EURASIP Journal on Image and Video Processing

RESEARCH Open Access

Robust hand gesture recognition using multiple shape-oriented visual cues

Samy Bakheet[1]* and Ayoub Al-Hamadi[2]

*Correspondence:
samy.bakheet@fci.sohag.edu.eg
[1]Faculty of Computers and Information, Sohag University, Sohag, Egypt
Full list of author information is available at the end of the article

Abstract

Robust vision-based hand pose estimation is highly sought but still remains a challenging task, due to its inherent difficulty partially caused by self-occlusion among hand fingers. In this paper, an innovative framework for real-time static hand gesture recognition is introduced, based on an optimized shape representation build from multiple shape cues. The framework incorporates a specific module for hand pose estimation based on depth map data, where the hand silhouette is first extracted from the extremely detailed and accurate depth map captured by a time-of-flight (ToF) depth sensor. A hybrid multi-modal descriptor that integrates multiple affine-invariant boundary-based and region-based features is created from the hand silhouette to obtain a reliable and representative description of individual gestures. Finally, an ensemble of one-vs.-all support vector machines (SVMs) is independently trained on each of these learned feature representations to perform gesture classification. When evaluated on a publicly available dataset incorporating a relatively large and diverse collection of egocentric hand gestures, the approach yields encouraging results that agree very favorably with those reported in the literature, while maintaining real-time operation.

Keywords: Hand gesture recognition, Shape oriented features, Fourier descriptor, Moments invariants, SVM

1 Introduction

Automatic vision-based recognition of hand gestures has recently received a great deal of researchers' attention in pattern recognition, computer vision, and biometrics communities, due to its potential applicability across a wide variety of applications, ranging from intelligent human-computer interfaces and human-machine communication to machine translation of sign languages of people with severe/profound speech and hearing impairments [1]. However, the task of recognizing hand gestures in unconstrained real-life scenarios has shown to be longstanding, intractable, and particularly challenging, due to a variety of potential inherent challenges presented by real-world environments, such as partial occlusion, drastic illumination variation, substantial background clutter, extreme hand pose variation, large intra-class variability within each class, changes in scale, and viewpoint and appearance [2]. Roughly speaking, it might be argued that gestures constitute the earliest form of human communication, since they are likely thought to have been

Springer Open

used long by the early man to communicate with others before spoken languages were developed. In the same vein, the need to automate the semantic interpretation process of human gestures by means of mathematical models enables automatic gesture recognition to be at the heart of a rich variety of automation technologies and real-world applications of intelligent vision, including natural user interfaces, intelligent surveillance, virtual reality, and motion gaming [3].

Body or sign language (i.e., gestures) can be easily recognized by a human being, due to the composition of vision and synaptic interactions that were created along human brain development [4]. However, replicating this kind of behavior (or skill) in machines is one of the most ambitious and yet challenging goals of a large number of researchers in computer vision, which remains a highly active area at the forefront of research. To accomplish this seemingly arduous task, some potentially difficult issues needs to be solved, including how to properly separate objects of interest from the background in target images and which digital image-capturing technologies and classification approaches are more appropriate or convenient to use than others.

The rapid revolutionary pace of modern technological advances and innovation in instrumentation, computational hardware, and software has contributed significantly to the unprecedented explosive evolution of new acquisition devices, such as Kinect and Leap Motion which allow to obtain a maximally informative description of human body and hand poses in monocular RGB images or videos [5–7]. Ever since the development and prevalence of such relatively cheap motion-sensing devices, vision-based hand gesture recognition not only has become a staple in the thriving field of human-computer interaction (HCI), but it also has been widely applied in diverse application areas, such as medicine, computer graphics, sign language translation, robotics, and augmented reality [8, 9].

In this work, we propose to address the relatively robust deployment of hand gesture recognition in uncontrolled real-world scenarios, which is an increasingly prominent and promising field of research, with potential applicability in various domains of human-computer interfaces and security industry. Nevertheless, in real-world environments, it remains too difficult or challenging to deploy robust gesture recognition, due to the more complex processing challenges posed from uncontrolled realistic scenes, such as background clutter, fast motion, occlusions, illumination variations, and ambiguous hand locations in the presence of other people or skin-colored objects [10].

In fact, the highly non-rigid nature of human hand in video sequences, resulting, for example, from large variations in hand pose, high dimensionality of hand motion, severe self-occlusion and self-similarity of hand fingers, or erratic hand motion, still presents an overwhelming challenge to vision-based hand detection and gesture recognition. Furthermore, while real-time performance is a matter of great concern in computer vision and related areas, especially for embedded systems, the majority of existing state-of-the-art systems for gesture recognition make use of sophisticated feature extraction and learning algorithms that constitute a formidable barrier to the real-time performance of these systems. The automatic recognition of hand gestures is still an evolving and open research area, due to the lack of a general-purpose model and most existing approaches remain limited in their performance and robustness. Therefore, new approaches are required to solve the issues with the current approaches.

This paper proposes a real-time method for hand gesture recognition based on an optimized shape representation build from multiple shape cues. The proposed method proceeds as follows. As a preliminary preprocessing step, the hand region of interest (ROI) is first segmented from the input depth map image. After the segmentation step, as shape features, a rich set of representative descriptors including both boundary-based (Fourier descriptors and curvature features) and region-based (moments invariants and moment-based features) is then extracted from the hand silhouette or contour. A one-dimensional (1D) feature vector is generated and finally fed into a one-vs-all SVM classifier for gesture classification. The remainder of the paper is organized as follows. Section 2 presents related works on hand gesture recognition. Then, a detailed description for the proposed framework for hand gesture recognition is given in Section 3. In Section 4, the experiments conducted to evaluate the performance of the presented gesture recognition system are reported and the obtained results are discussed. Finally, in Section 5, conclusions are drawn and scope of further work is provided.

2 Related work

Over the course of the past two decades or so, a great deal of research has been carried out (and still being carried out) on analyzing, modeling, and recognizing hand gestures in still images or video streams. Despite these long years of intense work, this problem is still open and remains challenging for the researchers in diverse fields, e.g., biometrics, pattern recognition, and computer vision communities. Therefore, further rigorous research is urgently required to contribute to the development of novel and innovative vision-based approaches and techniques to efficiently address the problem of hand gesture recognition.

According to the literature, vision-based methodologies for gesture recognition typically fall into two main categories: static and dynamic [11]. Static hand gesture recognition aims to classify hand gestures (i.e., otherwise referred to as hand postures) into a predefined number of gesture classes (or categories), relying only on appearance and hand posture cues captured from still images, without considering any motion cue. Hence, the recognition of static gestures sufficiently needs to process only a single image at the input of the classifier [11–17]. On the other side, dynamic hand gestures are modeled and recognized primarily by means of the use of temporal information (i.e., hand detection and tracking), in order to acquire the motion cues of hand gestures [18–21].

