BALKAN JOURNAL OF ELECTRICAL & COMPUTER ENGINEERING, Vol. 9, No. 3, July 2021

Real-Time Hand Motion Recognition: A Robust Low-Cost Approach

Anıl Bas and Hasan Erdinç Koçer

Abstract— **This study presents a robust, low-cost hand motion recognition approach designed to run on low-end computer systems. Our method detects and tracks hand region using real-time images obtained from a low-resolution camera (i.e. webcam) and is not depended on any training or calibration and is not required any special camera apparatus or selectors. The proposed system involves several image processing techniques such as background subtraction, face detection, skin colour detection and template matching. The experimental results show promising performance under various conditions. The method has a wide range of applications where more natural ways of interaction required, such as virtual reality applications, assistive technologies and simulation.**

Index Terms— **Background subtraction, Face detection, Hand movement recognition, Motion tracking, Template matching**

I. Introduction

AS A RESULT of the rapid development of information technology, computers have become an indispensable part of human lives. Although we see both hardware and software advancements on many levels, it has not changed or affected how we use conventional input devices such as mouse and keyboard. Unfortunately, the control ability of the user is very limited with these devices in virtual reality applications like simulation games, robotic surgery and military training which often require more natural ways of interaction.

In recent years, using the human body as an "input device" has become more popular from education [1] to entertainment [2] to health care [3-4]. Moreover, rather than relying on physical control devices, end-users prefer to communicate with computers using interactive input interfaces (e.g. touch screen and voice command) [5]. Within this context, movement recognition is emerging as an important research area in human-computer interaction and virtual reality.

The aim of this study is to develop a robust, accurate and computationally efficient hand motion recognition system without using any special camera apparatus and particular selectors such as bracelets, gloves or finger tapes. Furthermore, the study can be used as a base library and renew outdated applications by adding interactivity at no cost.

In this paper, we present a holistic approach to hand movement detection. First, the moving regions are determined on the real-time video frames by using background subtraction. Second, the area of interest is filtered from the unrelated movement parts such as face, hair and shoulders. Third, the hand template is searched in the filtered area. Fourth, the skin colour matching is performed on the best-resulted section from the previous phase. Finally, hand movements are categorised into standby, vertical, horizontal and cross directions using timed hand position data.

The paper is organised as follows: Section 2 provides an overview of the related work on hand detection and motion recognition. Section 3 presents the proposed algorithm by describing used image processing methods. Experimental results of our approach are shown in Section 4. Section 5 discusses the future work for improving the study.

II. Related Work

Throughout the years, a considerable amount of diverse techniques has been proposed in hand detection and motion recognition fields. In this section, we reviewed this body of literature as broadly as possible by focusing mainly (but not limited to) cost and performance aspects.

In Aran's video-based sign language recognition study [6], hand gestures and shapes were captured with the help of coloured gloves. Yin [7] performed hand gesture recognition with a camera directed to partly coloured gloves and obtained impressive results with regards to the control of map operations. With the help of infrared markers, Yang [8] operated a forklift truck effectively. Ikizler [9] focused on motion understanding from whole-body images. Results were separated into four categories: walking, running, throwing and catching movement. Al-Rajab [10] successfully applied the motion recognition process to a media player controller.

In order to obtain high success rates, additional video equipment (sensor and lighting apparatus) or distinctive pointers (gloves and finger tapes) are commonly used in the literature. Studies based on this approach have achieved very high accuracy rates, around 97% [11-14].

Another approach for motion recognition is to use only one hand with a constant background. Wang and Qin [15] presented

ANIL BAS is with Department of Computer Engineering, Marmara University, Istanbul, Turkey, (e-mail: anil.bas@marmara.edu.tr).

https://orcid.org/0000-0002-3833-6023

HASAN ERDİNÇ KOÇER, is with Department of Electrical and Electronics Engineering, Selcuk University, Konya, Turkey, (e-mail: ekocer@selcuk.edu.tr).

https://orcid.org/0000-0002-0799-2140

Manuscript received December 25, 2020; accepted May 4, 2021.
DOI: 10.17694/bajece.845276

a hand tracking and gesture recognition framework that allows users to control their fingers as a virtual mouse with six-degree-freedom.

Dardas and Georganas [16] and Dardas and Petriu [17] focused on solving the real-time hand gesture recognition problem by classifying hand poses with pattern learning models such as Support Vector Machine (SVM) and Principal Component Analysis (PCA).

