SN Computer Science (2021) 2:436
https://doi.org/10.1007/s42979-021-00827-x

REVIEW ARTICLE

Methods, Databases and Recent Advancement of Vision-Based Hand Gesture Recognition for HCI Systems: A Review

Debajit Sarma[1] · **M. K. Bhuyan**[1]

Received: 27 April 2021 / Accepted: 19 August 2021 / Published online: 29 August 2021

Abstract
Hand gesture recognition is viewed as a significant field of exploration in computer vision with assorted applications in the human–computer communication (HCI) community. The significant utilization of gesture recognition covers spaces like sign language, medical assistance and virtual reality–augmented reality and so on. The underlying undertaking of a hand gesture-based HCI framework is to acquire raw data which can be accomplished fundamentally by two methodologies: sensor based and vision based. The sensor-based methodology requires the utilization of instruments or the sensors to be genuinely joined to the arm/hand of the user to extract information. While vision-based plans require the obtaining of pictures or recordings of the hand gestures through a still/video camera. Here, we will essentially discuss vision-based hand gesture recognition with a little prologue to sensor-based data obtaining strategies. This paper overviews the primary methodologies in vision-based hand gesture recognition for HCI. Major topics include different types of gestures, gesture acquisition systems, major problems of the gesture recognition system, steps in gesture recognition like acquisition, detection and pre-processing, representation and feature extraction, and recognition. Here, we have provided an elaborated list of databases, and also discussed the recent advances and applications of hand gesture-based systems. A detailed discussion is provided on feature extraction and major classifiers in current use including deep learning techniques. Special attention is given to classify the schemes/approaches at various stages of the gesture recognition system for a better understanding of the topic to facilitate further research in this area.

Keywords Human–computer interaction (HCI) · Vision-based gesture recognition (VGR) · Static and dynamic gestures · Deep learning methods

Introduction

In this period of innovation, where we are profound into the information age, technological progression has arrived at such a point that nearly everybody in each nook and corner of the world independent of any discipline, has interacted with computers somehow or the other. However, in general, a typical user ought not to need to secure computer education to utilize computers for basic undertakings in regular day-to-day life. Human–computer interaction (HCI) is a field of study which plans to encourage the communication of clients, regardless of whether specialists or fledglings, with computers in a simple way. It improves user experience by distinguishing factors that help to diminish the expectation to learn and adapt for new users and furthermore gives arrangements like console easy routes and other navigational guides for common users. In designing an HCI system, three main factors should be considered: functionality, usability and emotion [73]. Functionality denotes actions or services a system avails the user. However, a system's functionality is only useful if the user can exploit it effectively and efficiently. The usability of a system denotes the extent to which a system can be used effectively and efficiently to fulfill user requirements. A proper balance between functionality and usability results in good system design. Taking account of emotion in HCI includes designing interfaces that are pleasurable to use from a physiological, psychological, social, and aesthetic perspective. Considering all three factors, an interface should be designed to fit optimally between the user, device, and required services. Figure 1 illustrates this concept.

✉ Debajit Sarma
s.debajit@iitg.ac.in

[1] Department of Electronics and Electrical Engineering, IIT Guwahati, Guwahati, India

