Robust Webcam-Based Hand Detection for Initialisation of Hand-Gesture Communication

Tilo Strutz(✉), Alexander Leipnitz, and Björn Senkel

Institute of Communications,
Leipzig University of Telecommunications (HfTL), Leipzig, Germany
{strutz,alexander.leipnitz,bjoern.senkel}@hft-leipzig.de

Abstract. The recognition of hand gestures is still a challenging task in real-life scenarios, especially when the hardware is restricted to a cheap optical camera. The first step in such systems is to find at least one hand that can be tracked in order to identify postures or gestures. We propose a robust and real-time method that is able to reliably detect the hand in various environments to initialize hand-gesture communication. It is based on an innovative combination of different sources of information (colour, motion, trajectory) and a dynamic hand-wave gesture commencing hand tracking and hand gesture recognition.

Keywords: Hand detection · Human-machine interaction
Initialisation of communication · Gesture recognition

1 Introduction

The research on human-machine interaction based on visual gestures can look on a history, which longs back to the mid-90th. The progress of research and development led even to a dedicated book [1]. Nevertheless, until today, it remains difficult to identify the hand and to decide about its actions when the illumination and the background can be arbitrary or they are even changing.

Hand detection got a push with the availability of affordable depth cameras. Depth information benefits the detection of object contours and extends the scene analysis to the third dimension. As depth cameras output infra-red light, they also can measure the distances of objects in dark environments. This advantage turns into a drawback in scenarios where the objects are exposed to bright sunlight, which contains a significant amount of infra-red radiation disturbing the camera light [2]. Another problem arises, when there are other objects (head or body) having the same distance as the hand and it cannot immediately be decided which pixels belong to the hand.

The intention of the proposed approach is to provide a reliable low-cost and real-time technology that quickly determines the position of an operating hand in an arbitrary scene. This detection is the prerequisite, for example, for the

A. Ronzhin et al. (Eds.): ICR 2018, LNAI 11097, pp. 259–269, 2018.
https://doi.org/10.1007/978-3-319-99582-3_27

subsequent tracking and posture or gesture classification. The performance is demonstrated under various conditions.

As soon as the position of the hand centroid can be reliably determined in each frame of an image sequence, the tracking of the hand and further processing is possible. Information from the time-line can even benefit the detection process, since it can be assumed that the hand does not arbitrarily jump from one position to another, for example.

2 Related Work and Proposal

Hand detection can be described as a segmentation problem where different features are needed for the discrimination between the object of interest and the remaining content of the image (i.e. background and other objects). These features are mainly derived from colour, texture, motion, and/or depth. While textural information is hardly used for foreground-background classification, it is typically required for the derivation of motion information. There are also attempts to find hands based on texture, as for instance in [3].

This paper focusses on hand detection based on colour and motion information. Depth information is not considered yet, because the low-cost restriction limits the required hardware to a simple web camera. The aim is to make full use of the information provided by the optical camera. However, the proposed approach does not exclude the possible integration of depth information in future set-ups.

There have been numerous attempts in the past to solve the problem of detection and tracking. The majority of them is limited to fixed conditions. A typical (and wrong) assumption is that there is something like "skin colour" and this is enough to distinguish between hand and background. This colour is mostly defined either by one or more regions in the three-dimensional colour space, or each RGB-triple is assigned a probability of belonging to skin. Videos from real-life show, however, that there can be many objects in the background also having skin-like colour, including the body (e.g. forearm or the face) of the operator. In addition, the illumination may heavily influence the appearance. Shifts in the RGB values can be observed as well as shaded areas or bright reflections in the hand region depending on the position of the light source. In [4] this had been already taken into account and a more advanced technique was proposed combining colour and motion information, while the uncertainty in finding the correct hand position was compensated using a Hidden-Markov model.