Hsieh et al. [18] explored the recognition of four-way hand movement and achieved 93.13% classification rate on 750 images. Trigueiros et al. [19] compared four different training algorithms to measure the detection accuracy of static hand gestures. Testing K-NN, Naive Bayes, ANN and SVM algorithms on two datasets, they obtained success rates of 95.45%, 25.87%, 96.99%, 91.66% and 88.52%, 66.50%, 85.18%, 80.02% respectively.

As part of their study, Sangineto and Cupelli [20] presented a model-based approach that applies curve and graph matching techniques to finger models. They achieved 90% hand detection rate on 1645 images. Similar to our approach, Toni and Darko [21] reached a 78% success rate using skin colour classification and background subtraction.

Recent studies in a similar vein include Jacobs et al. [22] and Molchanov et al. [23]. Both incorporate an additional focus on deep learning which is used as a classifier for dynamic hand gesture recognition.

A. *Contributions*

Hand segmentation and hand motion recognition have been studied for many years in both computer vision and human computer interaction societies. Numerous works have been presented in the past two decades. These also include existing commercial solutions like Microsoft Kinect [24] and Leap Motion [25].

Our primary aim in this paper is not to compete directly with these methods. Moreover, the quantitative comparison with state-of-the-art methods would not be reasonable given that our approach rigidly focuses on low-cost and real-time functioning. For this reason, we only provide an evaluation that examines the performance of our work.

The main contribution of study is as follows: (1) Our approach is designed to run on low-end computer systems under the assumption of no learning or training process and using limited sources. (2) The algorithm identifies the hand region on real-time images obtained from a low-resolution camera (i.e. webcam), without using any special camera apparatus and selectors. (3) Proposed work could be used as a base library and renew outdated applications at no cost. (4) To our knowledge, such software is not publicly available (free) for educational or academic purposes. We would like to fill this gap by releasing the source code of our algorithm.

III. Methods

Our approach to motion detection is primarily based on examining the correlation between current and previous frames. The major problem for similar systems is that the process should be accurate and efficient enough to perform task operations accordingly. Therefore, delays and failures are two critical concerns for the correct functioning, especially during complex procedures such as template matching or frame segmentation. We begin by highlighting the novelty of the research and giving a brief description of the methods used in the study.

A. *Background Subtraction*

The background subtraction is a cleaning process by separating the moving parts from the constant ones in an image. We apply this process right after the conversion from RGB to grayscale. The grayscale representation simplifies the algorithm and reduces computational requirements as grayscale transformations are often used for extracting descriptors instead of operating directly on colour images. The weighted sum of the R (red), G (green) and B (blue) components is used in the conversion of values in RGB colour space to grayscale form [26]. We follow the equation:

$$P_{grey} = 0.2989 \times P_{red} + 0.5870 \times P_{green} + 0.1140 \times P_{blue} \qquad (1)$$

where P_{grey} is a grey level value of pixel, P_{red}, P_{green} and P_{blue} represent RGB pixel values, respectively.

After the grayscale transformation, we start the background subtraction process. The motion detection is carried out by a practical comparison of two vectors which each contains pixel values of two sequential images. Because the comparison process is conducted at the pixel level, each pixel value is subtracted from the one that is at the same coordinate in the previous frame. The variance is calculated by:

$$J_t(x) = \begin{cases} 1, & if\ I_t(x) - I_{t-1}(x) > T \\ 0, & otherwise. \end{cases} \qquad (2)$$

where $J_t(x)$ is the pixel value of x position at time t and $I_{t-1}(x)$ is the previous pixel value of x position (at time $t-1$) and T is the threshold value [27].

B. *Face Detection*

Avoiding unnecessary regions is a feasible and common technique in image processing to reduce the processing time and simplify refinement operations. Therefore, we excluded the face area from the control area. We used the face detection algorithm of Viola and Jones, which is based on Haar Cascade classifiers [28]. Figure 1 shows sample Haar-like features applied on a face image. Note that any type of face detection method can be used at this stage. The use of the technique has been exemplified in these studies [29],[30].

