non-skin color maps: (a) input video frame; (b) final outcome used as training data for the skin color map; and (c) training data for the non-skin color map.

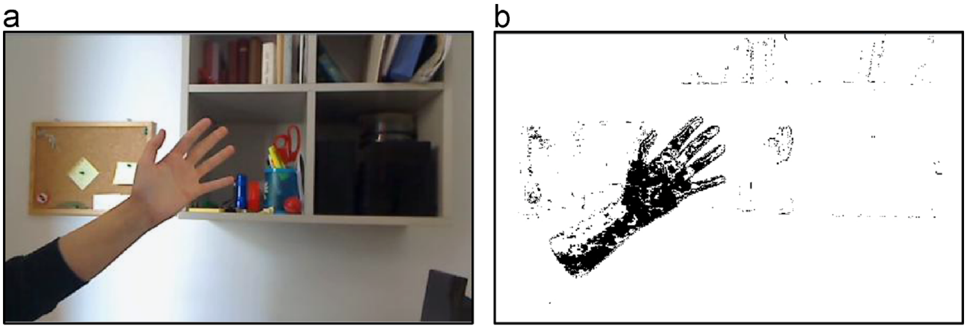

Fig. 8. Skin probability map classification: (a) input video and (b) result.

outcome that derives from the detected hand Fig. 9(b). In details, the algorithm steps are:

1. Scan the image from left to right. For each black pixel (x, y) define the distance to the contour (i.e. the first white pixel). The distance is called Horizontal Left to Right $dHLtoR(x, y)$.
2. Scan the image from right to left. Define the distance Horizontal Right to Left, $dHRtoL(x, y)$.
3. Scan the image from top to bottom. Define the distance Vertical Top to Bottom, $dVTtoB(x, y)$.
4. Scan the image from bottom to top. Define the distance Vertical Bottom to Top, $dVBtoT(x, y)$.

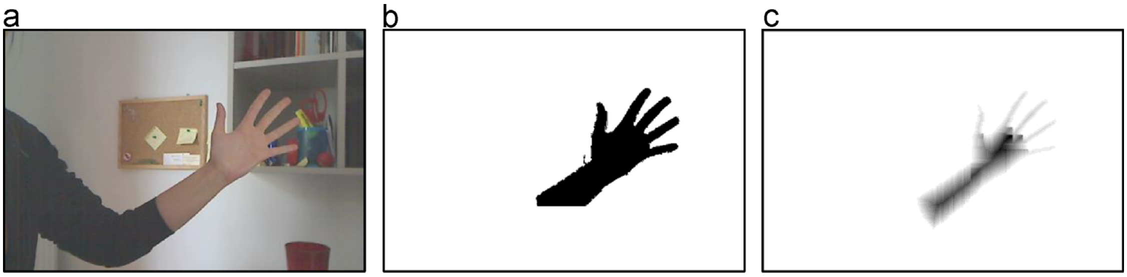

Fig. 9. Morphology weights: (a) input video frame; (b) detected hand and (c) result.

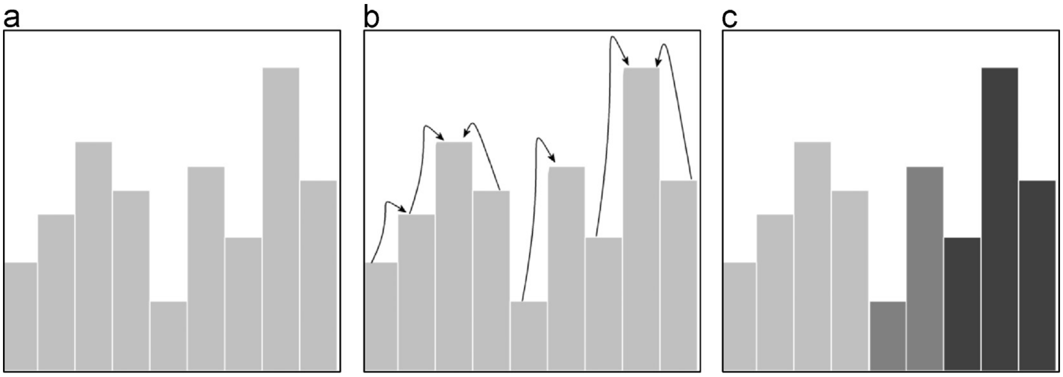

Fig. 10. Example for graph theoretical clustering applied to grayscale image (Sobottka et al., 2000): (a) histogram; (b) chains of pointers for neighborhood size 1 and (c) Clusters.

5. The morphology weight for a pixel (x, y) is calculated according to the equation:

$$morphW(x,y) = \frac{\min(dHLtoR(x,y), dHRtoL(x,y), dVTtoB(x,y), dVBtoT(x,y))}{\max(\min(\overrightarrow{dHLtoR}, \overrightarrow{dHRtoL}, \overrightarrow{dVTtoB}, \overrightarrow{dVBtoT}))} \tag{11}$$

2.5. Combination of information

The final stage of the proposed hand detection method is the combination of the information derived from the four aforementioned steps: motion detection, skin color detection, morphology weights and skin-colored background. The innovation of the combination method is that it employs region processing in order to achieve higher robustness. Instead of a simple window-based region, an arbitrarily shaped area of neighboring pixels with similar color is used. More specifically, the image is oversegmented into similar color regions by applying a color reduction algorithm, named graph theoretical clustering by Matas and Kittler (1995) and Sobottka et al. (2000).

2.5.1. Graph theoretical clustering

Graph theoretical clustering is an unsupervised clustering algorithm and can be described as follows:

Firstly, the image histogram of the input frame is calculated. The histogram is then divided into bins of specified size (Fig. 10(a)). For each bin of the histogram, a pointer to its largest bin in a given neighborhood is stored (Fig. 10(b)). When all pointers are set, the histogram contains chains of bins pointing to a local maximum. The set of bins belonging to such a chain, build a cluster (Fig. 10(c)). The example shown in Fig. 10 relates to the case of a grayscale image.

The graph-theoretical clustering algorithm requires two parameters: the size of a histogram bin (histogram quantization) and the size of the neighborhood, when searching for the largest bin. For example, in a 3D color histogram and for neighborhood size equal to 1, the algorithm searches for the maximum in 26 neighboring bins.

In the proposed method, the choice of these parameters depends on the following criteria:

- The hand should not merge with the background.
- The objects should consist of at least one region.
- The computational cost should be as low as possible.

These criteria are met, when the histogram bin size is 64 and the neighborhood size is 1.

In our current work, the graph theoretical algorithm uses the RGB color space and so a 3D-color histogram is constructed. It should be noted that this is applied only on the mROI, defined during the motion detection stage, in order to reduce the computational cost and to achieve better results by removing the redundant information of the non-moving area. An example is shown in Fig. 11.

2.5.2. Region based combination

The extracted information of motion, skin color and morphology is combined in each one of the regions created through color reduction. In particular, the regions are rated and this rate describes the probability that they belong to the hand.

More specifically, every region R_i is rated by examining the pixels it contains ($(x, y) \in R_i$) and by applying the following rules:

1. Let $MD(x, y)$ be a pixel of the output of motion detection. If the pixel is black then its value is 1, otherwise its value is equal to 0 (Fig. 12(b)).
2. Let $SCD(x, y)$ be a pixel of the output of skin color detection. If the pixel is black then its value is 1, otherwise its value is equal to 0 (Fig. 12(c)).

3. The range of the morphology weights is $0 \le morphW(x,y) \le 1$ (Fig. 12(d)).
4. Let $sB(x,y)$ be a pixel of the skin-colored background model. Its value range is $0 \le sB(x,y) \le 1$.