Upon scanning the literature, one finds that a significant body of existing work on hand gesture recognition mainly follows the typical steps of pattern analysis, starting with image preprocessing, segmentation, feature extraction, and classification [22–24]. In [25], an efficient multilayer perceptron (MLP) neural network-based approach is presented for the recognition of static gestures of alphabets in Persian Sign Language (PSL), where wavelet features are extracted using discrete wavelet transform (DWT) and an average recognition rate of 94.06% was obtained. In a similar vein, Cao et al. [26] introduce an approach for hand posture recognition, by integrating multiple heterogeneous image features and multiple kernels learning support vector machine (SVM). In their approach, the multiple trained kernels of SVM are used to predict the right category for the input unseen posture. The model was tested on the publicly available Jochen-Triesh hand posture dataset, achieving a competitive recognition accuracy of 99.16%. Additionally, in [27], an attention-based method is proposed for hand posture detection and recognition in presence of complex background objects, where the feature-based visual attention is constructed using a combination of both high-level (texture and shape) and low-level (color) image features. For posture classification, the multi-class SVM was used, achieving an overall accuracy of 94.36%.

Another thread of research has focused on the analysis of contour shape to recognize hand postures. For instance, in [11], the authors propose a method for recognizing 14 sign language gestures using contour-based features, where time curvature analysis is used for describing the silhouette of the hand shape and an SVM classifier is then learnt using the extracted features. The convolutional neural networks (CNNs) have been extensively investigated in several gesture recognition works [11, 16, 28], achieving state-of-the-art performance compared to baseline approaches. For example, in [16], a hybrid CNN-SVM approach is proposed, where CNN and SVM are employed as a feature extractor and gesture recognizer, respectively. Moreover, in [13], the authors develop a CNN-based system for hand gesture analysis on iPhone and robot arm platforms, where the binary hand masks are classified into multiple gestures.

Most recently, due to the recent advent of commercially available depth sensors, new opportunities have been opened up to advance hand gesture recognition by providing depth information [15, 17, 29]. In [29], Thalmann et al. make use of depth context features and random forest classification for palm pose tracking and gesture recognition in augmented reality, whereas in [15], the static posture of the hands and fingers is best described by employing static-dynamic voxel features that capture the amount of point clouds within a voxel. Furthermore, depth-based gesture recognition was widely perceived to have substantially greater discriminative ability than the color-based counterpart [17]. However, besides the usual inherent drawbacks of wearable depth sensors, e.g., unstable regions and holes in depth videos, their performance was significantly worse in outdoor scenarios, due to the coarse or severely reduced resolution of depth maps [17].

In regard to dynamic hand gesture recognition, one of the most common approaches is to make use of a descriptor (HOF) derived from the optical flow features to characterize dynamic gestures, which adequately constitutes a simplified, yet plausible model for learning and classification of hand motion patterns [18–21]. For instance, in [18, 19], an egocentric gesture classification system is developed to recognize the user's interactions with artworks in a museum. After camera-motion compensation, the feature points at different spatial scales in the hand's region of interest (ROI) are computed and tracked. Multiple local descriptors (e.g., HOG, HOF, and MBH) are then extracted from the spatiotemporal volume. After applying the power normalization to the above descriptors to decease the sparsity of the semi-local features, the normalized descriptors are encoded via Bag-of-Words (BoW) model and fed into linear SVMs for multiple gesture classification. In a similar vein, in [20, 21], dynamic gestures with a single movement, such as swiping left, right, up, and down, are recognized, based on the direction of the flow vectors computed over the entire duration of gestures, and using predefined thresholds on the gesture movement orientation.

Recently, there is also a growing body of research dominated by the idea that dynamic hand gestures can be usefully thought of as a generalization of static gestures [15, 17]. For instance, in [15], Jang et al. present a hierarchical approach to the estimation of hand gestures, which is based on a static-dynamic random forest to hierarchically predict the labels of hand gestures. The approach begins by recognizing static gestures at the top-level of the hierarchy to select a virtual object corresponding to the configuration of the detected hand (e.g., holding a stylus pen). Then, the dynamic gestures (e.g., pressing or releasing the button on the pen) conditional to the formally detected static gesture are recognized. In addition, Zhang et al. [17] have presented a comparison between traditional approaches using hand engineered features and deep-learning approaches (e.g., 2DCNN, 3DCNN, and recurrent models), and the reported results have shown that 3DCNN is much more appropriate for dynamic gesture recognition and combining both of color and depth information can significantly improve the recognition performance than the two image modalities separately.

3 Proposed method

This section describes the presented framework for hand gesture recognition using computer vision-based scheme. Figure 1 shows a general schematic outline of the framework. As illustrated in the above figure, the general scheme of the approach works as follows. As a preliminary preprocessing step, the hand region of interest (ROI) is segmented from the input depth map image and then the basic morphological operations of dilation, erosion, opening, and closing are applied to refine the segmented ROI to obtain a high-quality binary ROI containing only a hand gesture. As shape features, a set of representative descriptors containing boundary-based (such as Fourier descriptors, curvature features, etc.) and region-based (such as moments invariants and moment-based features, etc.) is then extracted from the hand silhouette (or contour). A one-dimensional (1D) feature vector is generated and then fed into a one-vs-all SVM classifier for gesture classification.

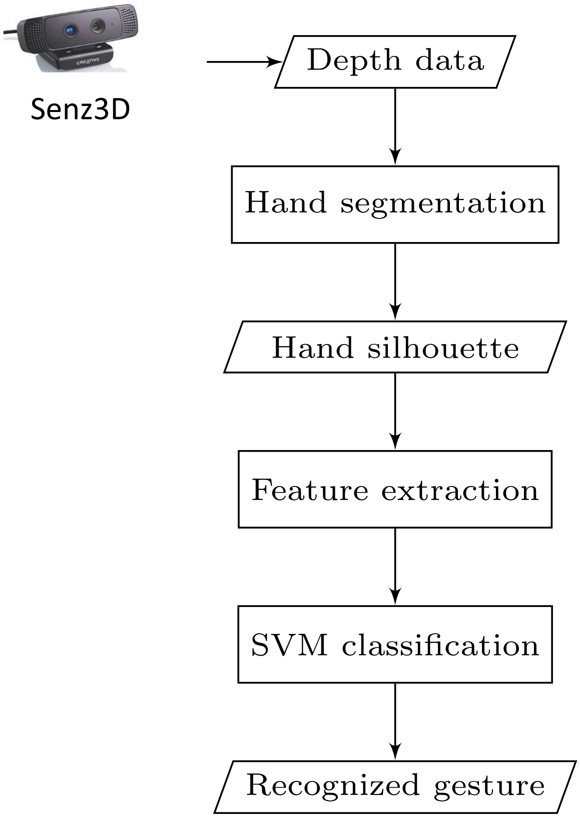

Fig. 1 The general workflow of the proposed framework

The SVM classifier is trained and evaluated on a real-world dataset [30] of static hand gestures acquired using a Creative Senz3D consumer depth camera. The subsections below provide the details of each component of the implemented recognition system.