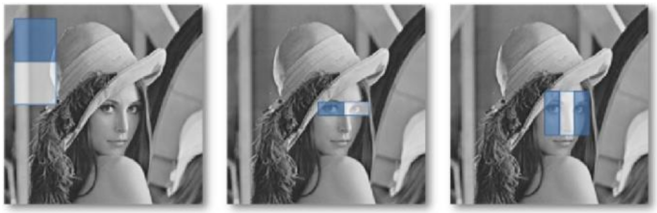

Fig. 1. The working principle of Haar-like features. Rapid face detection can be achieved by calculating features (the difference of the sum of pixels of white and blue rectangles)

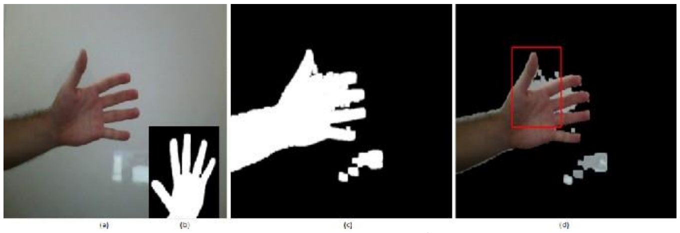

Fig. 2. The template matching progress. Main frame (a), hand pattern (b), motion area (c), template detection (d)

As a fundamental part of our detection mechanism, the majority of the motionless regions are blacked out in the background subtraction process. The remaining movement areas mainly appear around the face region. Another advantage of this is that it allows us to avoid involuntary body movements (e.g. head, face and shoulders). An example blockage case is shown in Figure 5. This increases the speed of the recognition process and provides a better initialisation.

C. *Template Matching*

Template matching is a similarity measurement process between the referenced template and each potential sub window of an image. The position of the sub window with the highest similarity indicates the location of the template in an image [31]. The matching progress and the scanned pattern are shown in Figure 2.

We used the normalised correlation coefficient method as a template matching operation. The process and the coefficient calculation are shown in Figure 3 and in Equation 3:

$$p[u,v] = \frac{\sum_{u=-n}^{n}\sum_{v=-n}^{n}\left(f_1[X_1+u,Y_1+v]-\bar{f}_1\right).\left(f_2[X_2+u,Y_2+v]-\bar{f}_2\right)}{\sqrt{\sum_{u=-n}^{n}\sum_{v=-n}^{n}\left(f_1[X_1+u,Y_1+v]-\bar{f}_1\right)^2 . \sum_{u=-n}^{n}\sum_{v=-n}^{n}\left(f_2[X_2+u,Y_2+v]-\bar{f}_2\right)^2}} \quad (3)$$

where X_1, Y_1, X_2, Y_2 are the image coordinates of the search windows, u and y are the coordinates of relation windows and f_1, f_2 are the gray values of windows (left and right images). We assume the correlation coefficient is between [-1,1] range. If the calculated $p = -1$ then there is no relationship between two windows. If $p = 1$ then there is an exact match.

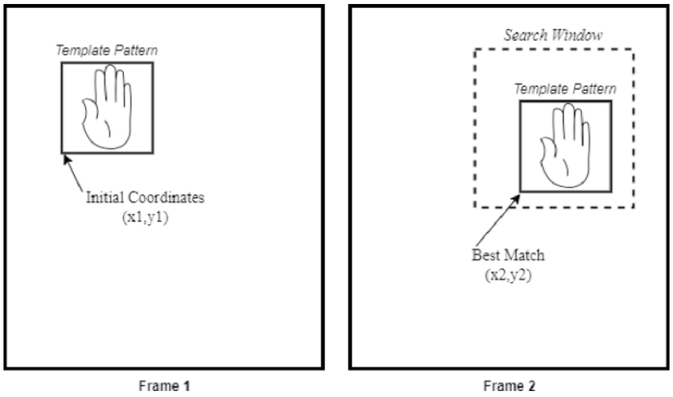

Fig. 3. Normalised cross correlation computation. The template is compared with possible candidates in the search window within the frame to detect the highest similarity