The total rate of a region R_i is calculated by the equation:

$$rate(R_i) = \frac{\sum_{(x,y) \in R_i} (MD(x,y) + SCD(x,y) + morphW(x,y) - sB(x,y))}{3Size(R_i)} \tag{12}$$

where $Size(R_i)$ is the total number of pixel that the region contains.

That is to say that a region is rewarded for every black $MD(x,y)$ and $SCD(x,y)$ pixel it contains, as well as for every pixel that belongs to the detected hand in the previous frame. In contrast, it is penalized for every $sB(x,y)$ pixel, in order to deal with the problem of false positives created by the skin-colored objects of the background. The result of the rating system is a grayscale image (Fig. 12(e)) that depicts the possibility of each region to belong to the hand. Black pixels represent possibility equal to 1, whereas white pixels represent possibility equal to 0. The image is binarized by applying the Otsu algorithm (Otsu, 1979), in order to define the regions possessing a higher certainty of being hand regions. The binary image contains small black regions that do not

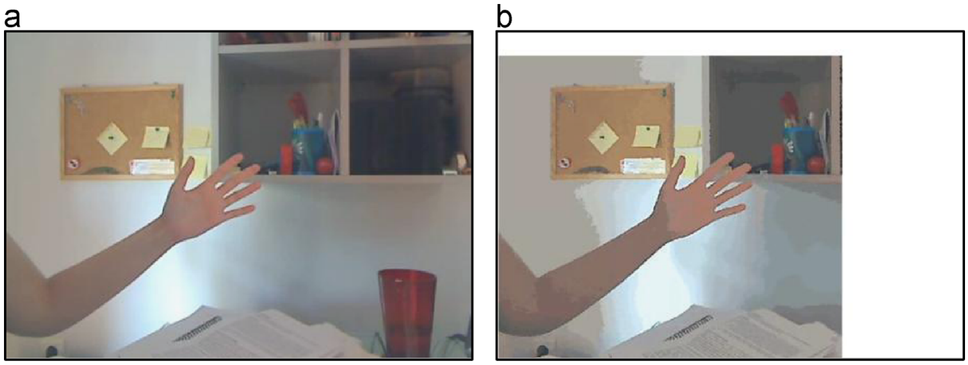

Fig. 11. Color reduction by applying the graph theoretical algorithm: (a) Input video frame and (b) color reduced image.

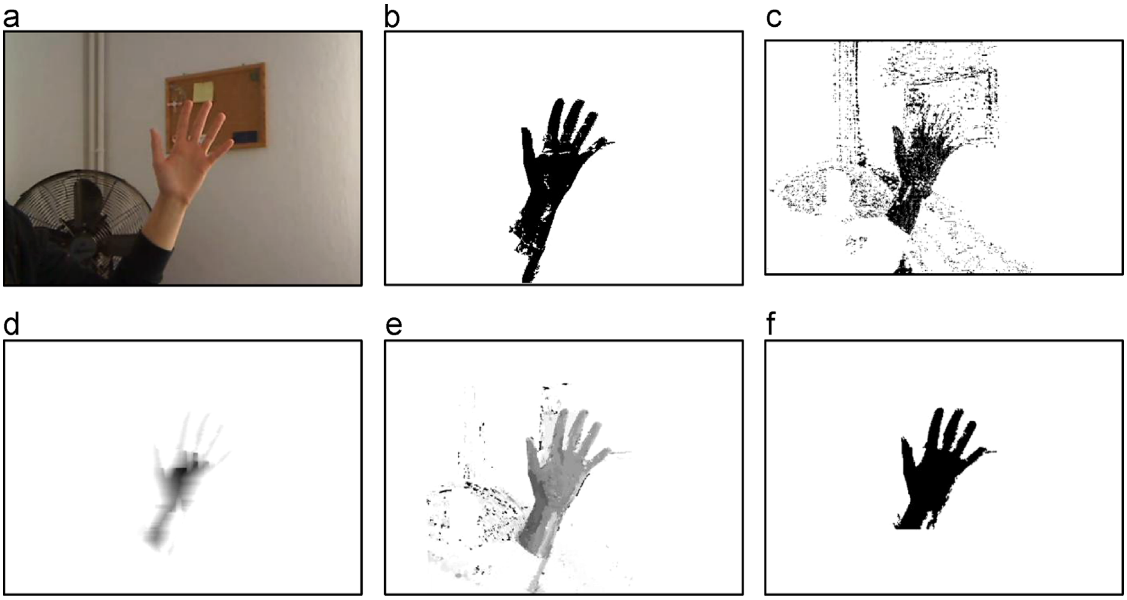

Fig. 12. Region based combination: (a) input video frame; (b) motion detection; (c) skin color detection; (d) morphology weights; (e) grayscale image resulting from the rating of the regions; (f) final detected hand.

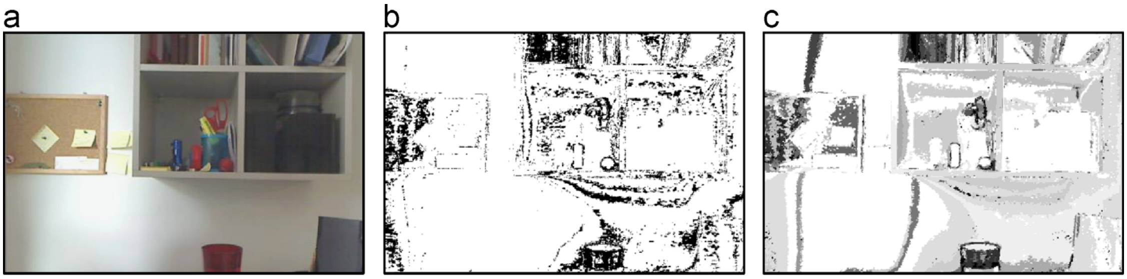

Fig. 13. (a) Background model; (b) pixelwise skin-colored background model and (c) region-based skin-colored background model.

belong to the hand and so are removed through size filtering. The output image is the final outcome of the system that represents the detected hand (Fig. 12(f)).

2.6. Skin-colored background model

The skin-colored background model used during the stages of motion detection and combination of information is created based on: (i) the background model of the background subtraction algorithm, (ii) color reduction and (iii) skin probability map classification. It is actually the outcome of a region-based skin color detection algorithm applied on the background model.

More specifically, let $\overrightarrow{B}_n$ (Fig. 13(a)) be the background model at time n defined during the background subtraction algorithm, described in Section 2.2.3. The algorithm steps for the construction of the skin-colored background model $\overrightarrow{sB}_n$ are:

1. Perform skin color detection through the classifier described in Section 2.3.2, in order to create the pixelwise skin-colored background model $\overrightarrow{psB}_n$ (Fig. 13(b))
2. Perform color reduction through graph theoretical algorithm (Section 2.5.1), so as to segment the background model in regions of similar color. The outcome is called color reduced background, $\overrightarrow{crB}_n$.
3. For every region R_i of the $\overrightarrow{crB}_n$, the percentage of skin pixels is calculated (black pixels of the $\overrightarrow{psB}_n$)

$$skinPercentage(R_i) = 100\frac{\sum_{(x,y)\in R_i}\overrightarrow{psB}_n(x,y)}{Size(R_i)},$$
$$\text{where}\overrightarrow{psB}_n(x,y) = \begin{cases} 1, & \text{ifblack} \\ 0, & \text{ifwhite} \end{cases} \tag{13}$$

4. A region based skin-colored background model $\overrightarrow{rsB}_n$ (Fig. 13(c)) is created by assigning each pixel with the value of the skin percentage of the region it belongs

$$\overrightarrow{rsB}_n(x,y) = skinPercentage_n(R_i), \text{where } (x,y)\in R_i \tag{14}$$

5. Finally, the skin-colored background model is created as follows:

$$\overrightarrow{sB}_n(x,y) = (1-\beta)\overrightarrow{sB}_{n-1}(x,y) + \beta\overrightarrow{rsB}_n(x,y) \tag{15}$$

where β is an adaptation factor, taken equal to 0.7.