3.1 Hand region segmentation

As mentioned previously, the objective of this preliminary step is to accurately detect and segment the gesture region of interest (i.e., hand region) from the input depth-map image. To accomplish this objective, the depth samples corresponding to the hand region are extracted from the depth map. In the present acquisition setting, since the hand that performs the gesture is the nearest object (i.e., closer distance object) to the depth camera, the segmentation algorithm can effectively locate the nearest hand to the camera using adaptive thresholding and hierarchical connected components analysis on the 2D depth-map image. More specifically, the point closest to the depth camera in the acquired depth-map image is initially detected. To this end, let $\boldsymbol{Z} = \{z_i, i = 1, \ldots, n\}$ be a given set of 3D points captured by the depth camera sensor, and also let $\boldsymbol{D}(\boldsymbol{Z}) = \{d(z_i), i = 1, \ldots, n\}$ be the corresponding depth map. Then, $d(z_{min})$ will be the depth of the nearest sample to the depth camera.

Due to image noise and other disturbing factors (e.g., image artifacts), an isolated artifact might be incorrectly identified as the closest point. To effectively tackle this problem, in the presented system, the selection of the closest point inevitably hinges primarily on the existence of a relatively large number of depth samples in a neighborhood of a certain size (e.g., 5×5) around each candidate closest point. A predefined threshold value is used to decide candidate closest points.

The cardinality of a set of extracted points is compared with the threshold value. If the cardinality does not exceed this value, the next candidate closest point is checked and the verification process is repeated until the admitted threshold is exceeded. Once the closest point is found, the set of all points of a depth value within a threshold τ_1 from z_{min} and with a distance from z_{min} smaller than another threshold τ_2 is calculated as follows,

$$\Omega = \{z_i | d(z_i) - d(z_{min}) < \tau_1 \wedge \|z_i - z_{min}\| < \tau_2\} \tag{1}$$

where $\wedge$ denotes the AND operator. It should be emphasized at this point that the proper values of the above two thresholds are often manually configured based on the size of hands in the original images of the dataset. In our experiments, we have varied the values of these thresholds and it has been empirically found that the "optimal" values for τ_1 and τ_2 are 10 and 40, respectively. Furthermore, in order to prevent the detection of extremely small non-hand objects and to avoid the confusion of isolated artifacts with hand objects of interest, appropriate additional precautions are to be taken, e.g., applying additional two low-pass filters on Ω. Firstly, an adaptive morphological open-close filter is iteratively applied to the resulting binary image in order to remove very small objects from the image, while other objects of a relatively large size or shape are selectively maintained. Such a filter is preferentially realized by a cascade of erosion and dilation operations, using locally adaptive structuring elements that can preserve the geometrical features of image objects.

During this step, another filter (namely size adaptive filter) is also applied to get rid of isolated components (i.e., amorphous objects) that do not meet a minimum pixel count (i.e., size). More specifically, after applying this filter, very small objects of average size less than a certain pre-defined threshold value (e.g., 5% of image size) are eliminated from the binary hand image. Now, after filtering out almost all unwanted image components and isolated artifacts, an improved Canny edge detection algorithm [31] is applied to selectively extract high-contrast hand region boundaries, where Gaussian filter was replaced by self-adaptive filter and morphological operator was used to refine the edge points detected and obtain single pixel level edge. Figure 2 shows an example of applying the hand segmentation scheme to extract the hand ROI from an input depth image. As is clear from the figure, the scheme can potentially realize high accuracy segmentation of hand samples from the scene objects and from the other body parts. Furthermore, the largest circle contained in the hand contour can be used to reliably locate the hand palm centroid, as fingertips are closely related to the convex defects, which are close to the start and end contour points of convex defects. Thus, it is possible to detect the fingertips from hand contour and convex defects [32, 33].

3.2 Feature extraction

Broadly speaking, feature extraction is fundamental but most challenging and time-consuming in the framework of any approach to pattern recognition. A wide variety of informative and distinctive features can be employed exclusively or combined with others for hand gesture recognition. In this work, the shape features that describe the shape structures of the segmented hand silhouettes are employed to represent different static hand poses. As robust shape-based features, a variety of visual features including Fourier descriptors, invariant moments, and curvature features are employed, which allow a much

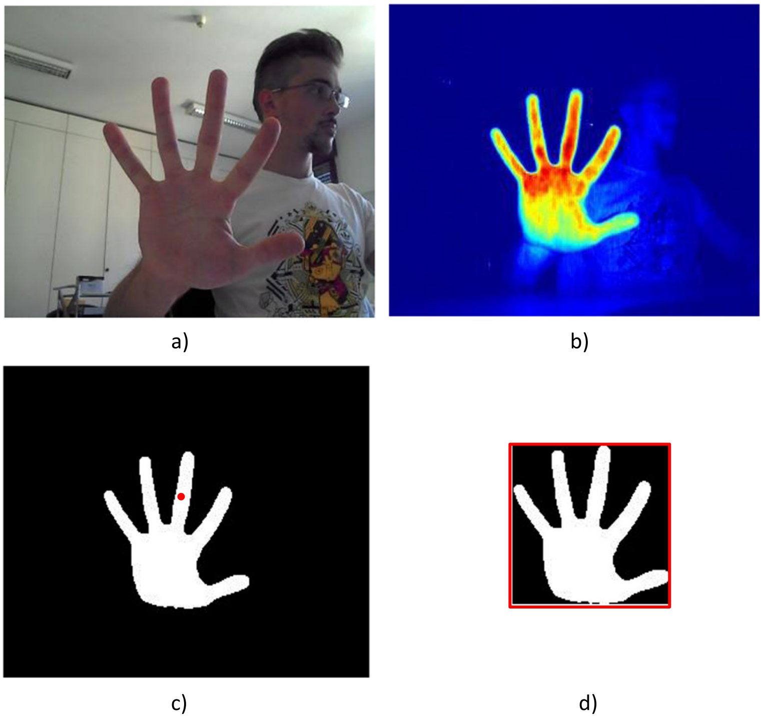

Fig. 2 Hand region of interest segmentation: **a** acquired color image, **b** acquired depth map, **c** binary mask image (the closest sample is depicted in red), and **d** extracted hand ROI

better characterization of static hand gestures. In the following subsections, we describe in more detail how these features are locally extracted from segmented hand regions.

3.2.1 Fourier descriptors

Fourier descriptors (FDs) of a given 2D shape depend on the notion that any 2D shape border (namely, contour) can be mathematically represented by a periodic complex function: $z_i = x_i + jy_i$, where $x_i, y_i, i = 0, 1, \ldots, n-1$ indicate x and y coordinates of the contour points. Consequently, the discrete Fourier transform coefficients can be determined as follows,

$$a_k = \frac{1}{n} \sum_{i=0}^{n-1} z_i \ \exp\left(-\frac{j2\pi ik}{n}\right), \ k = 0, 1, \ldots, n-1 \tag{2}$$

The adapted Fourier descriptors can be selectively obtained from the above coefficients a_k for instance by truncating the first two coefficients, a_0, a_1 and dividing the remaining coefficients by $|a_1|$, as follows,

$$b_k = \frac{|a_{k+2}|}{|a_1|}, k = 0, 1, \ldots, n-3 \tag{3}$$

It is straightforward to see that this choice of coefficients will not only guarantee that the generated descriptors are invariant with respect to shape translation, scale and rotation, but also ensure that they are independent of the choice of the starting point on the shape border.