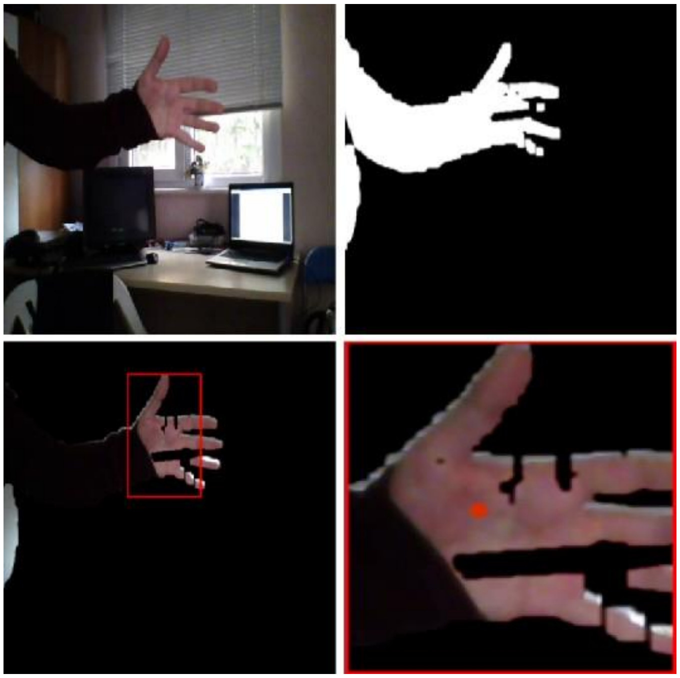

Fig. 4. Skin colour detection on a cluttered background

D. *Skin Colour Detection*

The main aim of detecting the skin colour is to decide on the movement categorisation and to make sure that the interest area contains a realistic hand, instead of hand-shaped objects in the scene. In this stage, the detected area of interest (obtained from the template matching process) is merged with the raw input image to compute skin colour. Pixels with similar colours in that area create blocks. The largest block is determined by the number of connected pixels and the centre of the largest part is marked as the motion point. The algorithm makes the prediction based on the position of the marks. Figure 4 shows the marking and the handshape area on a complex scene.

This step is necessary to compute the movement direction between frames. However, it is not directly aiding hand detection and used for the motion recognition task only.

IV. Results

The proposed system was tested with several conditions on different backgrounds. The experimental study contains eight categories of hand movements (up, down, left, right, up-left, up-right, down-left and down-right) and the click action (stop at a certain time, at a certain point). Two main status (hand only or entire body) and arm conditions (bare or clothed) were examined considering the effectiveness of the performance. The experiments repeated on four various backgrounds (white, red, green and complex). Figure 5 shows a mixture of settings examined in our analysis. Ten individual trials were made for each movement, condition and background. In total, 1440 movements were collected for the measurement.

The study was conducted on a computer that has a 3rd generation Intel i7 processor, 8 GB RAM, a Nvidia GT650M graphics card and a 1.3 MP webcam. We used EmguCV library which is a C# wrapper for the well-known OpenCV video-processing library; both of which are open-source and freely licensed [32]. EmguCV is preferred because of the accessibility and abundance of source code, its easy-to-use multi-platform structure and, most importantly, memory management (automatic garbage collection). In our experiments, our method approximately runs at 20 fps rate, which is suitable for real-time applications.

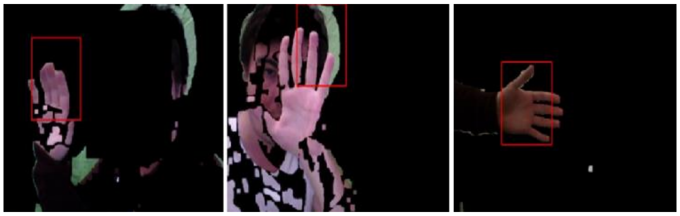

Fig. 5. Various experimental settings: entire body-clothed arm (left), entire body-bare arm (middle) and hand only-clothed arm (right)

Table 1 presents the movement recognition rates of the proposed method. Total of 360 experiments collected on 4 different static backgrounds and with 2 conditions. We identify four significant factors that emerged from this table.

First, a comparison between hand only and entire body results shows that using hand only in the field-of-view gives better results in the recognition process. Second, similarly, the complexity of the scene directly affects the successful detection rate. Third, we observed higher values on clothed arm than bare arm results. This outcome is reasonable because of the skin colour similarity between hand and arm.