3. Experimental results

In order to determine the performance of the proposed hand detection method, an evaluation was conducted. The evaluation video set consists of 45 video sequences (32.182 frames) of a hand moving in front of different cluttered backgrounds and under a variety of lighting conditions, captured by a QuickCam OrbitSphere AF web camera with 640 × 480 resolution. The hand detection system was implemented in the Delphi RAD environment.

3.1. Experiment 1: individual stages output

Experiment 1 shows the results of the individual stages of the proposed hand detection method in a complex background. Fig. 14 depicts sample frames of an input video sequence and the final outcome. As one can observe, the background is much cluttered, consisting of many objects with skin-like color. Nonetheless, the hand is detected and also the palm shape and the fingers are well defined. The entire video containing the hand segmentation results can be found online here (Test Video Sequence 1).

Fig. 15 presents the image differencing results. In this sequence, the hand moves from left to right. In the beginning of the sequence, shown in Fig. 15(a), image differencing produces a compactly detected hand. As the hand progresses, “holes” are created in the moving area, due to hand overlapping between successive frames (Fig. 15(b)). Finally, in Fig. 15(c), the hand stops moving.

Fig. 16 shows the results of background subtraction. The detected hand is more compact compared to image differencing results. The modified background subtraction algorithm reduces the problem of false negatives, which is created when the hand and background objects with skin color overlap, but it does not eliminate it completely (Fig. 16(a) and (b)).

In Fig. 17(a), the estimated background model is presented over time. As it can be seen, it is a good approximation of the real background. Fig. 17(b) shows the skin-colored background model over time.

Fig. 18 shows the results of the SPM technique at frames $n=20$, $n=100$, $n=600$. As it can be observed, the technique adapts to user's skin color, the background colors and the lighting conditions. Thus, the results tend to improve over time. In particular, the false positives rate reduces while the true positives rate increases. This improvement is due to the proposed online training procedure.

Fig. 19(b) demonstrates the morphology weights created based on the detected hand of Fig. 19(a). They describe the probability that a pixel is part of the hand in the current frame.

Fig. 20 shows the outcome of the color reduction step. The graph theoretical clustering is applied on the mROI. The input video frames are oversegmented and divided into similar-color areas, which are used for the region-based comparison stage. As it can be observed, the hand is not merged with the background and the objects consist of at least one region.

Fig. 21(a) presents the grayscale images created by the application of a rating system on the similar colored regions. Fig. 21(b) shows the final detected hand after the application of the Otsu binarization and size filtering. As can be observed in the leftmost Fig. 21(a), background regions are rated as possible to be part of the hand. The Otsu algorithm succeeds in segmenting the areas of higher certainty from those of lower, thus succeeding in extracting the hand.

3.2. Experiment 2: validation

Due to the nature of the input data (live data/video streaming), the construction of a ground truth set is not feasible. Furthermore, the size of the evaluation set (32.182 frames) is too large to allow a typical pixel-level rating of the system. Therefore, a task-oriented evaluation process was chosen, which analyzes the system performance based on the quality of its results. A result is considered correct, if it can be used as an input to the features extraction stage of an overall hand gesture recognition system. In other words, the hand detection system is assessed by measuring how well it prepares the input (hand) for the gesture recognition task.

More specifically, each frame from the 45 videos in the evaluation set is examined manually. The system outcome is classified as either a successful or an erroneous detection result. A system output is considered successful, if both the shape of the palm and the raised fingers are well defined. In contrast, a detection result is considered inaccurate, if any of the following error types occurs:

- **Error type 1:** Poorly extracted palm region, containing large white areas (false negatives) or palm region not detected at all.
- **Error type 2:** Raised fingers not detected.

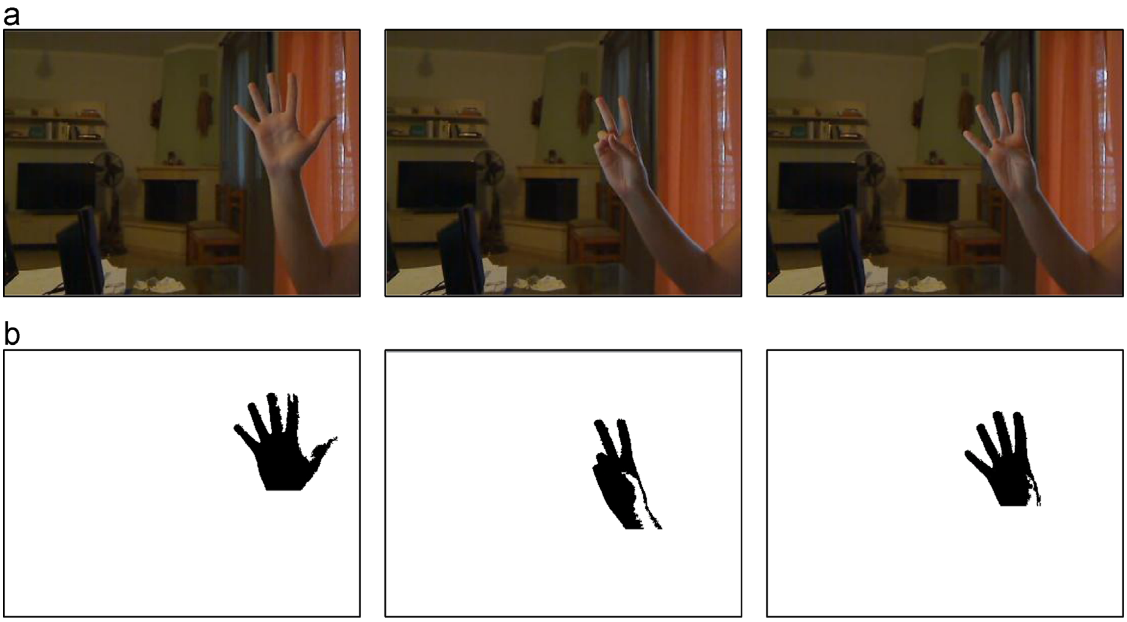

Fig. 14. (a) Input video frames and (b) final outcome – detected hand (Test Video Sequence 1).

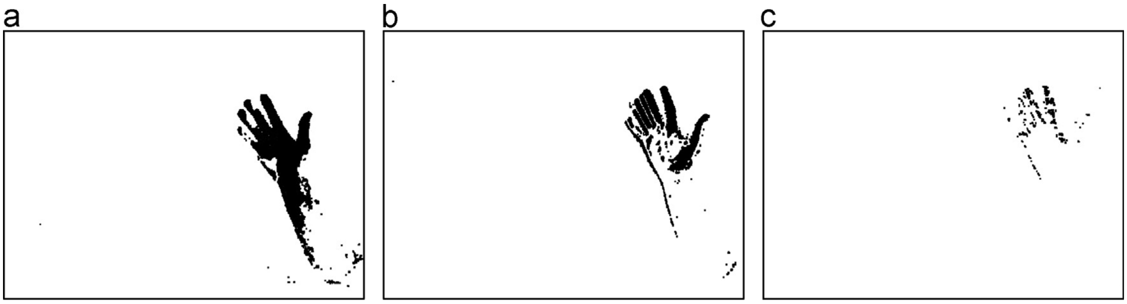

Fig. 15. Image differencing.