The problem of hand detection has also been addressed in a different context than tracking and gesture recognition. [5] discusses an approach that tries to find all hands in still images. Based on three methods working in parallel and utilising oriented gradients, skin colour, and face and arm detection, the hand positions and their rotated bounding boxes are determined. While showing excellent detection performance, this approaches is far too complex for real-time application.

A significant step towards reliable hand tracking including the provision of the hand shape has been presented by Stergiopoulou and his co-workers [6]. They also combine colour and motion information, while the initial colour model adapts to the actual image content. Besides of this, a background model containing the skin-like regions has been used to improve the detection. [6] also serves as a good review of earlier proposals, which cannot be discussed here again. Recent successes in image classification with convolutional neural networks (CNN) have also inspired investigations into object detection. In [7], CNNs are used to find hands in still images with high reliability, while in [8] the detection approach is part of a hand segmentation and activity recognition process. However, the high computing effort does not allow real-time applications with low-cost hardware.

This paper proposes a novel method that initially identifies the operating hand based on a probabilistic and innovative combination of colour information, motion information, and trajectory. The influence of skin-coloured background is significantly reduced by applying a background model that is updated frame by frame. Even moving objects such as other hands or faces are effectively suppressed. The overall complexity remains low so that real-time applications are possible.

3 Method

The underlying idea of the proposed detection approach is to use a hand-wave gesture. This has at least three advantages. Firstly, a wave gesture is intuitive, like "hello, here I am". It is similar to the initial voice command used in voice-controlled devices, like Amazon Echo or Apple's Siri. Secondly, this gesture does not require hand contours or other precise information and, therefore, it can be recognised based on relative simple techniques. Thirdly, the detection is robust as it combines colour, motion, and constraints on the hand-position sequence, while minimizing the chance of false detections of other skin coloured objects.

In our system, the captured images are resized down to 320×240 pixels ensuring a real-time processing on state-of-the-art computers. All empiric thresholds are related to this size, if not differently stated. The next subsections describe how the necessary information is derived and combined.

3.1 Colour Information

Based on the ideas of [10] a skin-colour model is established that defines for each RGB triplet a probability of belonging to a skin region. These probabilities have been learned once in an offline training phase based on annotated pictures (HGR and ECU data banks [10,11]). The probability of colour c belonging to a skin region is determined by

$$P_{\text{predef}}(skin|c) = \frac{n(c, X_{skin})}{n(c, X)}, \tag{1}$$

with $n(c, X_{skin})$ being the frequency that this colour occurred in a skin region and $n(c, X)$ being the total number of observations of this colour.

As already discussed in the introduction, such a predefined (offline trained) model might fail when the actual light conditions produce divergent colours. To overcome this problem, we implement a second colour probability model $P_{\text{adapted}}(skin|c)$ that is initialised by the predefined probabilities. If the hand-blob position could be determined with a sufficient reliability, the second colour probability model is updated frame-by-frame by increasing the number of observations for each colour according to Eq. (1). Strictly speaking, this adapted model does not represent general skin-colour probabilities but hand-colour probabilities in the actual scenario.