The insufficient hit ratio of the click movement stands out as a fourth factor. There are two likely causes for this result: One is that the consistency between continuous frames could not be appropriately achieved since there is no learning/training

TABLE I
MOVEMENT RECOGNITION RATES (%)

Condition #1	Condition #2	Movement	White Bg	Red Bg	Green Bg	Complex Bg
Hand only	Bare arm	up	80	100	90	80
		down	90	80	80	60
		left	100	90	100	90
		right	80	80	90	80
		up-left	90	90	90	60
		up-right	80	90	80	70
		down-left	90	80	90	80
		down-right	80	80	80	70
		click	70	70	60	60
	Partial Success Rate		84.84	84.84	84.84	72.22
	Clothed arm	up	100	100	90	80
		down	90	90	90	80
		left	100	100	100	90
		right	90	100	90	70
		up-left	90	90	90	80
		up-right	90	90	100	80
		down-left	90	90	100	70
		down-right	100	100	90	90
		click	90	90	80	70
	Partial Success Rate		93.33	94.44	92.22	78.89
Entire body	Bare arm	up	90	80	80	80
		down	70	90	70	70
		left	100	80	70	70
		right	80	80	90	80
		up-left	90	70	80	70
		up-right	70	60	70	60
		down-left	100	70	60	70
		down-right	80	70	80	60
		click	50	60	50	50
	Partial Success Rate		81.11	73.33	72.22	67.78
	Clothed arm	up	90	100	100	90
		down	90	90	100	80
		left	90	70	90	60
		right	100	90	80	70
		up-left	80	90	80	60
		up-right	100	100	90	80
		down-left	80	90	70	70
		down-right	90	90	90	80
		click	60	70	60	50
	Partial Success Rate		86.67	87.78	84.44	71.11
Total Success Rate			**86.39**	**85**	**83.33**	**72.5**

option in our low-cost approach. The other cause is related to involuntary (reflex) hand and finger movement that fails the template matching correlation in a very short period of time. This is because the click data collection is mainly relied on freezing the hand in the same region.

Finally, the movement values achieved a better success rate when the hand is positioned above the mid-frame. The reason behind this is that, while the subject moves its hand to a lower level, the handshape slightly loses its parallel structure to the camera. Therefore, the algorithm could not cope with this challenging hand posture which is difficult to perceive with a single device and thus counted as a wrong move.

Figure 6 provides an example comparison between the success rates of bare arm and clothed arm conditions. It is clearly observed that our algorithm is performing better on a clothed arm (textile blocking the region of interest) compared to bare arm. This can be partially explained by the false detection of the arm region in some cases.

The proposed work has achieved an average of 81.81% success rate on 1440 real-time video sequences, which is highly promising considering possible complications including illumination, cluttered scene, camera distance, camera quality absence of any indicators or apparatus, Moreover, the evaluation has shown better performance than some of the studies [19],[21] mentioned in the literature review section. We would like to note again that the approach is not depended on any training or calibration, or particular selectors such as finger tapes, bracelets or gloves.

V. Discussion and Conclusion

With the increasing use of computer-simulated technology in daily life, sensors and cameras have been moving gradually towards substituting for buttons and touch screens. Smart TVs and game consoles have already started to apply such advancements and it is expected to spread to other electronic devices in the near future.

While powerful new devices equipped with high resolution cameras and sensors are being developed rapidly, low-level machines are not completely ignored. Mainstream products (e.g. remote controller, headset and wristband) are able to fill this gap during the transition stage. However, they add further complexity to the current system as well as the required computational resources that are needed for compiling the code of the gear. Even though those hardware tools produce better results than software enhancements, it creates a financial burden for the user and is inadequate in terms of user preference and application compatibility.

The two main contributions of this paper are as follows. (1) It is possible to treat our application as a base library and build projects with high level performance. (2) Existing projects can extend to an interactive version by implementing the study. For this reason, we release the source code of the publication to the community for further research on the subject and development of new techniques.

Although the study provides solid results, some features could still be improved. First, optimising the face detection algorithm could directly affect the accuracy of the whole system. Incorrect area detection may occur given confusion caused by misleading objects such as a hat or glasses. Second, the clicking movement, which has the lowest success rate, could be refined by tracking the movement over frames without increasing the complexity. Third, multiple hand detection could not be applied due to real-time computational costs. This could be achieved by applying computationally lightweight skin detection, similar to Conaire et al. [33].

References

[1] Z. Merchant, E.T. Goetz, L. Cifuentes, W. Keeney-Kennicutt, T.J. Davis, "Effectiveness of virtual reality-based instruction on students' learning outcomes in K-12 and higher education: A meta-analysis", Computers & Education, Vol. 70, 2014, pp. 29–40.

[2] D. Chambers, "'Wii play as a family': the rise in family-centred video gaming", Leisure Studies, Vol. 31, 2012, pp. 69–82.