3.2.2 Shape moments

Invariant moments that are essentially a set of nonlinear moment functions are often used in various pattern recognition applications for providing regional descriptors to capture global geometric characteristics of a particular object shape within a given frame/image. This set of nonlinear functions can be easily derived from regular moments. Spatial moments of order $(p+q)$ of a given object shape $f(x, y)$ is defined as follows:

$$M_{pq} = \int\int_{\mathbf{C}} x^p y^q f(x, y) dx dy \tag{4}$$

It is easy to see that the spatial moments in Eq. (4) are, in general, not invariant under the main geometrical transformations such as translation, rotation, and scale change. To obtain invariance under translation, these functions can be computed with respect the centroid of the object shape as follows,

$$\mu_{pq} = \int\int (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \tag{5}$$

where $(\bar{x}, \bar{y}))$ is the centroid that matches to the center of gravity of that shape. The normalized central moments are therefore given by:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\alpha}}, \quad \alpha = \frac{p+q}{2} + 1 \tag{6}$$

On the basis of the above normalized central moments, a simple set of nonlinear moment functions [34] can be defined, which is invariant to translation, rotation and scale changes:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + (2\eta_{11})^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{03} - \eta_{21})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2\right. \\
&\quad \left. -3(\eta_{03} + \eta_{21})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \\
&\quad \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2\right] \\
\phi_6 &= (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2\right] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2\right. \\
&\quad \left. -3(\eta_{03} + \eta_{21})^2\right](\eta_{30} - 3\eta_{12})(\eta_{03} + \eta_{21}) \\
&\quad \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2\right]
\end{aligned}$$

3.2.3 Curvature features

In a way quite analogous to the Mel Frequency Cepstral Coefficients (MFCC) features extracted from the frequency-domain signal and dominantly used in speech and speaker recognition systems, a concise set of shape descriptors can be directly computed from the cepstrum of the shape curvature, as follows. Firstly, the shape curvature is normally computed along the hand contour from the Freeman chain-code representation [35]. Then, the cepstrum of the curvature signal (i.e., spectrum) is obtained using discrete Fourier transform. Finally, as a shape descriptor, a certain number of the largest coefficients are chosen to be added to the feature vector. Numerous experimentations have been performed and revealed that a small set of cepstrum coefficients can sufficiently reconstruct the curvature function, with a compression ratio of up to 10:1 in the original signal length [36].

3.2.4 Moment-based features

In addition to the prior features, a set of other features generated by the central moments can be extracted. The existing analogy between hand image moments and mechanical moments contribute to a deeper understanding of the second-order central moments, i.e., μ_{11}, μ_{02} and μ_{20}. These moments construct the components of the inertial tensor of the object's rotation about its center of gravity:

$$\mathcal{J} = \begin{bmatrix} \mu_{20} & -\mu_{11} \\ -\mu_{11} & \mu_{02} \end{bmatrix} \tag{7}$$

Upon the inertial tensor analogy, a set of invariant features using second-order central moments can be extracted. For instance, the major inertial axis can be obtained by the roots of the eigenvalues of the inertial tensor as follows,

$$\lambda_{1,2} = \sqrt{\frac{1}{2}(\mu_{02} + \mu_{20}) \pm \left(4\mu_{11}^2 - (\mu_{02} - \mu_{20})^2\right)^{1/2}} \tag{8}$$

where λ_1 and λ_2 correspond to the semi-major and semi-minor axes, respectively, of the ellipse that can be intuitively considered as a fairly good approximation of the hand object.

Furthermore, the orientation of the hand object defined as the tilt angle between the x-axis and the axis around which the hand object can be rotated with minimal inertia can also be obtained as follows:

$$\varphi = \frac{1}{2}\arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right) \tag{9}$$

where φ is the angle between the x-axis and the semi-major axis. The above principal value of the arc tangent is expressed in radians (in the interval $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$). A variety of other shape features such as eccentricity ε and roundness κ convey shape identification information or could possibly provide some perceptual representation of hand shape. The roundness or circularity κ is properly defined to be the ratio of the area of an object to the area of a circle circumscribing that object. More specifically, κ can be determined by simply dividing the square of the perimeter ℓ by the area of the object A. From a geometric perspective, it follows that out of all shapes, the circle has the maximum area for a given perimeter/circumference. Thus, κ can explicitly be given as follows,

$$\kappa = \frac{\ell^2}{4\pi A} \tag{10}$$

Notably $\kappa = 1$ for a circle, whereas for other objects $\kappa > 1$. The eccentricity ε can readily be calculated from the second-order central moments as follows:

$$\varepsilon = \frac{(\mu_{20} - \mu_{02})^2 - 4\mu_{11}^2}{(\mu_{20} + \mu_{02})^2} \tag{11}$$

All the computed descriptor vectors are concatenated resulting in a single feature vector for each hand posture (see Fig. 3). The resultant feature vectors can be independently normalized to the range [0,1] by means of a linear transformation to establish the best fit of the characteristics of the learning model. The normalized feature vectors that encode much of the shape information are then fed into the next classification layer for supervised gesture recognition.

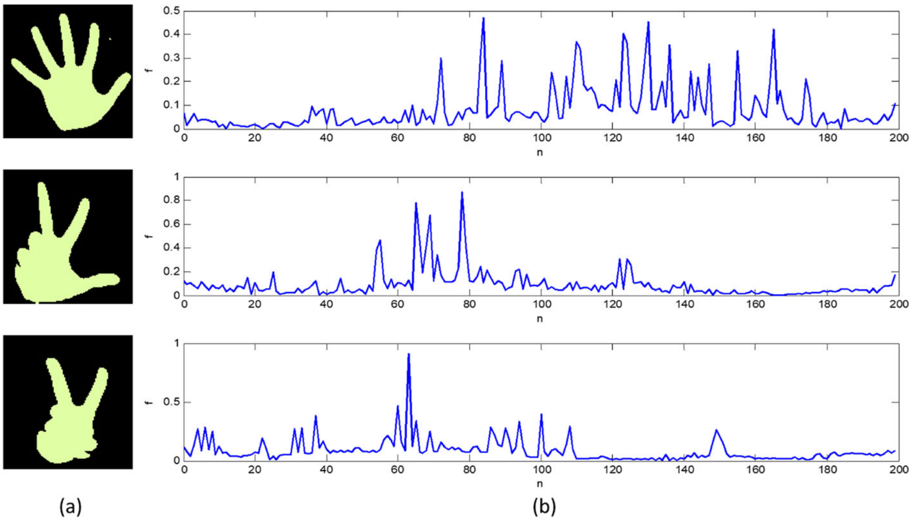

Fig. 3 Three sample hand postures along with their shape signatures: **a** segmented hand postures and **b** computed posture signatures

3.3 Gesture classification

This section explains the classification module as the last stage in the proposed hand gesture recognition framework, which primarily aims at classifying static gesture images based on gesture representations obtained using shape feature descriptors, as described in the previous section. The current gesture recognition task can normally be modeled as a multi-class classification problem with a unique class for each gesture category, and the ultimate goal is to assign a class label to a given gesture image. To achieve this objective, a set of one-vs-all nonlinear SVMs (Support Vector Machines) with RBF (Radial Basis Function) kernels is trained for each gesture category. Each SVM classifier learns to separate images that belong to a certain gesture category from those that do not. The final decision on the output category for a given image is made according to the classifier with the highest score.