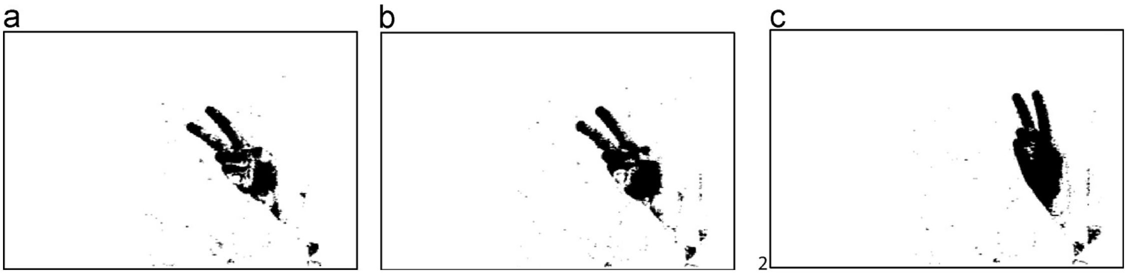

Fig. 16. Background subtraction.

- **Error type 3:** Total hand detection failure, or combination of errors 1 and 2.
- **Error type 4:** Classification of background objects as hand (false positives).

Fig. 22 presents typical examples of the four error types.

The manual examination of each of the 32.182 frames, has led to a rate of 88.02% successfully detected hands. Table 1 summarizes the results and presents the frequency of the four error types.

Error type 2, which corresponds to the failure to detect at least one of the raised fingers, is the most frequent mistake. This can be explained by the fact that fingers consist of small regions and thus are more susceptible to inaccuracies in the motion detection or the skin detection steps, compared to the larger regions of the palm. Error type 4, which refers to the false detection of background objects as hand, is the second most common inaccuracy. It mainly occurs due to the rating system, used in the combination of information stage, which occasionally fails to discriminate an overlapping hand from the skin-colored background objects. Third in frequency is the Error type 1, which defines the failure to extract a well-shaped palm. This mistake occurs when the hand overlaps with a skin-colored background object and the motion detection technique performs inadequately. Also, this error type occurs because of an inaccurate definition of the mROI. Finally, type 3 errors, i.e. total hand detection failure, occur very rarely. The main cause of these errors is the erroneous definition of the mROI during the motion detection. Figs. 23–26 show example frames from the evaluation video sequences.

The evaluation video sequence in (Test Video Sequence 2), (see Fig. 23), displays a scene with regular indoor illumination conditions and background objects featuring colors similar to the skin

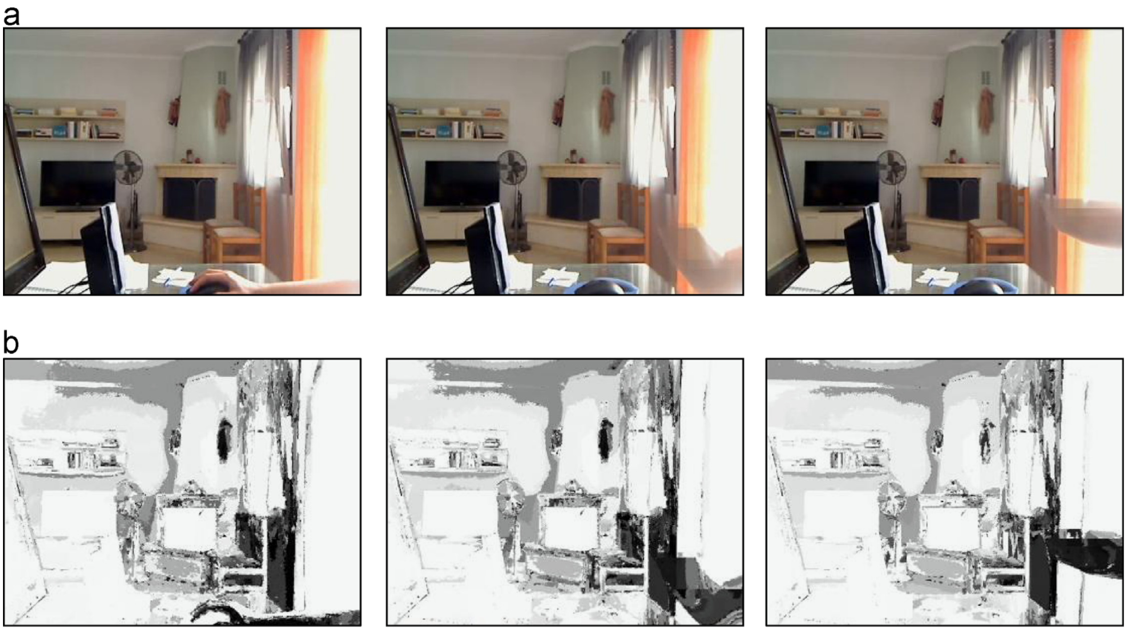

Fig. 17. (a) Background model and (b) Skin-colored background model.

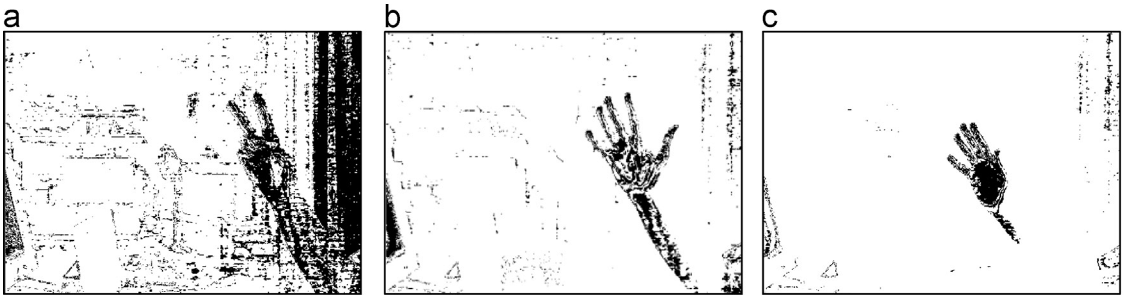

Fig. 18. Skin detection using the skin probability map technique: (a) Frame 20; (b) Frame 100 and (c) Frame 600.

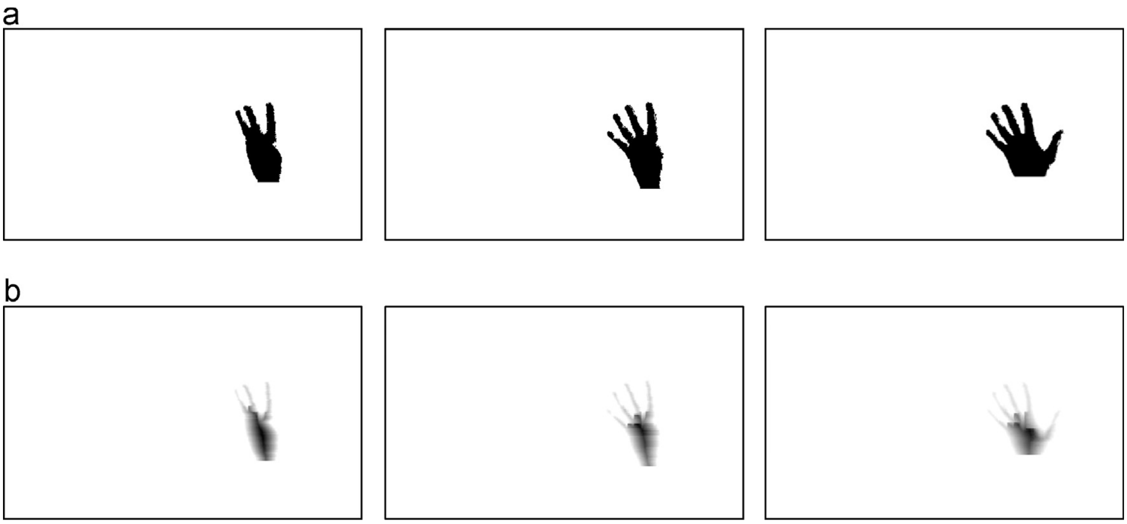

Fig. 19. Morphology weights: (a) detected hand and (b) result.