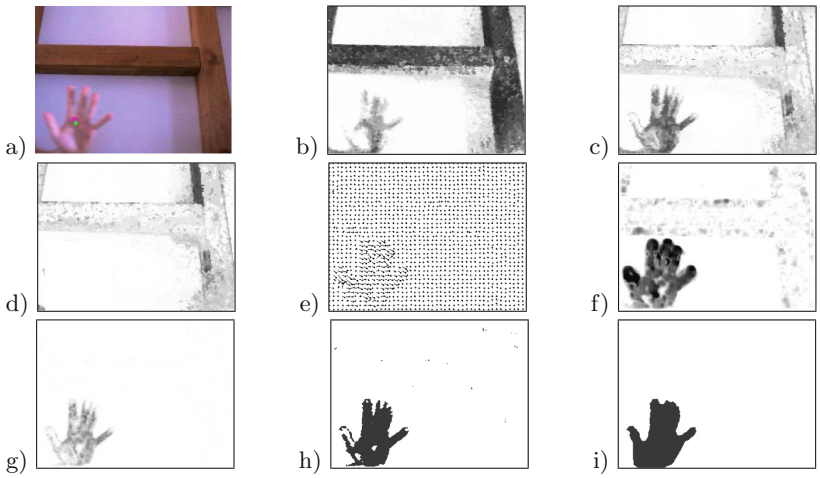

Fig. 1. Combination of skin and movement information: (a) original image, (b) skin-probability map (predefined); (c) skin-probability map (adapted); (d) skin-coloured background; (e) motion-vector field; (f) movement-probability map; (g) multiplicative combination of (c) and (f); (h) binarised version of (g); (i) after morphological processing

Figure 1 shows an example of an image and results of different processing steps. The pictures (b) and (c) visualise the difference between the predefined and the adapted model after 38 frames of a test sequence. In the adapted colour model, the wooden bars have a much lower probability of being part of a skin-coloured object and hand pixels now show a higher probability. However, as the update process cannot utilise the correct hand contour, also background pixels in regions that have been temporarily occluded by the moving hand could be assigned higher probabilities.

3.2 Motion Detection

Motion is the second source of information in our set-up. Based on the method of Farnebäck [12], a dense motion-vector field is generated. A vector $\mathbf{v} = (d_x, d_y)$ represents the horizontal and vertical movement of each pixel. The magnitudes of the motion vectors are converted into a movement-intensity map (Fig. 1e and f). It can be seen that, due to the camera noise, some movement is also detected for non-moving image content.

The normalisation of the vector magnitudes

$$|\mathbf{v}| = \sqrt{d_x^2 + d_y^2}, \tag{2}$$

to their maximum value transfers the intensity map into a probability map

$$P(motion|\mathbf{v}) = \frac{|\mathbf{v}|}{\max\left[10, \max\limits_i (|\mathbf{v}_i|)\right]}, \tag{3}$$

It must be considered, however, that the motion-detection process generates vectors that are affected by noise and the maximum motion vector can be very small if there are no moving objects. To avoid these problems, the normalisation value is limited to a minimum vector length of 10 pixels.

3.3 Combination of Skin and Movement Information

The hand-detection is based on the assumption that there must be a skin-coloured object that is moving. Consequently, the probability, whether a pixel belongs to the waving hand, is the product of both single probabilities

$$P(hand) = P(motion|\mathbf{v}) \cdot P_{\text{adapted}}(skin|c). \tag{4}$$

This is an effective method to eliminate non-moving objects in the background (Fig. 1g).

3.4 Determination of Hand Position

After computation of the hand probabilities the resulting map is binarised yielding the hand object and some spurious blobs. In contrast to many other approaches, we do not apply Otsu's method but determine the binarisation threshold in such a manner that the sum of the entropies of the resulting sub-sets is maximised [13], as it showed a better trade-off between missing hand parts and false detection of background pixels in our set-up. Figure 1(h) shows an example of a binarised probability map.

When the image contrast is rather low, it can happen that sufficient large motion vectors are determined only at the boundary of the moving hand, while no motion can be recognised for the inner part of the hand and the wrist. The binarisation can lead to scattered binary objects (blobs, Fig. 2c) necessitating

special morphological post-processing. At first, possibly scattered small blobs have to be merged via dilatation using a circular structural element (Fig. 2d). The diameter d_s of the structural element is dependent on the size of the largest blob, roughly according to $d_s = 25 \cdot \exp(-\text{blobSize}/9000)$. Afterwards, the largest blob is newly determined and connected with surrounding blobs if (i) they have a contour point that is closer to the largest blob than the half of their own contour length and (ii) they do not have a contour point that is farther away than the doubled width or height of largest blob's bounding box. The second condition avoids connections with lengthy blobs which probably belong to structures in the background.