[3] H.H. Mousavi, M. Khademi, "A review on technical and clinical impact of microsoft kinect on physical therapy and rehabilitation", Journal of Medical Engineering, 2014.

[4] N.E. Seymour, A.G. Gallagher, S.A. Roman, M.K. O'brien, V.K. Bansal, D.K. Andersen, R.M. Satava, "Virtual reality training improves operating room performance: results of a randomized, double-blinded study", Annals of Surgery, Vol. 236, No. 4, 2002, pp. 458–464.

[5] P.W. Lee, H.Y. Wang, Y.C. Tung, J.W. Lin, A. Valstar, "TranSection: hand-based interaction for playing a game within a virtual reality game", 33rd ACM Conference on Human Factors in Computing Systems, Seoul, South Korea, 18 – 23 April 2015.

[6] O. Aran, Vision based sign language recognition: modeling and recognizing isolated signs with manual and non-manual components, PhD Thesis, Bogazici University, Istanbul, Turkey, 2008.

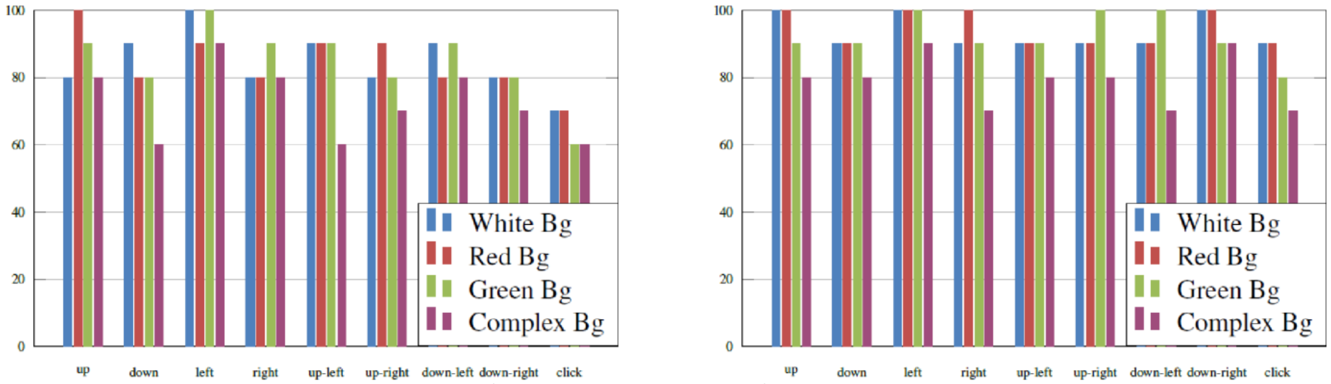

Fig. 6. Example comparison of hand only motion recognition for bare arm and clothed arm conditions

[7] Y. Yin, Toward an intelligent multimodal interface for natural interaction. MSc Thesis, Massachusetts Institute of Technology, Cambridge, MA, United States, 2010.

[8] X. Yang, A hand input-based approach to intuitive human-computer interactions in virtual reality. MPhil Thesis, The University of Hong Kong, Pokfulam, Hong Kong, 2010.

[9] N. Ikizler, Understanding human motion: recognition and retrieval of human activities, PhD Thesis, Bilkent University; Ankara, Turkey, 2008.

[10] M. Al-Rajab, Hand gesture recognition for multimedia applications, PhD Thesis, University of Leeds, Leeds, UK, 2008.

[11] C.P. Chen, Y.T. Chen, P.H. Lee, Y.P. Tsai, S. Lei, "Real-time hand tracking on depth images", Visual Communications and Image Processing, 2011, pp. 1–4.

[12] C. Keskin, L. Akarun, "STARS: Sign tracking and recognition system using input–output HMMs", Pattern Recognition Letters, Vol. 30, No. 12, 2009, pp. 1086–1095.

[13] L. Lamberti, F. Camastra, "Handy: a real-time three color glove-based gesture recognizer with learning vector quantization", Expert Systems with Applications, Vol. 39, No. 12, 2012, pp. 10489–10494.

[14] Y.H. Lee, S.K. Wu, Y.P. Liu, "Performance of remote target pointing hand movements in a 3D environment", Human Movement Science, Vol. 32, No. 3, 2013, pp. 511–526.