(yellow, orange). The hand is detected successfully throughout the video, except from a few frames, where the mROI is inaccurately estimated and thus causes partial cut of the fingers. Table 2 presents the evaluation results for this video sequence.

The main characteristic of the video sequence in (Test Video Sequence 3), (see Fig. 24), is the very intense illumination conditions. Many hand regions seem “saturated”. Nonetheless, the detection rate is 89.66%, as shown in Table 3, suggesting that the system is adaptive and thus robust to lighting conditions. The most frequent error here is the failure to detect all the raised fingers, which is caused by skin color detection mistakes.

In video sequence (Test Video Sequence 4), (see Fig. 25), the background is very cluttered and contains multiple objects of similar color to the skin. In addition, the scene is poorly

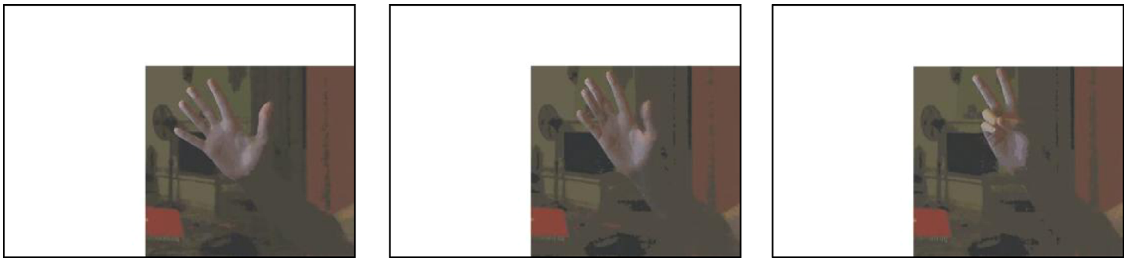

Fig. 20. Color reduction applied on the mROI.

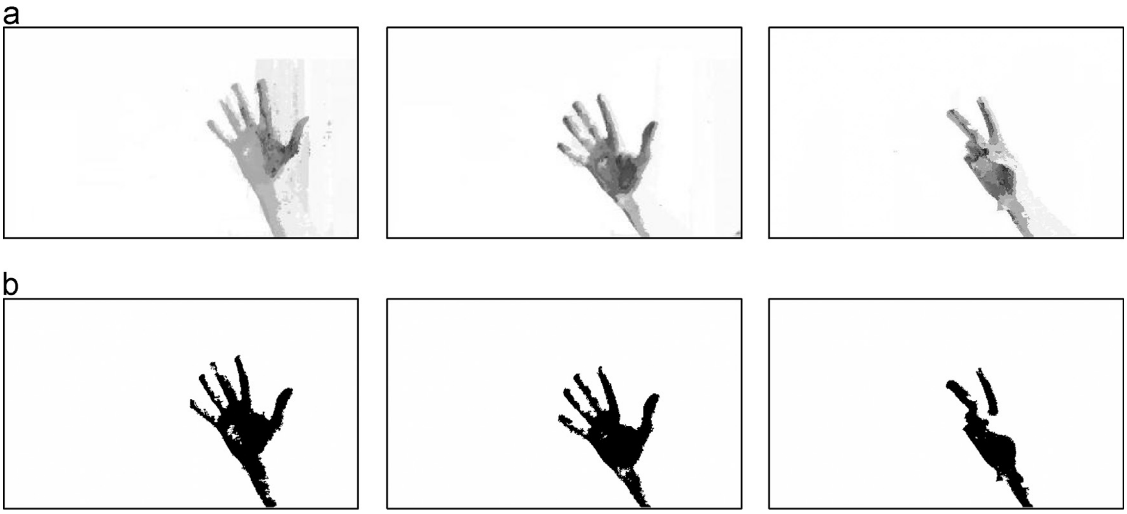

Fig. 21. Region-based combination: (a) Grayscale images resulting from the rating of the regions and (b) Final detected hand.

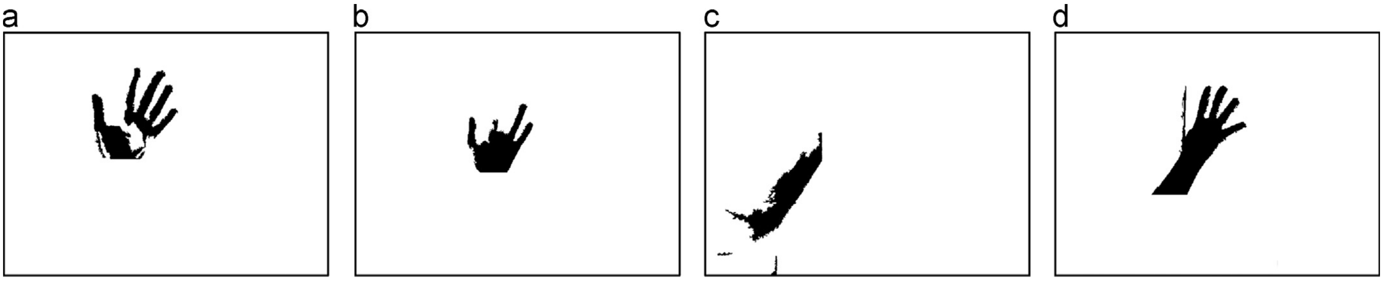

Fig. 22. Categories of erroneous results: (a) poorly detected palm area – error type 1; (b) missing raised fingers – error type 2; (c) missing hand – error type 3 and (d) detection of hand objects as hand – error type 4.

Table 1
Evaluation results.

Number of frames	Success rate (%)	Error type 1 rate (%)	Error type 2 rate (%)	Error type 3 rate (%)	Error type 4 rate (%)
32.182	88.02	2.78	4.93	1.25	3.02

illuminated. Given these circumstances, the success rate of 80.45% (Table 4) is very promising and so this video sequence is considered to be a representative example of the effectiveness of the proposed hand detection system. The most common mistake in this video is the detection of background objects as hand (Type 3). This occurs mainly when the hand overlaps with the window and is caused by shortcomings of the rating system during the combination step.

In video sequence (Test Video Sequence 5), (see Fig. 26), the scene is illuminated by a warm lighting source that makes the colors of the background appear close to orange and thus similar to the skin. The hand is detected successfully, with a rate of 87.72% and the two most common mistakes are the failure to detect all the raised fingers and the extraction of background objects as hand areas, however, at very low rates. Analytical evaluation results are presented in Table 5.

3.3. Comparative results

The proposed technique has been compared with other similar approaches and more specifically with the techniques described in the papers (Wilson and Salgian, 2008; Chen et al., 2003; Dardas and Georganas, 2011; Mao et al., 2009; Okkonen et al., 2007). We have found that only these techniques provide experimental results for the hand detection stage. It should be noted that there is no common database or test dataset that can be used in a comparison procedure. Moreover, we cannot compare the internal stages of our technique, because such data were not provided by other techniques. Another difficulty encountered is the different way of measuring the accuracy by each method, due to the different approaches and applications of hand detection (e.g. dynamic or static gesture recognition). Finally, no software