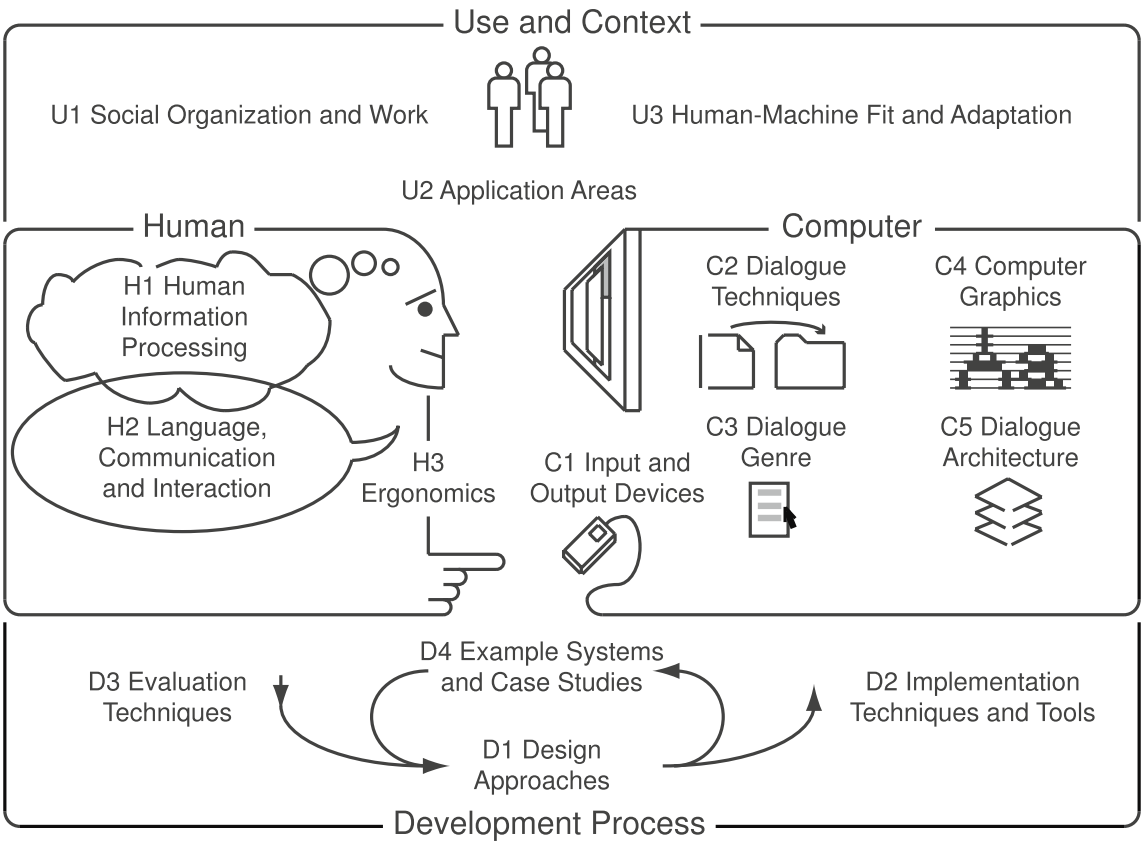

Fig. 1 Overview of human–computer interaction [73]

In recent years, significant effort has been devoted to body motion analysis and gesture recognition. With the increased interest in human–computer interaction (HCI), research related to gesture recognition has grown rapidly. Along with speech, they are the obvious choice for natural interfacing between a human and a computer. Human gestures constitute a common and natural means for nonverbal communication. A gesture-based HCI system enables a person to input commands using natural movements of the hand, head, and other parts of the body [171] (Fig. 2). And since the hand is the most widely used body part for gesturing apart from face [93], hand gesture recognition from visual images forms an important part of this research. Generally, hand gestures are classified as static gestures or simply postures and dynamic or trajectory-based gestures. Again, dynamic or trajectory-based gestures can be isolated or continuous.

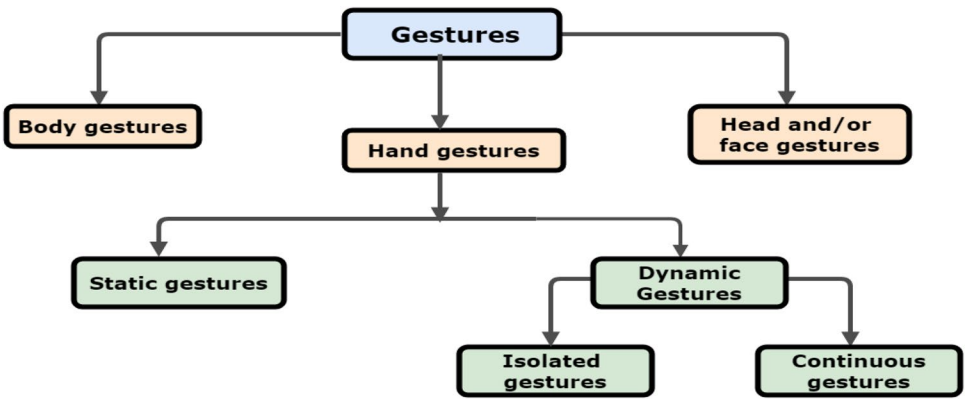

Fig. 2 Classification of different gestures based on used body-part

Gesture Acquisition

Before going into more depth, we want to first see how to acquire data or information for hand gesture recognition. The task of acquiring raw data for hand gesture-based HCI systems can be achieved mainly by two approaches [36]: sensor based and vision based (Fig. 3).

Sensor-based approaches require the use of sensors or instruments physically attached to the arm/hand of the user to capture data consisting of position, motion and trajectories of fingers and hand. Sensor-based methods are mainly as follows:

1. Glove-based approach measures position, acceleration, degree of freedom and bending of the hand and fingers. Glove-based sensors generally constitute flex sensors, gyroscope, accelerometer, etc.
2. Electromyography (EMG) measures human muscle's electrical pulses and decode the bio-signal to detect finger movements.
3. WiFi and radar use radio-waves, broad-beam radar or spectrogram to detect the changes in signal strength.
4. Others utilize ultrasonic, mechanical, electromagnetic and other haptic technologies.

Vision-based approaches require the acquisition of images or videos of the hand gestures through video cameras.

1. Single camera—it includes webcams, different types of video cameras and smart-phone cameras.
2. Stereo-camera and multiple camera-based systems—a pair of standard color video or still cameras capture two simultaneous images to give depth measurement. Multiple monocular cameras can better capture the 3D structure of an object.
3. Light coding techniques—projection of light to capture the 3D structure of an object. Such devices include PrimeSense, Microsoft Kinect, Creative Senz-3D, Leap Motion Sensor, etc.
4. Invasive techniques—body markers such as hand color, wrist bands, and finger marker. But the term vision based is generally used for capturing images or videos of the bare hand without any glove and/or marker. The sensor-based approach reduces the need for pre-processing and segmentation stage, which is essential to classical vision-based gesture recognition systems.