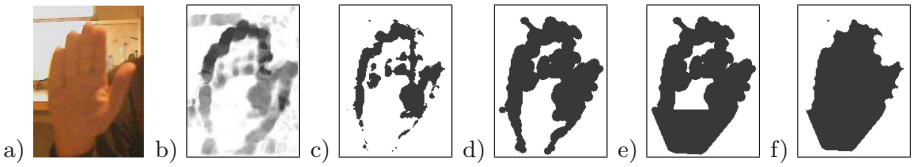

Fig. 2. (a) low contrast within hand region; (b) probability map $P(motion)$; (c) binarised $P(hand)$; (d) after dilatation and connection of blobs; (e) filled convex hull in the lower third of the blob; (f) blob after holes are filled and erosion

This results in a binarised hand probability covering only some boundary parts of the hand region (Fig. 2c), which cannot be closed by simple dilatation. After connecting the largest blob with small nearby blobs (Fig. 2d) two little blobs are spuriously connected to the main blob, however, the bottom region could correctly be extended. Nevertheless, the hand structure remains open in the wrist region. Drawing the convex hull around the entire blob would close the open parts and would surely comprise all pixels of the hand. If, however, the fingers are spread, too much background pixels would be integrated in the blob spoiling the skin-colour-adaptation process. The dilemma can be resolved by drawing the convex hull only around the lower third of the blob (Fig. 2e). This closes the blob structure in the hand-palm region while keeping the separated fingers.[1]

Finally, all holes in the resulting blob structure are filled and the blob is eroded (Fig. 2f) using the same structural element as the dilatation described above. This also removes the spurious spot, which can be seen in the top-left of Fig. 2d). This well-designed sequence of morphological operators yields a binary image in which the largest blob represents the region of the waving hand and can be used to determine the current hand position (centroid of the blob).

[1] During the hand-wave gesture, it can be assumed that the hand is presented with fingers pointing upwards.

3.5 Removing the Forearm

The detected moving foreground object does not only contain the hand but also the forearm if the latter is uncovered. This affects the hand-position determination and the adaptation of the skin probabilities. Hence, the forearm has to be removed. This problem had been addressed already in [9,14,15]. Typically the palm region is first identified based on a distance transform of the blob, then the wrist position is located. We follow a similar approach including some new steps that increase the reliability in difficult scenarios. The entire procedure is explained in the following with reference to Fig. 3.

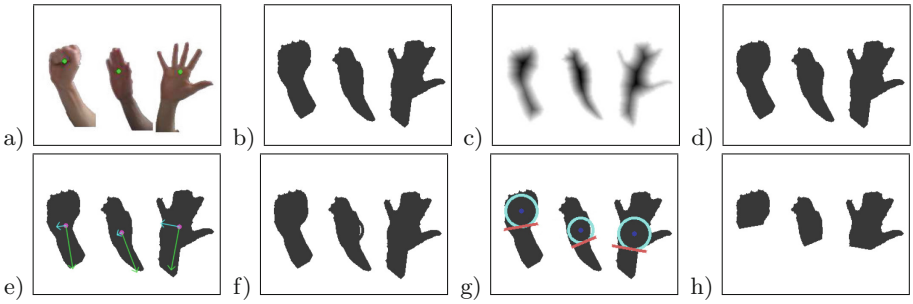

Fig. 3. Removing the forearm: (a) hands from original image; (b) hand blobs; (c) distance transformation; (d) points of largest distance and maximum inscribing circle; (e) principal component analysis; (f) shifted inscribing circles; (g) dividing lines between hand and forearm; (h) final hand blobs (Color figure online)

Palm and Orientation of the Hand. A distance transformation (Fig. 3c) of the hand blob yields a point $\mathbf{c} = (c_x, c_y)$ having a distance to the blob boundary that is equal to the radius r of the largest inscribing circle. So, ideally $\mathbf{c}$ is representing the palm (Fig. 3d). Based on a principal-component analysis, the orientation of the blob is determined (Fig. 3e, green line: main axis; blue line: orthogonal axis).