Broadly speaking, there is a wide variety of machine learning (ML) algorithms that can be utilized to learn a model to recognize certain patterns of hand gesture robustly and effectively. In this work, we propose to apply an ensemble of SVMs to hand gesture classification as a first step towards integration into a full gesture recognition framework. The motivation behind using SVMs is the superior generalization capabilities as well as their well-deserved reputation of a highly accurate paradigm. Moreover, thanks to the kernel principle, SVMs could be conveniently trained with almost no parameter tuning, since they can reliably find optimal parameter settings automatically, with low computational overhead independent of their dimensionality. All this enables SVMs to be very promising and extremely competitive with other classification techniques in pattern recognition.

SVMs were originally designated [37] to specifically handle dichotomic classes in a higher-dimensional feature space, where an optimal maximum-margin separating hyperplane is constructed. To determine the maximum margin, two parallel hyperplanes are built, one on each side of the separating hyperplane, as illustrated in Fig. 4. The primary aim of SVM is then to find out the separating hyperplane that maximizes the distance between the two parallel hyperplanes. Intuitively, a good separation is accomplished by the hyperplane that has the largest distance to the closest training data points of any class (so-called functional margin), since generally the larger the margin, the lower the generalization error of the classifier. Formally speaking, assume $\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^d, y_i \in$

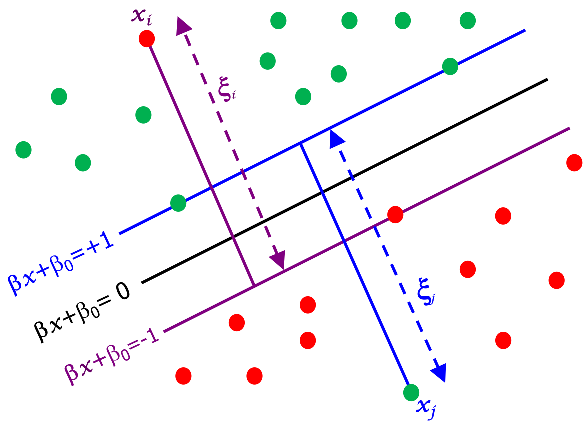

Fig. 4 Generalized optimal separating hyperplane

$\{-1, +1\}\}$ be a training set, Coretes and Vapnik [37] have argued that this problem is best addressed by allowing some data points to violate the margin constraints. These potential violations can be elegantly formulated through the use of some positive slack variables ξ_i and a penalty parameter $C \geq 0$ that penalize the margin violations. Therefore, the optimal maximum-margin separating hyperplane is obtained by solving the following quadratic programming. (QP) problem:

$$\min_{\boldsymbol{\beta},\beta_0} \quad \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_i \xi_i \tag{12}$$

subject to $\quad (y_i(\langle \mathbf{x}_i, \boldsymbol{\beta}\rangle + \beta_0) \geq 1 - \xi_i \quad \forall i) \wedge (\xi_i \geq 0 \quad \forall i).$

Geometrically, $\boldsymbol{\beta} \in \mathbb{R}^d$ is a vector passing through the origin and measured perpendicularly to the separating hyperplane. The offset parameter β_0 is included in the optimization problem to allow the margin to increase and to not force the hyperplane to go through the origin that restricts the practicality of the solution. From the computational point of view, it is probably most convenient to solve SVM in its dual formulation by first forming the Lagrangian associated with the problem and then optimizing over the Lagrange multiplier $\boldsymbol{\alpha}$ instead of the primal variables. The decision function, also known as hypothesis, in the classifier is described by a weight vector: $\boldsymbol{\beta} = \sum_i \alpha_i \boldsymbol{x}_i y_i,\ 0 \leq \alpha_i \leq C$. The training instances $\boldsymbol{x}_i$ that lie at the edge of their respective class space have non-zero α_i. These data points are called *support vectors*, since they uniquely define or "support" the maximum-margin hyperplane and are closest to it.

In the presented framework, several classes of hand gestures are defined and one-vs-all SVMs with RBF kernel are independently trained to learn the pattern of each class of gestures, based on the shape features extracted from the gesture images in the training set. Among the most popular and powerful kernels, we have selected the more related with our work which is the RBF kernel (also referred to as Gaussian kernel) given as follows:

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(\|\boldsymbol{x} - \boldsymbol{y}\|^2 / \left(2\sigma^2\right)\right) \tag{13}$$

where σ is the kernel width that can typically be seen as a tuning or smoothing parameter. It is perhaps pertinent to mention here that the SVMs with RBF kernels have emerged as a flexible and powerful tool for creating predictive models that can potentially handle non-linearly separable data by mapping the input feature space to a higher dimensional feature space (denoted as a kernel space) in the hope that in this higher-dimensional space the data could become more easily separated or better arranged. More specifically, in the higher-dimensional space, linear boundary functions are constructed to make it possible to perform the linear separation for classification. When brought back to the input space, these functions could potentially produce non-linear boundaries to effectively separate data of different classes. Another point worth mentioning here is that, when using the RBF kernel, there are some important parameters such as c and γ that need to be properly tuned. Therefore, several tests have been performed in order to establish optimum values for the latter two parameters. Notably, improper selection of such parameters tends to make the classification model highly prone to either overfitting or underfitting that in turn causes the classifier to deliver a poor generalization capability and poor classification performance.

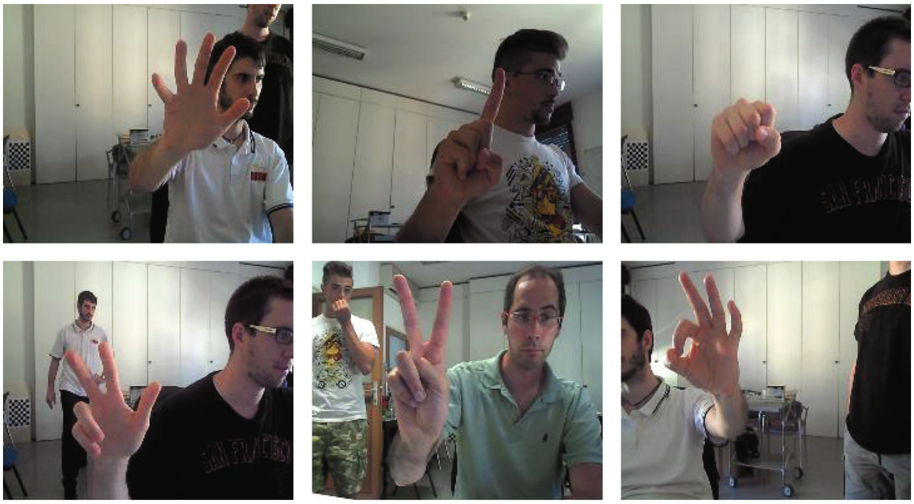

Fig. 5 Sample snapshots of six gestures from the dataset [30] used in our experiments

4 Results and discussion

In this section, experiments are carried out and their results are presented along with a detailed performance comparison with other common approaches for gesture recognition to reveal the effectiveness and feasibility of the proposed recognition framework. In order to investigate the performance of the proposed recognition framework, a series of experiments were conducted using a large-scale publicly available dataset [38] of real-world indoor images. The dataset is captured by using an Intel-Creative Senz3D camera (also known as SoftKinetic DepthSense 325) that houses an upgrade RGB camera along with a time-of-flight (ToF) depth sensor. It is worth of mentioning that Creative Senz3d cameras adopt a ToF approach of depth measurement instead of the trigonometry-based depth computations performed the Kinect devices and other structured light-based products. In addition, the acquiring device can produce dense depth maps with a resolution of 320×240 pixels at a frame rate ranging between 6 and 60 fps (frame per second). A fairly good initial full-range depth map can be obtained by applying a confidence threshold (i.e., a function of the IR response amplitude).