HCI Systems Architecture

The architecture of HCI systems can be broadly categorized into two groups based on their number and diversity of inputs and outputs: unimodal HCI systems and multimodal HCI systems [83] (Fig. 4).

1. *Unimodal HCI systems* Unimodal systems can be (a) vision based (e.g., body movement tracking [147], gesture recognition [146], facial expression recognition [115, 189], gaze detection [206], etc.), (b) audio based (e.g., auditory emotion recognition [47], speaker recognition [105], speech recognition [125], etc.), or (c) based on different types of sensors [113].
2. *Multimodal HCI systems* Individuals for the most part utilize different modalities during human to human correspondence. Subsequently, to survey a user's expectation or conduct extensively, HCI frameworks ought to likewise incorporate data from numerous modalities [162]. Multimodal interfaces can be arranged utilizing blends of data sources, for example, gesture and speech [161] or facial posture and speech [86] and so forth. Some of the major applications of multimodal systems

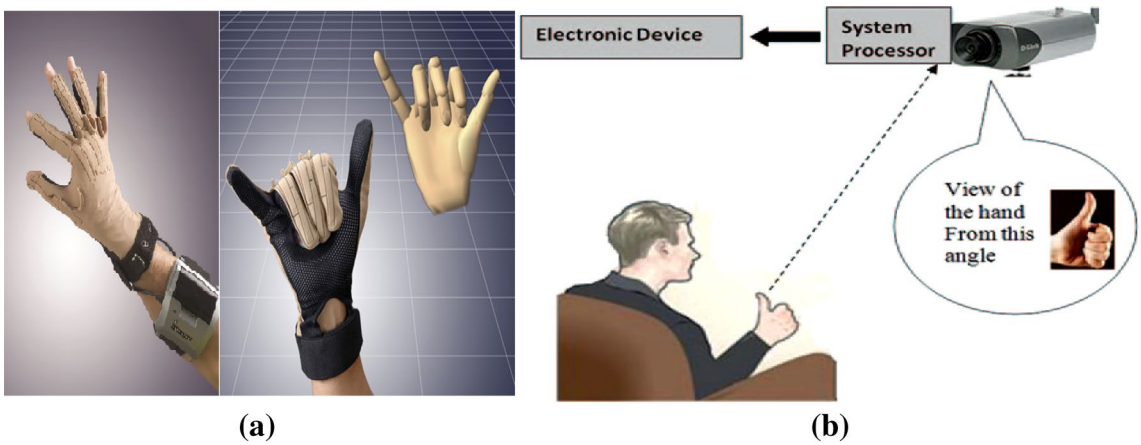

Fig. 3 Human–computer interaction using: **a** CyberGlove-II (picture courtesy: https://www.cyberglovesystems.com/products/cyberglove-II/photos-video), **b** vision-based system

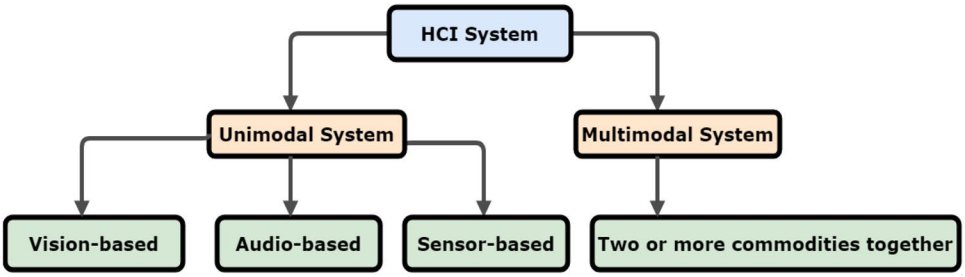

Fig. 4 General taxonomy of HCI system based on input channels

are assistance for people with disabilities [106], driver monitoring [204], e-commerce [9], intelligent games [188], intelligent homes and offices [144], and smart video conferencing [142].

Major Problems

It is an essential ability for computers to perceive the gestures of the hand visually for the future advancement of vision-based HCI. Static gesture recognition or pose estimation of the isolated hand, in constrained conditions, is roughly a solved problem to quite an extent. Notwithstanding, there are as yet numerous aspects of dynamic hand gestures that must be addressed, and it is an interdisciplinary challenge mainly due to three difficulties:

- Dynamic hand gestures vary spatio-temporally with assorted and different implications;
- The human hand has a complex non-unbending design making it hard to perceive; and
- There are as yet numerous difficulties in computer vision itself making it a poorly presented problem.

A gesture recognition system depends on certain subsystems associated in arrangement. In view of the arrangement of subsystems, the general exhibition of the framework is reliant on the precision of every subsystem. Along these lines, generally execution is profoundly influenced by a subsystem that is a "feeble connection". All the gesture-based applications are dependent on the ability of the device to read gestures efficiently and correctly from a stream of continuous gestures. To develop human–computer interfaces using the human hand has motivated researchers for continuous hand gesture recognition. Two major challenges present in the process of continuous hand gesture recognition are—constraints related to segmentation and problems in spotting the hand gestures perfectly in a continuous stream of gestures. But there are many other challenges apart from these which we will discuss now. More on constraints in hand gesture recognition can be found in [32] by the same authors.