[15] X. Wang, K. Qin, "A Six-degree-of-freedom Virtual Mouse based on Hand Gestures", International Conference on Electrical and Control Engineering, Wuhan, China, 25–27 June 2010.

[16] N.H. Dardas, N.D. Georganas, "Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques", IEEE Transactions on Instrumentation and Measurement, Vol. 60, No. 11, 2011, pp. 3592–3607.

[17] N.H. Dardas, E.M. Petriu, "Hand gesture detection and recognition using principal component analysis", IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, Ottawa, Canada, 19–21 September 2011.

[18] C.C. Hsieh, D.H. Liou, D. Lee, "A real time hand gesture recognition system using motion history image", 2nd International Conference on Signal Processing Systems, Dalian, China, 5–7 July 2010.

[19] P. Trigueiros, F. Ribeiro, L.P. Reis, "A comparison of machine learning algorithms applied to hand gesture recognition", 7th Iberian Conference on Information Systems and Technologies, Madrid, Spain, 20–23 June 2012.

[20] E. Sangineto, M. Cupelli, "Real-time viewpoint-invariant hand localization with cluttered backgrounds", Image and Vision Computing, Vol. 30, 2012, pp. 26–37.

[21] B. Toni, J. Darko, "A robust hand detection and tracking algorithm with application to natural user interface", 35th International Convention MIPRO, Opatija, Croatia, 21–25 May 2012.

[22] K. Jacobs, M. Ghasiazgar, I. Venter, R. Dodds, "Hand Gesture Recognition of Hand Shapes in Varied Orientations using Deep Learning", Annual Conference of the South African Institute of Computer Scientists and Information Technologists, Johannesburg, South Africa, 26–28 September 2016.

[23] P. Molchanov, S. Gupta, K. Kim, J. Kautz, "Hand gesture recognition with 3D convolutional neural networks", IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, United States, 7–12 June 2015.

[24] Microsoft Kinect, "Kinect for Windows", https://developer.microsoft.com/windows/kinect (24.12.2020).

[25] Ultraleap, "Leap Motion Controller", http://www.ultraleap.com/product/leap-motion-controller (24.12.2020).

[26] C. Kanan, G.W. Cottrell, "Color-to-grayscale: does the method matter in image recognition?", PloS one, Vol. 7, 2012, pp. e29740:1–7.

[27] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, "Wallflower: Principles and practice of background maintenance", International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999.

[28] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features", IEEE Conference on Computer Vision and Pattern Recognition, Kauai, HI, United States, 8–14 December 2001.

[29] S.A. Dabhade, M.S. Bewoor, "Real time face detection and recognition using haar-based cascade classifier and principal component analysis", International Journal of Computer Science and Management Research, 2012, pp. 59–64.

[30] R. Padilla, C. Costa Filho, M. Costa, "Evaluation of haar cascade classifiers designed for face detection", International Journal of Computer, Electrical, Automation, Control and Information Engineering, Vol. 6, No. 4, 2012, pp. 466–469.

[31] S. Mattoccia, F. Tombari, L. Di Stefano, "Efficient template matching for multi-channel images", Pattern Recognition Letters, Vol. 32, No. 5, 2011, pp. 694–700.

[32] Emgu CV, "Emgu CV: OpenCV in .NET", http://www.emgu.com (24.12.2020).

[33] C.O. Conaire, N.E. O'Connor, A.F. Smeaton, "Detector adaptation by maximising agreement between independent data sources", IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, United States, 17–22 June 2007.

BIOGRAPHIES

ANIL BAS is an assistant professor at the Department of Computer Engineering, Faculty of Technology, Marmara University, Turkey. He received his BSc degree in Electronics and Computer Education from Kocaeli University in 2011, MSc degree in Computer Engineering from Selcuk University in 2013 and PhD degree in Computer Science from the University of York, UK in 2018. His research interests include computer vision, computer graphics, image processing and human computer interaction.

HASAN ERDİNÇ KOÇER is an associate professor at the Department of Electrical and Electronics Engineering, Faculty of Technology, Selcuk University, Turkey. He received his BSc degree in Electronics and Computer Education from Marmara University in 1998, MSc degree in Electronics and Computer Systems Education and PhD degree in Electrical and Electronic Engineering from Selcuk University in 2001 and 2007, respectively. His research interests are related to image processing, computer vision, biometric identification and pattern recognition.