Modifying the Position of the Inscribing Circle. There are cases where the inscribing circle is not at the correct position. Figure 3(d) shows three examples where (i) the circle is almost at the top of the blob (blob of a fist), (ii) the circle is in the middle of the hand (with closed fingers), and (iii) the circle is at the wanted position (hand with spread fingers). Obviously, in some cases the position of the circle has to be corrected by shifting it towards the wrist. This is done by taking advantage of the distance d between the circle centre $\mathbf{c}$ and the top of the hand (with respect to the orientation of the hand).

In order to find the offset $\Delta\mathbf{c}$ by which the centre point of the inscribing circle has to be moved along the main hand axis, three fix points can be identified. First of all, if the palm is correctly located ($d = 3 \cdot r$), no movement is necessary.

Secondly, if the distance $d = r$ (circle touches the top of the hand), it is assumed that the hand is actually a fist and probably the circle is located right. The maximum $\Delta\mathbf{c} = r$ is required when the circle is in the middle of the hand blob ($d = 2 \cdot r$). Between these points, a linear function is assumed to be helpful.

When the distance d is greater than $3 \cdot r$, then no correction is performed because it is most likely that r simply has been underestimated. This can happen if the blob contains only part of the entire hand.

The result of the modification can be seen in Fig. 3(f). The shifted circle represents the palm region much better than the original one.

Calculate Tangent of the Inscribing Circle. In order to correctly separate hand and forearm, the tangent of the inscribing circle that is orthogonal to the main principal axis is required, Fig. 3(g). All blob pixels below this tangent are erased yielding the final hand blob, Fig. 3(h). The entire procedure enables a fast and largely accurate forearm detection and removal with respect to the orientation. It is rotationally invariant on the condition that the hand is always in an upright position and not tilted more than 90° to the left or the right. This method offers an excellent compromise between forearm-removal ability and computational time compared to the approaches in the cited literature.

3.6 Evaluation of the Wave Gesture

If the hand-blob position can be determined for a sequence of images, it must be checked, whether the direction of movement has changed. As we assume that the hand wave is performed horizontally, only the horizontal components of the corresponding motion vectors have to be evaluated.

The entire process uses a simple state machine comprising the states: "no object found", "skin-coloured object is moving", "movement has changed its direction once", and "movement has changed its direction twice". If the last state is reached and the distance (d(wave) in Fig. 4) between the two reversing points is larger than a half of the blob's bounding box, then the hand has reliably been detected and can be tracked. The state machine is accompanied by a consistency check. The state machine is reset, when the current centroid position is not within a region that can be predicted based on the previous position and the motion vector field. Changes of motion directions are only taken into consideration if the

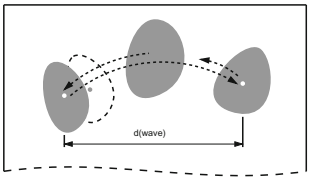

Fig. 4. Determination of direction changes

average horizontal movement exceeds a distance of three pixels.[2] In addition, the motion direction must have been consistent for the last 200 milliseconds assuming that the whole motion in one direction typically lasts about a half second.

At the reversing points, the movement is typically close to zero and the hand blob cannot be determined correctly because the probability of being part of a hand decreases for all pixels, see Eq. (4). A missing blob is tolerable for a period of about 200 milliseconds. This duration has been determined empirically. If the blob cannot be found for a longer time, the state machine is reset to its initial state.

3.7 Generation of a Skin-Colour Background Image

After the first direction change has been detected, the hand-blob position is known with certain reliability. From now on, the adapted skin-probability map is copied for each frame into a skin-colour background image excluding the region of the identified hand blob.