- *Challenges in segmentation* Exact segmentation of the hand or the gesturing body part from the caught recordings or pictures still remains a challenge in computer vision for some limitations like illumination variation, background complexity, and occlusion.
  - *Illumination variation* The precision of skin color segmentation techniques is generally influenced by illumination variation. Because of light changes, the chrominance properties of the skin tones may change, and the skin color will appear different from the original color. Many methods use luminance invariant color spaces to accommodate varying illuminations [27, 66, 89, 90, 173]. However, these methods are useful only for a very narrow range of illumination changes. Moritz et al. found that the skin reflectance locus and the illuminant locus are directly related, which means that the perceived color is not independent of illumination changes [209]. Sigal et al. used dynamic histogram segmentation technique to counter illumination changes [199, 200]. In the dynamic histogram method, a second-order Markov model is used to predict the histogram's time-evolving nature. The method is applicable only for a set of images with predefined skin-probability pixel values. This method is very promising for videos with smooth illumination changes but fails for abrupt illumination changes. Also, this method is applicable to the time progression of illumination changes. In many cases where the illumination change is discrete, and input data is a set of skin samples obtained under randomly changed illumination conditions, this method performs poorly. Stern et al. used color space switching to track the human face under varying illumination [208]. King et al. used RGB color space and normalized it, and then converted it to YCbCr space. Finally, the Cb-Cr components are chosen to represent the skin pixel to reduce illumination effects. Kuiaski et al. performed a comparative study of the illumination dependency over skin-color segmentation methods [111]. They used naïve Bayesian classifier and histogram-based classification [87] over different skin samples obtained under four different illumination conditions. It

was observed that dropping the illumination component of a color space significantly reduces the illumination vulnerability of segmentation methods as compared with methods based on standard RGB color space. However, from the ROC curves obtained under different illumination conditions, it is evident that no color space is fully robust to illumination condition changes. Guoliang et al. grouped the skin-colored pixels according to their illumination component (Y) values in YCbCr color space into a finite number of illumination ranges [236]. It is evident from their analysis and previous literature review that the chrominance components are not independent of the illumination component. As shown in Fig. 5, the shape and position of the color histogram of the image change significantly due to the changes in illumination and the notion of independence can only be applied for a very narrow range of illumination changes. A Back Propagation Neural Network (BPNN) can be used to fit the data, which consists of the mean value of Cb and Cr, namely m_i, co-variance matrix C_i and the mean value of the i_{th} interval of Y, i.e., Y_i as given below

$$\begin{aligned} m_i &= E\left[\underline{x}_i\right] \\ \underline{x}_i &= \left[Cb_i, Cr_i\right]^T \\ C_i &= E\left[\left(\underline{x}_i - m_i\right)\left(\underline{x}_i - m_i\right)^T\right], \end{aligned}$$

where, $i = 1, 2, .., N$, $\underline{x}_i$ are the Cb-Cr samples belong to i_{th} illumination range. Here, Y_is are used as input and the Gaussian model $G_i(m_i, C_i)$ are the output. The model is then used to classify the skin and non-skin pixels for a particular illumination level. Bishesh et al. used a log-chromaticity color space (LCCS) by taking the logarithm of ratios of color channels and obtained intrinsic images to reduce the effect of illumination variations in skin color segmentation [53, 100]. However, LCCS gives a correct detection rate (CDR) of 64.84% and a false detection rate (FDR) of 4.50%, which are not so good results. Liu et al. used face detection to get the sample skin colors and then applied a dynamic thresholding technique to update the skin color model based on a Bayesian decision framework [130]. This method is dependent on the accuracy of the detected skin pixels from the face detection method, and it may fail if the face is not detected perfectly or the detected face has a mustache, beard, spectacles, or hair falling over it. Although a color correction strategy is used to convert the colors of the frame in the absence of a face, this solution is temporary and prone to error. In [190], the authors converted RGB colorspace into HSV and YCbCr color-cues to compensate illumination variation in the skin-segmentation method to segment the hand portion from the background. Biplab et al. has utilized a fusion-based picture explicit model for skin division to deal with the issue of segmentation under differing enlightenment conditions [31].

– *Background complexity* Another serious issue in gesture recognition is the appropriate division of skin-shaded items (e.g., hands, face) against an intricate static/dynamic background. An example of a complex background is shown in Fig. 6. Different types of complex backgrounds exist:

  - *Cluttered background (Static)* Although the background statistics are fairly constant, the background color and texture are highly varied. This kind of background can be modeled using Gaussian mixture models (GMMs). However, to model backgrounds of increasing complexity, more Gaussians should be included in the GMM.
  - *Dynamic background* The background color and texture change with time. Although hidden Markov models (HMMs) are often used to model signals that have a time-varying structure, unless they follow a well-defined stochastic process, their application to background modeling is computationally complex. The precision of skin division strategies is restricted because of the presence or movement of skin-colored objects behind the scenes which increment false positives.
  - *Camouflage* The background is skin-colored or contains skin-colored regions, which may abut the region of interest (e.g., the face, hands). For example, when a face appears behind a hand, this complicates hand gesture recognition, and when a hand appears behind a face, this complicates face region segmentation. These kinds of cases render it nearly impossible to segment the hand or face regions solely from pixel color information. Figure 7 shows a case of camouflage. The major problem with almost all segmentation methods based on the color space is that the feature space lacks spatial information on the objects, such as their shape.