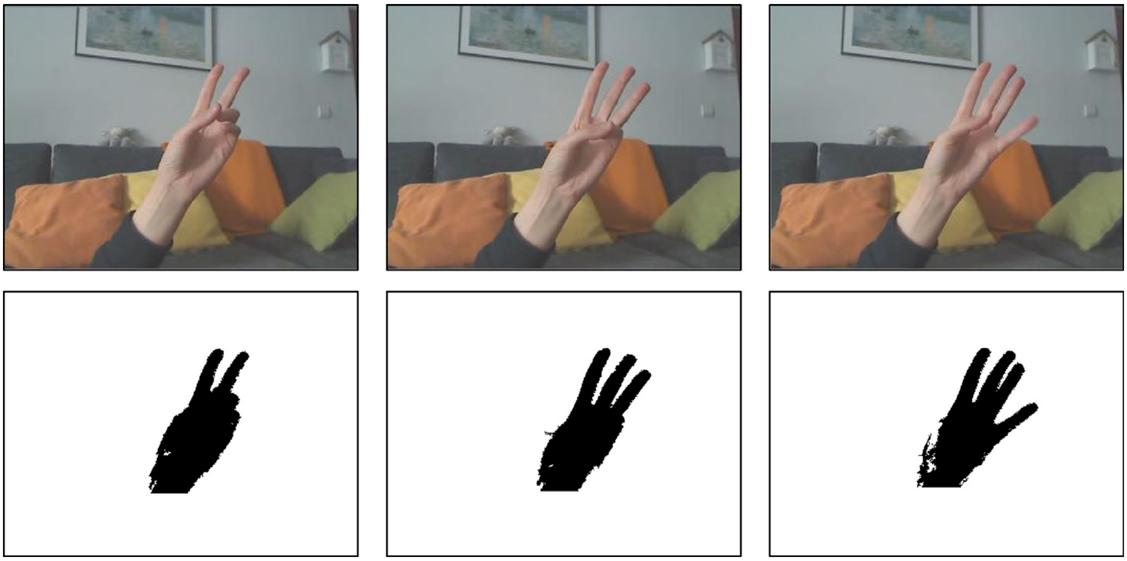

Fig. 23. Input frames and final outputs of the detected hand. The video sequence can be found in (Test Video Sequence 2).

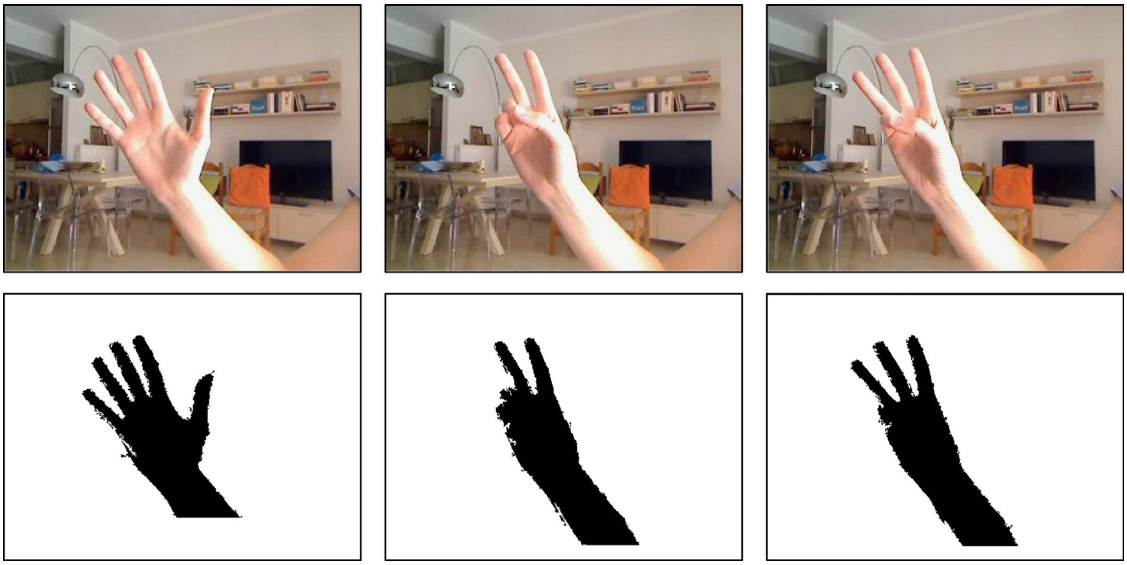

Fig. 24. Input frames and final outputs of the detected hand. The video sequence can be found in (Test Video Sequence 3).

implementing these techniques was publicly available. As mentioned in the introduction, we can divide the methods into two categories: those that obtain the region containing the hand and those that extract the exact shape of the hand. In our case, we obtain evaluation results for both, hand region detection and shape extraction. The results are outlined in Table 6.

Given the evaluation results (Table 1), our technique presents a high accurate detection rate for the hand region, since the error type 3, which defines total hand detection failure, is only 1.25%. The proposed technique gives promising results compared to other techniques. Comparing our method to Wilson and Salgian (2008) and Chen et al. (2003), which use similar approaches, it can be seen that the detection rate is improved, proving the efficiency of the proposed combination of methods (Table 6).

In addition, the success rate of our technique for shape detection is 88.02%, as the hand shape error is the sum of errors type 1, 2, 3, 4. Compared to the methods described in Dardas and Georganas (2011) and Okkonen et al. (2007), the proposed technique presents higher accuracy rates, which is very promising. Compared to the method described in Mao et al. (2009), our method suffers slightly due to the use of the robust object detector, but, as mentioned in Guo et al. (2012), false positive rates of object detectors may depend highly on the complexity of background, making difficult the comparison between different datasets.

4. Conclusions

This paper proposes a new method for real-time hand detection, which combines motion, skin color and morphology features, in order to achieve robustness and effectiveness. Its main novelty lies in the region-based combination of these features.

Firstly, a hybrid motion detection technique is applied that uses image differencing and background subtraction. The method adapts quickly to the changes of the scene and defines the area (mROI) and the shape of the hand with satisfactory accuracy. Secondly, skin detection is implemented through a modified version of the Skin Probability Map classification that employs an online color map training procedure. As a result, the technique becomes adaptive to the user's individual skin color, the background colors and the illumination conditions. Thirdly, morphology

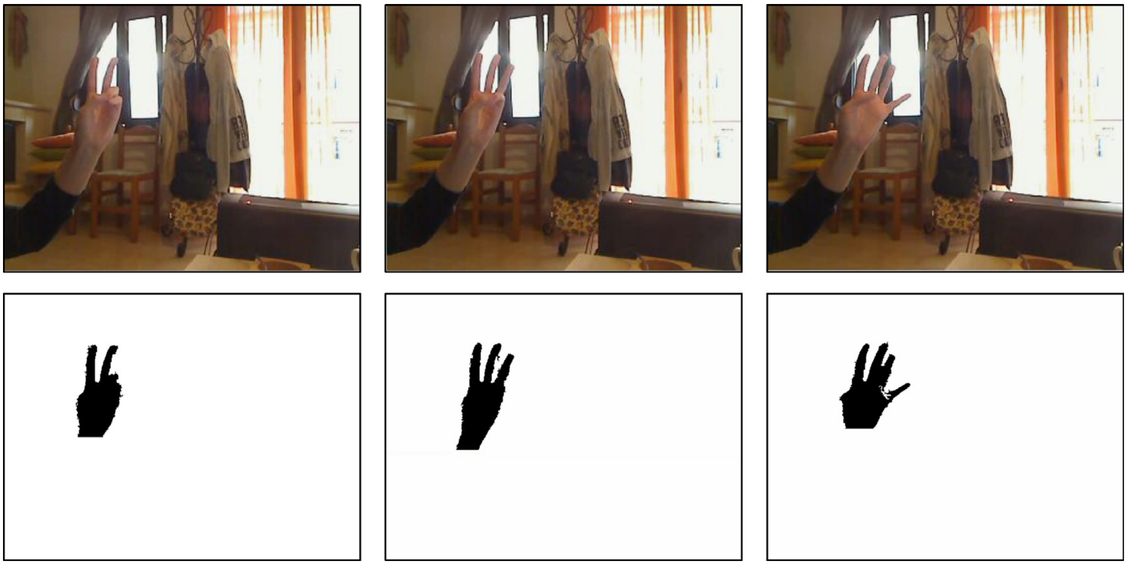

Fig. 25. Input frames and final outputs of the detected hand. The video sequence can be found in (Test Video Sequence 4).