As in this work, we are mainly interested in the depth map data that represent the distance of objects from the depth sensor, the visual color data that represent the perceived light by the sensor from the objects are not yet being exploited. The dataset includes a total of 11 distinct classes of static hand gestures. In an indoor laboratory environment, each gesture was performed by four different subjects whose hands came in various sizes. Each participant was instructed to perform each gesture 30 times, yielding a total of 120 shots from all participants per gesture or a total of 330 shots for all gestures per participant. Hence, there are a total of $11 \times 4 \times 30 = 1320$ still images (along with a similar number of depth maps) included in this dataset. The dataset is now available free of charge solely for research purposes on the Internet[1]. This link permits interested researchers in the area of gesture recognition to readily access and download a verbatim copy of the entire dataset's files. Sample snapshots of six gestures involved in our experiments are shown in Fig. 5.

[1] http://lttm.dei.unipd.it.

It is perhaps relevant to indicate here that the dataset employed in this work is claimed to possess several challenging properties that closely simulate or incorporate real-world situations, including variations in illumination and scale, partial occlusion, and clutter in the scene. In addition, there are several different gestures that have similar number of raised fingers, gestures of fingertips touching each other, and gestures involving fingers very close to each other. Therefore, this dataset is most likely to be used as a defacto standard benchmark by many communities in computer vision and pattern recognition to evaluate the performance of hand gesture recognition models.

In order to validate the effectiveness of the proposed gesture recognition system, a series of comprehensive experiments and evaluations have been carried out using a 10-fold cross-validation framework on the static hand gesture dataset described previously. To depict a pictorial illustration of the performance of the developed system, the contingency table (or so-called confusion matrix) is generated, which is widely regarded as a comprehensive and intuitive reflection of the recognition capabilities of the framework. In the presented work, given a total of 11 classes of static hand gestures of interest, the confusion matrix that counts instances of correct and incorrect classifications would have a layout in a square tabular form of size 11 × 11, where the diagonal entries count the number of gestures that are correctly classified, while the off-diagonal entries count the number of misclassifications. In order to inspect and analyze the class-wise performance of the proposed recognition system, the detailed class wise accuracies are presented in the form of confusion matrix in Table 1.

In light of the figures presented in the preceding confusion matrix (Table 1), a number of interesting observations can be made. The first and most significant result in the confusion matrix is that the recognition accuracy is either close to or slightly greater than 90% for most gesture classes considered in this work and surely greater than 80% for all gesture classes. In addition, gestures G3 and G11 are recognized with an outstanding accuracy of 100%. Another observation from the above table is that the larger the value of the matrix diagonal entry, the higher the recognition rate of the corresponding gesture. It is also amply evident from the table that the presented gesture recognition framework achieves an average recognition rate of 93.3% for 11 static hand gestures based on the results of extensive experiments. Furthermore, upon closer inspection of the figures in the table, it is apparent that most gestures are successfully recognized with a fairly good hit ratio, while a few gestures are ineluctably being confused. The confusion matrix further depicts that,

Table 1 Confusion matrix for the proposed 11-class hand gesture recognition system

Gesture	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11
G1	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00
G2	0.00	0.89	0.00	0.08	0.00	0.00	0.03	0.00	0.00	0.00	0.00
G3	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G4	0.00	0.00	0.00	0.91	0.05	0.00	0.02	0.00	0.00	0.00	0.02
G5	0.00	0.00	0.00	0.03	0.93	0.00	0.00	0.00	0.00	0.04	0.00
G6	0.00	0.02	0.00	0.00	0.00	0.89	0.00	0.00	0.03	0.00	0.06
G7	0.00	0.00	0.04	0.00	0.00	0.00	0.87	0.00	0.00	0.00	0.09
G8	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.97	0.02	0.00	0.00
G9	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.04	0.88	0.00	0.01
G10	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.94	0.00
G11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

as expected, there are a few confusions (misclassifications) between similar gestures with very small values, such as G2, G7, and G9. Another important observation is that gestures G2 and G3 that only differ in the thumb position are often mutually confused and thus classified as having the same category, because some of the variations in thumb position are extremely difficult to be properly spotted in the detected hand silhouettes. In regard to gestures G7 and G11, they tend to obviously confuse each other, since they have a single raised finger that frequently presents a notorious challenge for many state-of-the-art methods to effectively characterize the shape of the finger using shape oriented features with high stability. Figure 6 depicts a set of misclassified gestures. Due to the touching fingers, it would be practically difficult for gesture G9 to be recognized with a success rate of above 90%. This finding is in close accordance with those from several previous studies in literature, particularly in view of the fact that this gesture is most likely to be incorrectly recognized by many gesture recognition approaches based on fingertip detection.

At this point, it should be pointed out that it is now an experimentally well-established fact that the shape oriented features not only have a great potential to offer finer quantitative discrimination between candidate gestures, but also they turn out to be consistently robust across multiple repetitions of the same gesture.

Globally speaking, the obtained results are encouraging and promising and they show evidence with regard to the validity and efficacy of the proposed gesture recognition system, where an average recognition rate of over 93% was achieved (95% confidence interval of 0.89 to 0.97). Additionally, the performance analysis and experimental results clearly demonstrate that the proposed approach is competent and able to yield superior gesture recognition performance, without violating real-time requirements. In order to validate the usability and capability of the proposed recognizer, a number of experiments were conducted to explore the quality of the proposed approach. As a benchmark for the performance of our recognizer, we compare the recognition performance with that of several existing state-of-the-art recognition techniques [1, 38–40] Results of this comparison are summarized in Table 2.