These are the main issues of hand and face segmentation for gesture recognition. As shown in Fig. 6a, the background might be cluttered and have some skin-colored regions. In these conditions, it is difficult to segment actual skin regions (see Fig. 6b, c). Few works have reported signifi-

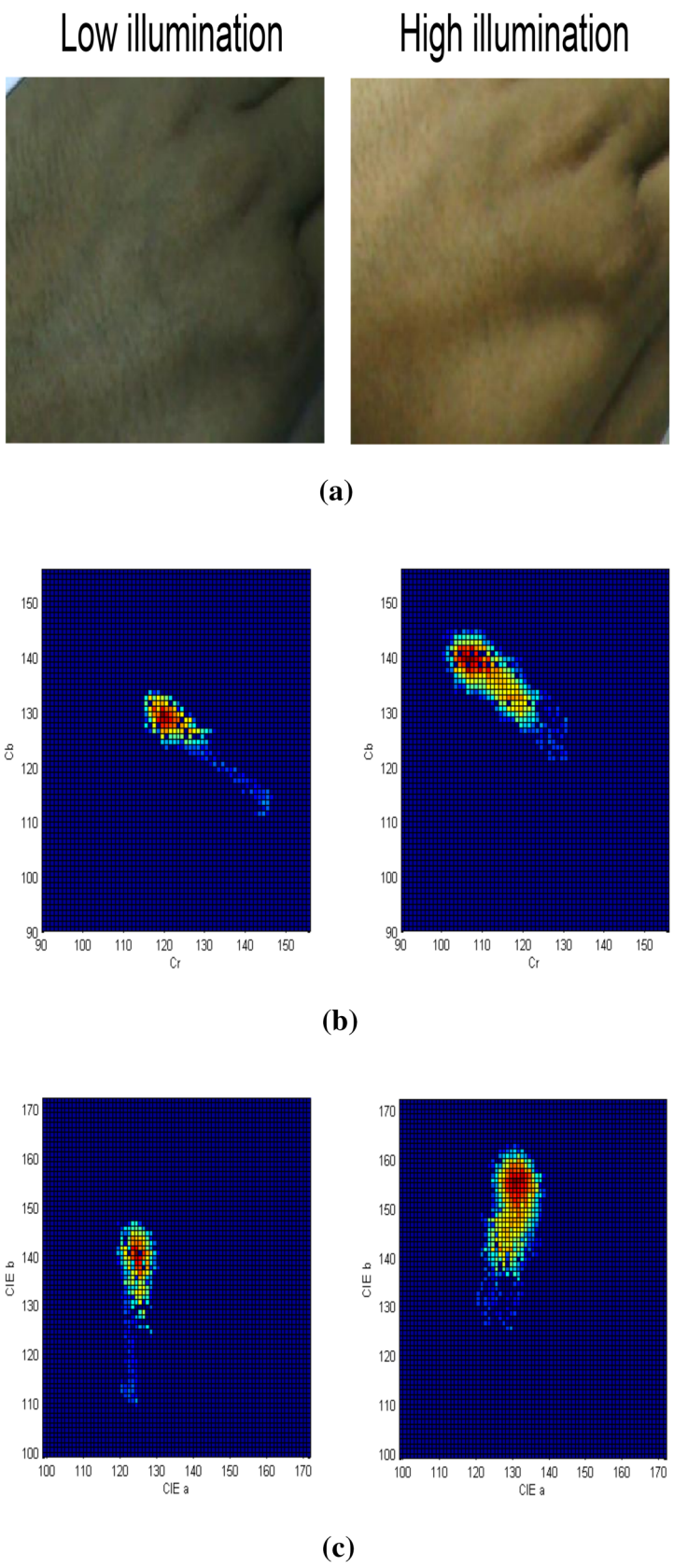

Fig. 5 Effect of illumination variations on perceived skin color: **a** skin color in low and high illumination conditions, **b** 2D color histogram in YCbCr space, and **c** 2D color histogram in CIE-Lab space