This background information can improve the combination of colour and motion as described in Subsect. 3.3. Equation (4) is modified to the heuristic formula

$$P(hand) = P(motion|\mathbf{v}) \cdot \max\left[0, P_{\text{adapted}}(skin|c) - b\right],$$

while $b = P_{\text{background}}(skin|c)$ is the skin-probability value of the corresponding background pixel. This technique effectively avoids the leakage of the hand blob into skin-coloured background regions and suppresses slightly moving objects like faces.

4 Results

Figure 5 shows the results directly after the second direction change of the putative hand blob has been detected. From left to right, the images contain: the original frame with a green dot indication the centroid of the detected blob, the probability map of the predefined skin-colour model, the probability map of the adapted skin-colour model, the skin-coloured background, the hand-probability image, and the final blob.

As can be seen, the original images are very diverse with respect to background colours and texture, the contrast, and the lighting conditions. The adapted skin-colour model reflects the colour of the moving hand at least as well as the offline trained model and mostly much better. In Fig. 5(b) the inner parts of the hand palm have the same colours as the wall in the background, which makes the distinction very difficult. Figure 5(c) shows challenging conditions not only with respect to the colours in the background, but also the shirt seems to be much more skin-like than the hand according to the predefined skin probability map. During the short wave gesture, the actual hand pixels are assigned higher

[2] Keep in mind that the vector magnitudes are often close to zero for inner hand parts.

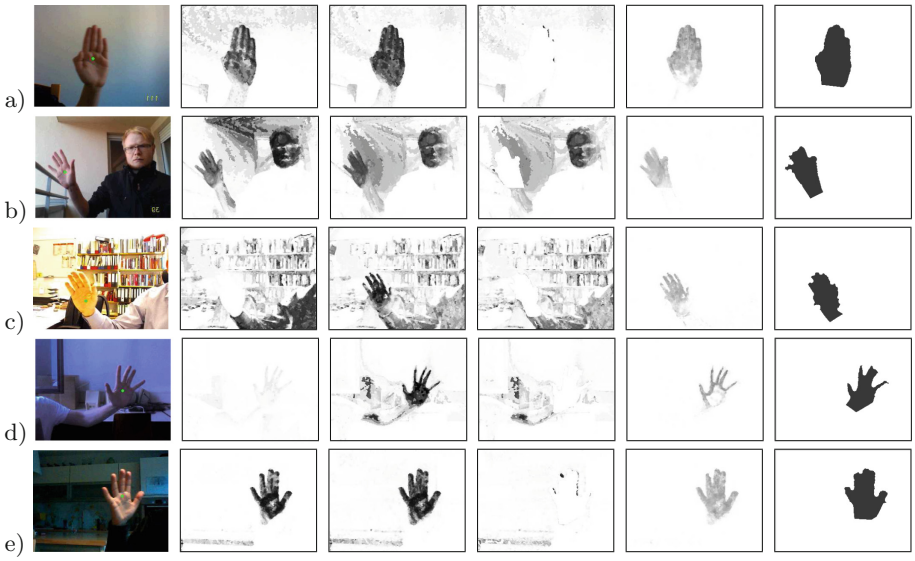

Fig. 5. Results after wave gesture has been completed for different sequences. The four pictures in the middle show the probability maps $P_{\text{predef}}(skin|c)$, $P_{\text{adapted}}(skin|c)$, $P_{\text{background}}(skin|c)$, and $P(hand)$; the darker the pixel is, the higher is the probability. See text for more details. (Color figure online)

probabilities in the adapted model. Nevertheless, the binarisation keeps parts of the shirt leading to a blob that is too large. The hand detection is still successful as hand and forearm could be separated using the dedicated processing step. Figures 5(d) and (e) underline that the approach is able to keep finger information when the conditions are sufficiently good. It has to be mentioned that the method also works well when the camera is slightly moving.