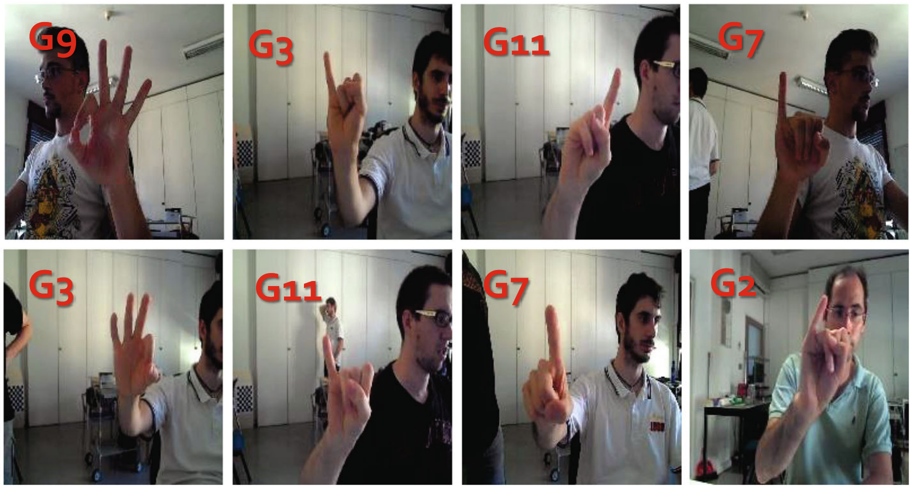

Fig. 6 Sample snapshots of some misclassified gestures

Table 2 Comparison of gesture recognition accuracy with other state-of-the-arts

Method	Accuracy (%)
Proposed method	93.3
Memo et al. [38]	90.0
Bakheet et al. [1]	90.6
Marin et al. [40]	89.7
Belongie et al. [39]	82.3

From this comparison, it turns out that the presented recognition system consistently outperforms the baselines established in the literature and competes very favorably with the state-of-the-art systems, by achieving higher recognition accuracy. It is worthy mentioning here that the aforementioned methods with which we compared our results have also been performed using similar experimental setups and operating environments. Consequently, the comparison turns out to be proper and quite fair. As a closing point to this discussion, it does also bear mentioning that all experiments performed in this work were implemented using Microsoft Visual Studio 2015 development tools and OpenCV vision open source library for basic image-manipulation functions. All tests and evaluations were carried out on a PC with Intel(R) Core(TM) i7 CPU - 3.07GHz, 8GB RAM and NVIDIA GeForce GTX 1080 GPU, running Windows 10 Professional (64 bits). As expected, despite the proposed gesture recognition system is structurally large with various interacting components, it is fast enough to deliver real-time processing performance ($\sim$3.0 ms latency), due to the usage of optimized GPU-accelerated algorithmic implementations in OpenCV library along with fast custom C++ functions. Table 3 clarifies the details of the total execution time (s) of the proposed recognition system.

5 Conclusion

This paper has introduced a framework for real-time hand gesture recognition using an optimal combination of multiple shape-oriented visual cues, where an effective shape-orbited descriptor is extracted from hand skeleton to estimate the hand skeletal posture. Finally, the extracted descriptors are fed as input into an ensemble of one vs. all SVMs to perform gesture classification and recognition. The results obtained from an extensive set of 10-fold cross-validation tests using a relatively large-scale gesture recognition dataset have been sufficiently encouraging to suggest that the proposed approach could achieve a highly competitive recognition performance against other state-of-the-art methods, in terms of both accuracy and computational time. Future work will deal with a twofold purpose. On one hand, it would be distinctly more advantageous for the generalizability of the obtained results to explore the experimental validation of the proposed framework on more challenging real-world gesture recognition datasets reflecting more realistic challenges in data processing. On the other hand, the learning model should be further refined, so that the recognition system can recognize a wider class of gestures (i.e., a set of gestures not limited to the gestural lexicon used for its training).

Table 3 Average execution time (s) for various phases of the proposed system

Phase	Runtime
Preprocessing and segmentation	1.42 ms
Feature extraction	0.54 ms
SVM classification	1.08 ms
Total	3.04 ms

Abbreviations

SVM: Support vector machine; ToF: Time-of-flight; HCI: Human-computer interaction ; MLP :Multilayer Perceptron; PSL Persian Sign Language; CNNs: Convolutional neural networks ROI: Region of interest; BoW: Bag-of-Words FDs: Fourier descriptors; MFCC: Mel frequency cepstral coefficients

Acknowledgements

The authors would like to thank the editor and the anonymous reviewers for their insightful comments and valuable suggestions that have enriched the quality of the present work.

Authors' contributions

Conceptualization, SB; methodology, SB; software, SB; validation, SB; formal analysis, SB; project administration, AA-H; funding acquisition, AA-H. All authors have read and agreed to the published version of the manuscript.

Funding

This work is funded by the Federal Ministry of Education and Research of Germany (BMBF) (Robo-Lab no 03zz04x02b, HuBa no. 03ZZ0470 367, RoboAssist no. 03ZZ0448L) within the Zwanzig20 Alliance 3Dsensation.

Availability of data and materials

Please contact the authors for data (in the experiment used videos or coding files with implemented method) requests.

Declarations

Ethics approval and consent to participate

Approved.

Consent for publication

Approved.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Faculty of Computers and Information, Sohag University, Sohag, Egypt. [2]Institute for Information Technology and Communications, Otto-von-Guericke-University Magdeburg, Magdeburg, Germany.

Received: 22 February 2021 Accepted: 2 July 2021
Published online: 19 July 2021

References

1. S. Bakheet, A. Al-Hamadi, Hand gesture recognition using optimized local gabor features. J. Comput. Theor. Nanosci. **14**(2), 1–10 (2017)
2. S. K. Leem, F. Khan, S. H. Cho, Detecting mid-air gestures for digit writing with radio sensors and a CNN. IEEE Trans. Instrum. Meas. **69**(4), 1066–1081 (2020). https://doi.org/10.1109/TIM.2019.2909249
3. R. Faugeroux, T. Vieira, D. Martinez, T. Lewiner, in *27th SIBGRAPI Conference on Graphics, Patterns and Images*, Simplified training for gesture recognition, (2014), pp. 133–140. https://doi.org/10.1109/SIBGRAPI.2014.46
4. J. Bransford, *How people learn: brain, mind, experience, and school: expanded edition (2000)*. (National Academies Press, Washington, DC, 2000)
5. S. Riofrio, D. Pozo, J. Rosero, J. Vasquez, Gesture recognition using dynamic time warping and kinect: a practical approach, (2017), pp. 302–308. https://doi.org/10.1109/INCISCOS.2017.36
6. A. Betancourt, P. Morerio, C. Regazzoni, M. R. Auterberg, A structure for deoxyribose nucleic acid. Circ. Syst. Video Technol. **25**(5), 744–760 (2015)
7. M. W. Cohen, N. B. Zikri, A. Velkovich, in *2018 11th International Conference on Human System Interaction (HSI)*, Recognition of continuous sign language alphabet using leap motion controller, (2018), pp. 193–199. https://doi.org/10.1109/HSI.2018.8430860
8. Technavio, Global robotics market. Res. Mark. Dublin Irel. **2015**, 2015–2019 (2015)
9. S. Bakheet, A. Al-Hamadi, A framework for instantaneous driver drowsiness detection based on improved HOG features and naïve Bayesian classification. Brain Sci. **11**(2), 240–254 (2021)
10. H. Cheng, L. Yang, Z. Liu, Survey on 3D hand gesture recognition. IEEE Trans. Circ. Syst. Video Technol. **26**(9), 1659–1673 (2016). https://doi.org/10.1109/TCSVT.2015.2469551
11. Y. Ren, X. Xie, G. Li, Z. Wang, Hand gesture recognition with multiscale weighted histogram of contour direction normalization for wearable applications. IEEE Trans. Circ. Syst. Video Technol. **28**(2), 364–377 (2018). https://doi.org/10.1109/TCSVT.2016.2608837
12. G. Serra, M. Camurri, L. Baraldi, M. Benedetti, R. Cucchiara, in *Proceedings of the 3rd ACM International Workshop on Interactive Multimedia on Mobile & Portable Devices*, Hand segmentation for gesture recognition in ego-vision (ACM Press, 2013). https://doi.org/10.1145%2F2505483.2505490
13. H. Song, W. Feng, N. Guan, X. Huang, Z. Luo, in *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*, Towards robust ego-centric hand gesture analysis for robot control, (2016), pp. 661–666. https://doi.org/10.1109/SIPROCESS.2016.7888345
14. D. Thalmann, H. Liang, J. Yuan, in *Computer vision, imaging and computer graphics theory and applications*. ed. by J. Braz, J. Pettré, P. Richard, A. Kerren, L. Linsen, S. Battiato, and F. Imai, First-person palm pose tracking and gesture recognition in augmented reality (Springer, Cham, 2016), pp. 3–15