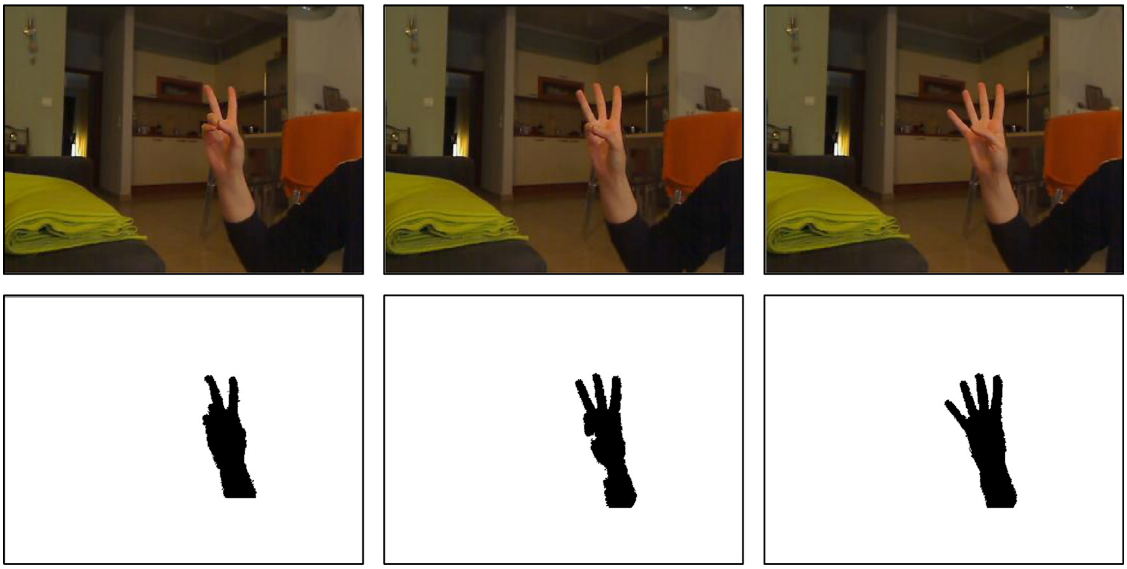

Fig. 26. Input frames and final outputs of the detected hand. The video sequence can be found in (Test Video Sequence 5).

Table 2
Evaluation results of the video sequence found in (Test Video Sequence 2).

Number of frames	Success rate (%)	Error type 1 rate (%)	Error type 2 rate (%)	Error type 3 rate (%)	Error type 4 rate (%)
395	98.98	0	1.02	0	0

Table 3
Evaluation results of the video sequence found in (Test Video Sequence 3).

Number of frames	Success rate (%)	Error type 1 rate (%)	Error type 2 rate (%)	Error type 3 rate (%)	Error type 4 rate (%)
483	89.66	2.07	6.41	0	1.86

Table 4
Evaluation results of the video sequence found in (Test Video Sequence 4).

Number of frames	Success rate (%)	Error type 1 rate (%)	Error type 2 rate (%)	Error type 3 rate (%)	Error type 4 rate (%)
405	80.45	0	0	19.55	0

Table 5
Evaluation results of the video sequence found in (Test Video Sequence 5).

Number of frames	Success rate (%)	Error type 1 rate (%)	Error type 2 rate (%)	Error type 3 rate (%)	Error type 4 rate (%)
814	87.72	0	6.38	0	5.9

weights are calculated based on the final detected hand of the previous frame. Finally, the input frame is divided into uniformly colored areas and a system, that combines the extracted information, rates their possibility to be part of the hand. The outcome is binarized, the arm is removed and the final output of the detected hand is created.

The technique has been extensively tested in a variety of backgrounds and illumination conditions. It has demonstrated great robustness in much cluttered backgrounds and under

Table 6
Comparative results.

Paper	Technique	Accuracy (Region detection)	Accuracy (Shape detection)
Wilson and Salgian (2008)	Background subtraction and offline trained Bayes Classifier and Gaussian Mixture Models for skin color	98.26%	N/A
Chen et al. (2003)	Image differencing and online skin color training	93.00%	N/A
Dardas and Georganas (2011)	Skin color filtering and contour templates comparison	N/A	70.00%
Mao et al. (2009)	Viola and Jones object detector combined with skin color filtering	N/A	**90.00%**
Okkonen et al. (2007)	Background subtraction and histogram based skin color segmentation	N/A	87.50%
Proposed	Image differencing and background subtraction, online and offline trained skin color classifier and morphological features	**98.75%**	88.02%

difficult lighting circumstances. The achieved hand region detection rate is promising at 98.75%. For future work, the proposed technique can be extended by incorporating depth information obtained by an RGB-ToF camera, such as the Microsoft Kinect, to improve hand detection.

Acknowledgment

This research has been co-financed by the European Union (European Social Fund - ESF) and Greek National Funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF)–Research Funding Program: THALES (MIS 379516) Investing in knowledge society through the European Social Fund.

References

Alexander, T., Ahmed, H., Anagnostopoulos, G., 2009. An open source framework for real-time, incremental, static and dynamic hand gesture learning and recognition. In: Jacko, J. (Ed.), Human-Computer Interaction Novel Interaction Methods and Techniques, 5611. Springer, Berlin - Heidelberg, pp. 123–130.

Alon, J., Athitsos, V., Yuan, Q., Sclaroff, S., 2005. Simultaneous localization and recognition of dynamic hand gestures. In: Seventh IEEE Workshops on Application of Computer Vision WACV/MOTIONS '05, vol. 1, Breckenridge, CO, USA, 2005, vol. 2, pp. 254–260.

Brand, J., Mason, J.S., 2000. A comparative assessment of three approaches to pixel-level human skin-detection. In: Proceedings of the 15th International Conference on Pattern Recognition, Barcelona, Spain, vol. 1, pp. 1056–1059 vol.1.

Chai, D., Bouzerdoum, A., 2000. A Bayesian approach to skin color classification in YCbCr color space. In: Proceedings of the TENCON, Kuala Lumpur, Malaysia, vol. 2, pp. 421–424.

Chen, F.-S., Fu, C.-M., Huang, C.-L., 2003. Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vision Comput. 21 (8), 745–758.

Chua, C.-S., Guan, H., Ho, Y.-K., 2002. Model-based 3D hand posture estimation from a single 2D image,. Image Vision Comput. 20 (3), 191–202.

Collins, R.T., Lipton, A.J., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., Burt, P., Wixson, L., 2000. A System for Video Surveillance and Monitoring. Carnegie Mellon University: The Robotics Institute.

Dadgostar, F., Sarrafzadeh, A., Messom, C., 2009. Multi-layered hand and face tracking for real-time gesture recognition. In: Köppen, M., Kasabov, N., Coghill, G. (Eds.), Adv. Neuro-Inf. Process., 5506. Springer, Berlin - Heidelberg, pp. 587–594.

Dardas, N.H., Georganas, N.D., 2011. Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans. Instrum. Meas. 60 (11), 3592–3607.

Donoser, M., H., Bischof, 2008. Real time appearance based hand tracking, In: 19th International Conference on Pattern Recognition. ICPR, pp. 1–4.

Duda, R.O., Hart, P.E., Stork, D.G., 2002. Pattern classification. Wiley Interscience, New York.

Ebert, A., Gershon, N.D., van der Veer, G.C., 2012. Human-computer interaction. Künstl Intell 26 (2), 121–126.

Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X., 2007. Vision-based hand pose estimation: a review. Comput. Vision Image Underst. 108 (1–2), 52–73.