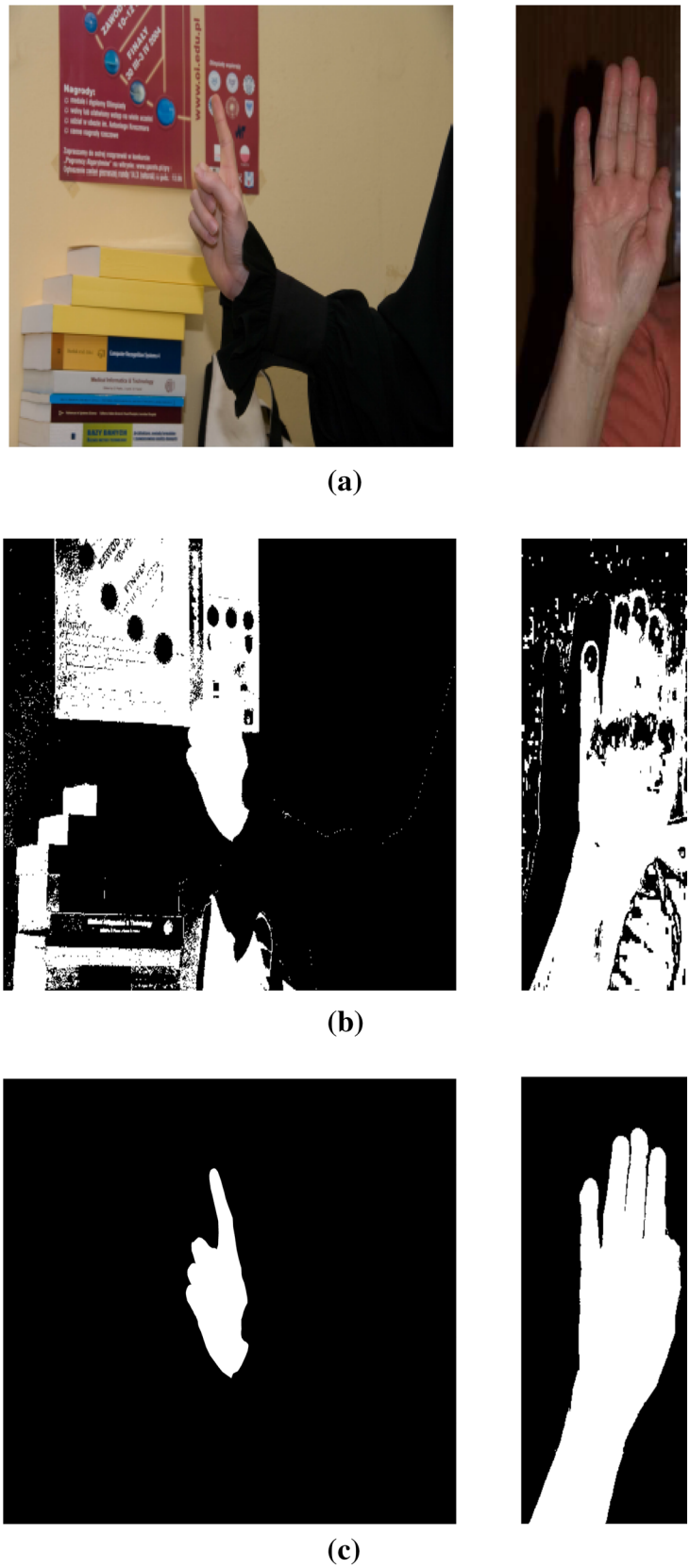

Fig. 6 Effect of complex background on skin color segmentation: **a** original images, **b** segmentation results, and **c** ground truth