15. Y. Jang, I. Jeon, T. Kim, W. Woo, Metaphoric hand gestures for orientation-aware VR object manipulation with an egocentric viewpoint. IEEE Trans. Hum.-Mach. Syst. **47**(1), 113–127 (2017). https://doi.org/10.1109/THMS.2016.2611824
16. P. Ji, A. Song, P. Xiong, P. Yi, X. Xu, H. Li, Egocentric-vision based hand posture control system for reconnaissance robots. J. Intell. Robot. Syst. **87**(3-4), 583–599 (2017). https://doi.org/10.1007/s10846-016-0440-2
17. Y. Zhang, C. Cao, J. Cheng, H. Lu, Egogesture: a new dataset and benchmark for egocentric hand gesture recognition. IEEE Trans. Multimedia. **20**(5), 1038–1050 (2018). https://doi.org/10.1109/TMM.2018.2808769
18. L. Baraldi, F. Paci, G. Serra, L. Benini, R. Cucchiara, in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Gesture recognition in ego-centric videos using dense trajectories and hand segmentation, (2014), pp. 702–707. https://doi.org/10.1109/CVPRW.2014.107
19. L. Baraldi, F. Paci, G. Serra, L. Benini, R. Cucchiara, Gesture recognition using wearable vision sensors to enhance visitors' museum experiences. IEEE Sensors J. **15**(5), 2705–2714 (2015). https://doi.org/10.1109/JSEN.2015.2411994
20. S. Hegde, R. Perla, R. Hebbalaguppe, E. Hassan, in *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, Gestar: real time gesture interaction for ar with egocentric view, (2016), pp. 262–267. https://doi.org/10.1109/ISMAR-Adjunct.2016.0090
21. S. Mohatta, R. Perla, G. Gupta, E. Hassan, R. Hebbalaguppe, in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Robust hand gestural interaction for smartphone based AR/VR applications, (2017), pp. 330–335. https://doi.org/10.1109/WACV.2017.43
22. S. Rautaray, A. Agrawal, Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**, 1–54 (2015). https://doi.org/10.1007/s10462-012-9356-9
23. P. Pisharady, M. Saerbeck, Recent methods and databases in vision-based hand gesture recognition: a review. Comp. Vision Image Underst. **141**, 152–165 (2015). https://doi.org/10.1016/j.cviu.2015.08.004
24. T. Orazio, R. Marani, V. Renò, G. Cicirelli, Recent trends in gesture recognition: how depth data has improved classical approaches. Image Vis. Comput. **52**, 56–72 (2016). https://doi.org/10.1016/j.imavis.2016.05.007
25. A. Karami, B. Zanj, A. Kianisarkaleh, Persian sign language (PSL) recognition using wavelet transform and neural networks. Expert Syst. Appl. **38**, 2661–2667 (2011). https://doi.org/10.1016/j.eswa.2010.08.056
26. J. Cao, Y. Siquan, H. Liu, P. Li, Hand posture recognition based on heterogeneous features fusion of multiple kernels learning. Multimedia Tools Appl. Springer. **75**(19), 11909–11928 (2016). https://doi.org/10.1007/s11042-015-2628-z
27. P. Pisharady, P. Vadakkepat, A. P. Loh, Attention based detection and recognition of hand postures against complex backgrounds. Int. J. Comput. Vis. **101**, 403–419 (2013). https://doi.org/10.1007/s11263-012-0560-5
28. M. Baydoun, A. Betancourt, P. Morerio, L. Marcenaro, M. Rauterberg, C. Regazzoni, in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hand pose recognition in first person vision through graph spectral analysis, (2017), pp. 1872–1876. https://doi.org/10.1109/ICASSP.2017.7952481
29. D. Thalmann, H. Liang, J. Yuan, in *International Joint Conference on Computer Vision, Imaging and Computer Graphics, vol. 598*, First-person palm pose tracking and gesture recognition in augmented reality (Spriner, 2016), pp. 3–15. https://doi.org/10.1007/978-3-319-29971-6_1
30. A. Memo, L. Minto, P. Zanuttigh, Exploiting silhouette descriptors and synthetic data for hand gesture recognition. STAG Smart Tools Apps Graph., 15–23 (2015)
31. L. Yuan, X. Xu, in *2015 4th International Conference on Advanced Information Technology and Sensor Application (AITS)*, Adaptive image edge detection algorithm based on canny operator, (2015), pp. 28–31. https://doi.org/10.1109/AITS.2015.14
32. M. A. Mofaddel, S. Sadek, in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT'10)*, Adult image content filtering: a statistical method based on multi-color skin modeling (IEEE, Luxor, 2010), pp. 366–370
33. S. Bakheet, A. Al-Hamadi, A hybrid cascade approach for human skin segmentation. Br. J. Math. Comput. Sci. **17**(6), 1–14 (2016)
34. M. K. Hu, Visual pattern recognition by moment invariants. Tr. Inf. Theory IRE. **8**, 179–187 (1962)
35. N. Alajlan, M. S. Kamel, G. Freeman, Multi-object image retrieval based on shape and topology. Signal Process. Image Commun. **21**, 904–918 (2006)
36. S. Bakheet, A. Al-Hamadi, Computer-aided diagnosis of malignant melanoma using Gabor-based entropic features and multilevel neural networks. Diagnostics. **10**, 822–837 (2020)
37. V. Vapnik, *The nature of statistical learning theory*. (Springer, New York, 1995)
38. A. Memo, P. Zanuttigh, Head-mounted gesture controlled interface for human-computer interaction. Multimedia Tools Appl. **77**, 27–53 (2018)
39. S. Belongie, J. Malik, J. Puzicha., Shape matching and object recognition using shape contexts. IEEE Trans. PAMI. **24**, 509–522 (2002)
40. G. Marin, F. Dominio, P. Zanuttigh, in *Proc. of IEEE International Conference on Image Processing (ICIP)*, Hand gesture recognition with leap motion and kinect devices (IEEE, 2014). https://doi.org/10.1109%2Ficip.2014.7025313

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.