Flasiński, M., Myśliński, S., 2010. On the use of graph parsing for recognition of isolated hand postures of Polish Sign Language. Pattern Recog. 43 (6), 2249–2264.

Ge, S.S., Yang, Y., Lee, T.H., 2008. Hand gesture recognition and tracking based on distributed locally linear embedding. Image Vision Comput. 26 (12), 1607–1620.

G. Gomez, E.F. Morales, 2002. Automatic feature construction and a simple rule induction algorithm for skin detection, In: Proceeding of the ICML Workshop on Machine Learning in Computer Vision, pp. 31–38.

Guo, J.-M., Liu, Y.-F., Chang, C.-H., Nguyen, H.-S., 2012. Improved hand tracking system. IEEE Trans. Circuits Syst. Video Technol. 22 (5), 693–701.

Jones, M.J., Rehg, J.M., 2002. Statistical color models with application to skin detection. Int. J. Comput. Vision 46 (1), 81–96.

Kakumanu, P., Makrogiannis, S., Bourbakis, N., 2007. A survey of skin-color modeling and detection methods. Pattern Recog. 40 (3), 1106–1122.

J.Y. Lee, S.I. Yoo, 2002. An Elliptical Boundary Model for Skin Color Detection, In: Proceedings of International Conference on Imaging Science, System and Technology, Las Vegas, Nevada, USA.

Lee, Y., 2008. Application of the particle filter for simple gesture recognition. In: Huang, D.-S., Wunsch, D., Levine, D., Jo, K.-H. (Eds.), Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence, 5227. Springer, Berlin - Heidelberg, pp. 534–540.

Mao, G.-Z., Wu, Y.-L., Hor, M.-K., Tang, C.-Y., Real-time hand detection and tracking against complex background. In: Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009. IIH-MSP '09, 2009, pp. 905–908.

Matas, J., Kittler, J., 1995. Spatial and feature space clustering: applications in image analysis. In: Hlavác, V., Šára, R. (Eds.), Computer Analysis of Images and Patterns, 970. Springer, Berlin - Heidelberg, pp. 162–173.

Moeslund, T.B., Hilton, A., Krüger, V., 2006. A survey of advances in vision-based human motion capture and analysis. Comput. Vision Image Underst. 104 (2–3), 90–126.

Okkonen, M.-A., Kellokumpu, V., Pietikäinen, M., Heikkilä, J., 2007. A visual system for hand gesture recognition in human-computer interaction. In: Ersbøll, B., Pedersen, K. (Eds.), Image Anal., 4522. Springer, Berlin - Heidelberg, pp. 709–718.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Trans. Syst., Man Cybern. 9 (1), 62–66.

Phung, S.L., Bouzerdoum, A., Chai, D., 2005. Skin segmentation using color pixel classification: analysis and comparison. IEEE Trans. Pattern Anal. Mach. Intell. 27 (1), 148–154.

Schmugge, S.J., Jayaram, S., Shin, M.C., Tsap, L.V., 2007. Objective evaluation of approaches of skin detection using ROC analysis. Comput. Vision Image Underst. 108 (1–2), 41–51.

Schwerdt, K., Crowley, J.L., 2000. Robust face tracking using color. In: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, France, pp. 90–95.

Shimada, N., Kimura, K., Shirai, Y., 2001. Real-time 3D hand posture estimation based on 2D appearance retrieval using monocular camera, In: Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, Vancouver, BC, Canada, pp. 23–30.

Shin, M.C., Chang, K.I., Tsap, L.V., 2002. Does colorspace transformation make any difference on skin detection? In: Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision (WACV 2002), Orlando, Florida, USA, pp. 275–279.

Sigal, L., Sclaroff, S., Athitsos, V., 2004. Skin color-based video segmentation under time-varying illumination,. IEEE Trans. Pattern Anal. Mach. Intell. 26 (7), 862–877.

Sobottka, K., Kronenberg, H., Perroud, T., Bunke, H., 2000. Text extraction from colored book and journal covers,. Int. J. Doc. Anal. Recog. 2 (4), 163–176.

Stergiopoulou, E., Papamarkos, N., 2009. Hand gesture recognition using a neural network shape fitting technique,. Eng. Appl. Artif. Intell. 22 (8), 1141–1158.

Test Video Sequence 1 ⟨http://www.youtube.com/watch?v=3F8lUpaaWWI⟩.

Test Video Sequence 2 ⟨http://www.youtube.com/watch?v=CwPqaiBFdeY⟩.

Test Video Sequence 3 ⟨http://www.youtube.com/watch?v=fLV69KV7kDg⟩.

Test Video Sequence 4 ⟨http://www.youtube.com/watch?v=Sh__cnL_o6E⟩.

Test Video Sequence 5 ⟨http://www.youtube.com/watch?v=v2TrRRi5wtY⟩.

Ueda, E., Matsumoto, Y., Imai, M., Ogasawara, T., 2003. A hand-pose estimation for vision-based human interfaces. IEEE Trans. Ind. Electron. 50 (4), 676–684.

Vatavu, R.-D., Grisoni, L., Pentiuc, S.-G., 2009. Gesture recognition based on elastic deformation energies. In: Sales Dias, M., Gibet, S., Wanderley, M., Bastos, R.

(Eds.), Gesture-Based Human-Computer Interaction and Simulation, 5085. Springer, Berlin - Heidelberg, pp. 1–12.

Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features, In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, Kauai, HI, USA, pp. I-511–I-518.

Wachs, J.P., Kölsch, M., Stern, H., Edan, Y., 2011. Vision-based hand-gesture applications. Commun. ACM 54 (2), 60–71.

Wilkowski, A., 2009. HMM-based system for recognizing gestures in image sequences and its application in continuous gesture recognition. In: Hippe, Z., Kulikowski, J. (Eds.), Human-Computer Systems Interaction, vol. 60. Springer, Berlin - Heidelberg, pp. 135–146.

Wilson, R., Salgian, A., 2008. Gesture recognition for a webcam-controlled first person shooter. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y., Rhyne, T.-M., Monroe, L. (Eds.), Advances in Visual Computing, 5359. Springer, Berlin, Heidelberg, pp. 889–896.

Xu, Z., Zhu, M., 2006. Color-based skin detection: survey and evaluation. In: Proceedings of the 12th International Multi-Media Modelling Conference, Beijing, China, pp. 143–152.

Yang, H.-D., Sclaroff, S., Lee, S.-W., 2009. Sign language spotting with a threshold model based on conditional random fields. IEEE Trans. Pattern Anal. Mach. Intell. 31 (7), 1264–1277.

Yang, M.-H., Ahuja, N., 1998. Detecting human faces in color images. In: Proceedings of the 1998 International Conference on Image Processing ICIP 98, Chicago, Illinois, USA, vol. 1, pp. 127–130.

Yoo, T.W., Oh, I.S., 1999. A fast algorithm for tracking human faces based on chromatic histograms,. Pattern Recog. Lett. 20 (10), 967–978.

Zabulis, X., Baltzakis, H., Argyros, A., 2009. Vision-based hand gesture recognition for human-computer interaction, The Universal Access Handbook. LEA.

Zhao, J., Chen, T., 2009. An approach to dynamic gesture recognition for real-time interaction. In: Wang, H., Shen, Y., Huang, T., Zeng, Z. (Eds.), The Sixth International Symposium on Neural Networks (ISNN 2009), vol. 56. Springer, Berlin - Heidelberg, pp. 369–377.

Zhao, S., Tan, W., Wen, S., Liu, Y., 2008. An improved algorithm of hand gesture recognition under intricate background. In: Xiong, C., Huang, Y., Xiong, Y., Liu, H. (Eds.), Intelligent Robotics and Applications, 5314. Springer, Berlin, Heidelberg, pp. 786–794.