5 Summary

We have presented a very robust method for the initial detection of a skin-coloured moving object (the hand). The probabilistic combination of colour and motion information, the inventive sequence of morphological processing steps together with a time-line observation of the position reliably finds the waving hand. The adapted skin-colour model is another key feature as the predefined skin-colour model tends to fail under realistic light conditions. The generated skin-coloured background image suppresses the influence of other skin-coloured objects, especially slightly moving faces or persons in the background. Forearms can be removed with sufficient accuracy.

The entity of the different processing steps is not required in each scenario and some steps might have no effect sometimes. However, the conditions in real-life applications vary a lot making it necessary to have a cure for each case

at hand. Supporting reproducible research, all image sequences and the hand detection software can be downloaded from [16].

References

1. Premaratne, P.: Human Computer Interaction Using Hand Gestures. Springer, Singapore (2014). https://doi.org/10.1007/978-981-4585-69-9
2. Langmann, B., Hartmann, K., Loffeld, O.: Depth camera technology comparison and performance evaluation. In: Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods, pp. 438–444 (2012)
3. Triesch, J., von der Malsburg, C.: Robust classification of hand postures against complex background. In: Proceedings of 2nd International Conference on Automatic Face and Gesture Recognition, pp. 14–16, October 1996
4. Chen, F.-S., Fu, C.-M., Huang, C.-L.: Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vis. Comput. **21**(8), 745–758 (2003)
5. Mittal, A., Zisserman, A., Torr, P.H.: Hand detection using multiple proposals. In: BMVC, pp. 1–11 (2011)
6. Stergiopoulou, E., Sgouropoulos, K., Nikolaou, N., Papamarkos, N., Mitianoudis, N.: Real time hand detection in a complex background. Engin. Appl. Artif. Intell. **35**, 54–70 (2014)
7. Deng, X., Zhang, Y., Yang, S., Tan, P., Chang, L., Yuan, Y., Wang, H.: Joint hand detection and rotation estimation using CNN. IEEE Trans. Image Process. **27**(4), 1888–1900 (2018)
8. Bambach, S., Lee, S., Crandall, D.J., Yu, C.: Lending a hand: detecting hands and recognizing activities in complex egocentric interactions. In: IEEE International Conference on Computer Vision (ICCV), pp. 1949–1957, December 2015
9. Palacios, J.M., Sagüs, C., Montijano, E., Llorente, S.: Human-computer interaction based on hand gestures using RGB-D sensors. Sensors **13**(9), 11842–11860 (2013)
10. Kawulok, M., Nalepa, J., Kawulok, J.: Skin detection and segmentation in color images. In: Celebi, M.E., Smolka, B. (eds.) Advances in Low-Level Color Image Processing. LNCVB, vol. 11, pp. 329–366. Springer, Dordrecht (2014). https://doi.org/10.1007/978-94-007-7584-8_11
11. Phung, S.L., Bouzerdoum, A., Chai, D.: Skin segmentation using color pixel classification: analysis and comparison. IEEE Trans. Pattern Anal. Mach. Intell. **27**(1), 148–154 (2005)
12. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Bigun, J., Gustavsson, T. (eds.) SCIA 2003. LNCS, vol. 2749, pp. 363–370. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-45103-X_50
13. Kapur, J.N., Sahoo, P.K., Wong, A.K.: A new method for gray-level picture thresholding using the entropy of the histogram. Comput. Vis. Grap. Image Process. **29**(3), 273–285 (1985)
14. Wang, B., Xu, J.: Accurate and fast hand-forearm segmentation algorithm based on silhouette. In: 2012 IEEE 2nd International Conference on Cloud Computing and Intelligent Systems (CCIS), vol. 2. IEEE (2012)
15. Chai, X., Fang, Y., Wang, K.: Robust hand gesture analysis and application in gallery browsing. In: IEEE International Conference on Multimedia and Expo, ICME 2009, pp. 938–941. IEEE (2009)
16. http://www1.hft-leipzig.de/strutz/Papers/RoHaDe-resources/ . Accessed 13 June 2018