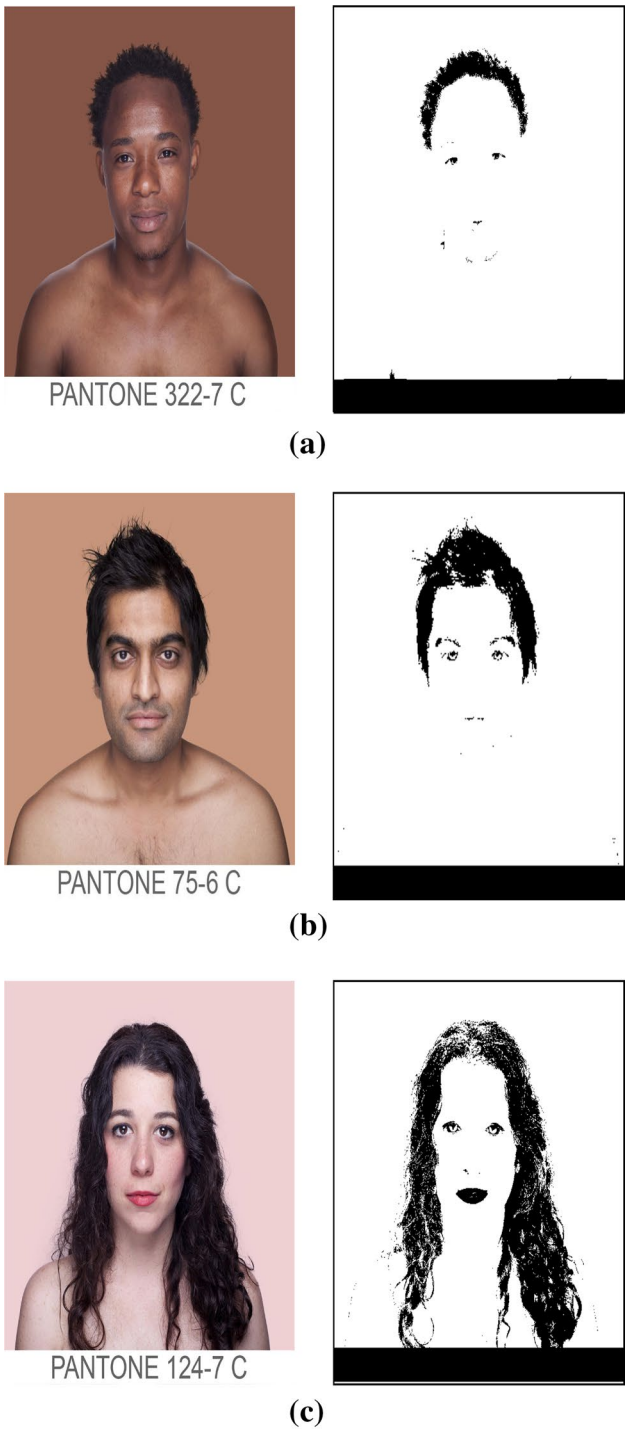

Fig. 7 Effect of camouflage on skin color segmentation (left column: original image, right column: segmented image): **a** African, **b** Asian, and **c** Caucasian

cant progress in this area. Phung et al. used skin texture information along with conventional pixel color details for skin region segmentation under a complex background [167]. This approach assumes that owing to the smooth texture of human skin, skin regions in images would be more homogeneous and have fewer edges than skin-colored regions in the background. The performance of this technique degrades when skin regions have many edges because of complex hand poses. Jhang et al. proposed an adaptive skin color segmentation method based on a skin probability distribution histogram (SPDH) [246]. The SPDH plots the total pixel count with a certain normalized skin probability with respect to the corresponding normalized skin probability of the pixel group in a particular image. Finally, the valley of SPDH is determined using a trained artificial neural network (ANN) as the optimum threshold for the image. The whole system's accuracy depends on how accurate the normalized skin probability is. Also, the color deviation histogram (CDH) method fails if the background color becomes similar to the skin color, as in that case, the color deviation will be very small for that group of pixels. Wang et al. combined the RGB and YCgCb color spaces and the texture information of the skin regions to detect the skin [225]. From the results, it is very evident that this method fails if there is a color similarity between the background and the skin regions. Avinash et al. proposed a skin color segmentation method by combining HSI and YCbCr color spaces with some morphological operations with labeling [13]. Their primary assumption was that the background color is different from the skin color, and thus this method fails drastically in the presence of skin-colored backgrounds. Pisharady et al. used biologically inspired features like Gabor wavelet to handle the problem of complex background [171] (Fig. 7).

– *Occlusion* Another major challenge is mitigating the effects of occlusion in gesture recognition. In single-handed gestures, the hand may occlude itself apart from some other objects. The problem is more severe in two-handed gestures where one hand may occlude the other while doing the gestures. The appearance of the hand is affected by both kinds of occlusion subsequently hampering recognition of gestures. In monocular vision-based gesture recognition, the appearance of gesturing hands is view dependent. As shown in Fig. 8, different hand poses appear to be similar in a particular view of observation due to self-occlusions. To solve occlusion problems there are some possible approaches:

- Use of multiple cameras for static/dynamic gestures.
- Use of tracking-based systems for dynamic gestures.

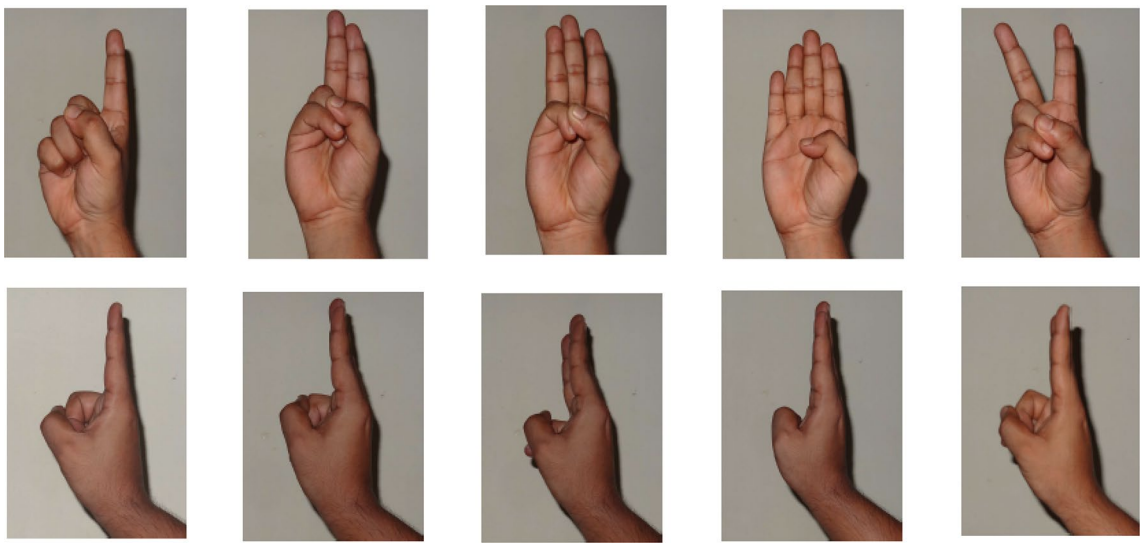

Fig. 8 Different hand poses and their side views

- Use of multiple cameras + tracking-based system for dynamic gestures.

– *Multiple camera-based gesture recognition* Utsumi et al. captured the hand with multiple cameras, selecting for gesture recognition the camera whose the principal axis is closest to normal to the palm. The hand rotation angle is then estimated using an elliptical model of the palm [218]. Alberola et al. used a pair of cameras to construct a 3D hand model with an occlusion analysis from the stereoscopic image [6]. In this model, a label is added to each of the joints, indicating its degree of visibility from a camera's viewpoint. The value of each joint's label range from fully visible to fully occluded. Ogawara et al. fitted a 26-DOF kinematic model to a volumetric model of the hand, constructed from images obtained using multiple infrared cameras arranged orthogonally [158]. Gupta et al. used occlusion maps to improve body pose estimations with multiple views [63].
– *Tracking-based gesture recognition* Lathuiliere et al. tracked the hand in real time by wearing a dark glove marked with colored cues [120]. The pose of the hand and the postures of the fingers were reconstructed using the position of the color markers in the image. Occlusion was handled by predicting the finger positions and by validating 3D geometric visibility conditions.
– *Multiple cameras with tracking-based gesture recognition* Instead of using multiple cameras and hand tracking separately, a fusion-based approach using both of them may be suitable for occlusion handling. Utsumi et al. used an asynchronous multi-camera tracking system for hand gesture recognition [219]. Though multiple camera-based systems are one solution for this problem, these devices are not purely accurate. View-invariant 3D models or depth measuring sensors can provide some more insight into this problem (Fig. 9).

- *Difficulties related to the articulated shape of the hand* The accurate detection and segmentation of the gesturing hand are significantly affected by variations in illumination and shadows, the presence of skin-colored objects in the background, occlusion, background complexity, and different other issues. The complex articulated shape of the hand makes it further tough to model the appearance of the hand for both static and dynamic gestures. Moreover, in the case of dynamic or trajectory-based gestures, the tracking of physical movement of the hand is quite challenging due to the varied size, shape and color of the hand. Generally, it is expected that a generic gesture recognition system should be invariant to the shape, size and appearance of the gesturing body part.

  The human hand has 27 bones—14 in the fingers, 5 in the palm, and 8 in the wrist (Fig. 10a). The 9 interphalangeal (IP) joints have one degree of freedom (DOF) each for flexion and extension. The 5 metacarpophalangeal (MCP) joints have 2 DOFs each: one for flexion and extension and the other for abduction or adduction (spreading the fingers) in the palm plane. The carpometacarpal (CMC) joint of the thumb, which is also called the trapeziometacarpal (TM), has 2 DOFs along nonorthogonal and nonintersecting rotation axes [74]. The palm is assumed to be rigid. Lee et al. proposed a 27-DOF hand

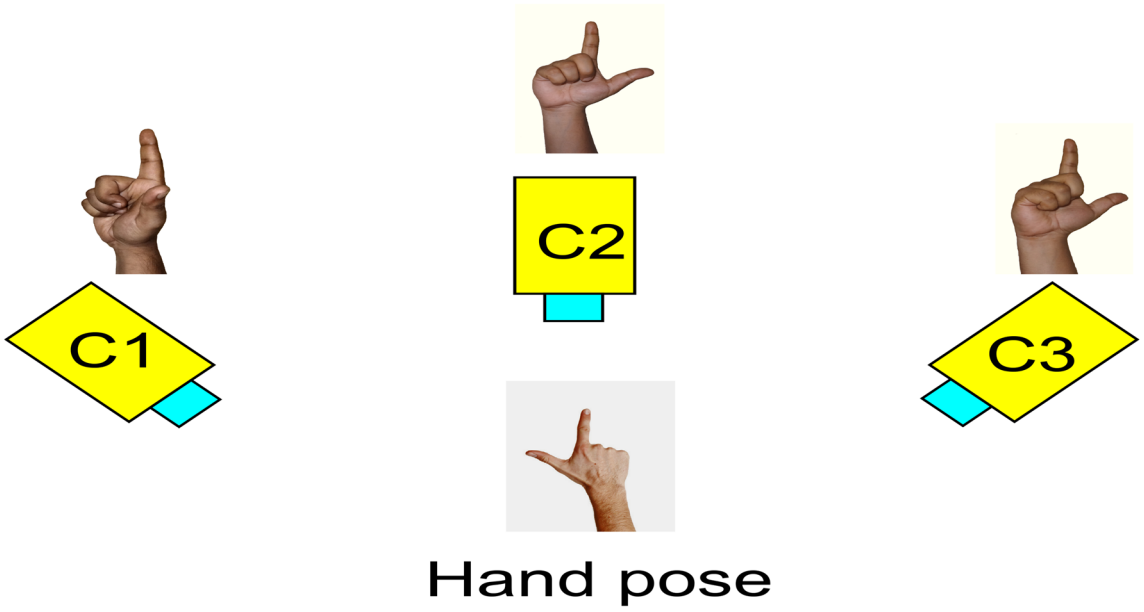

Fig. 9 Multiple camera-based gesture recognition

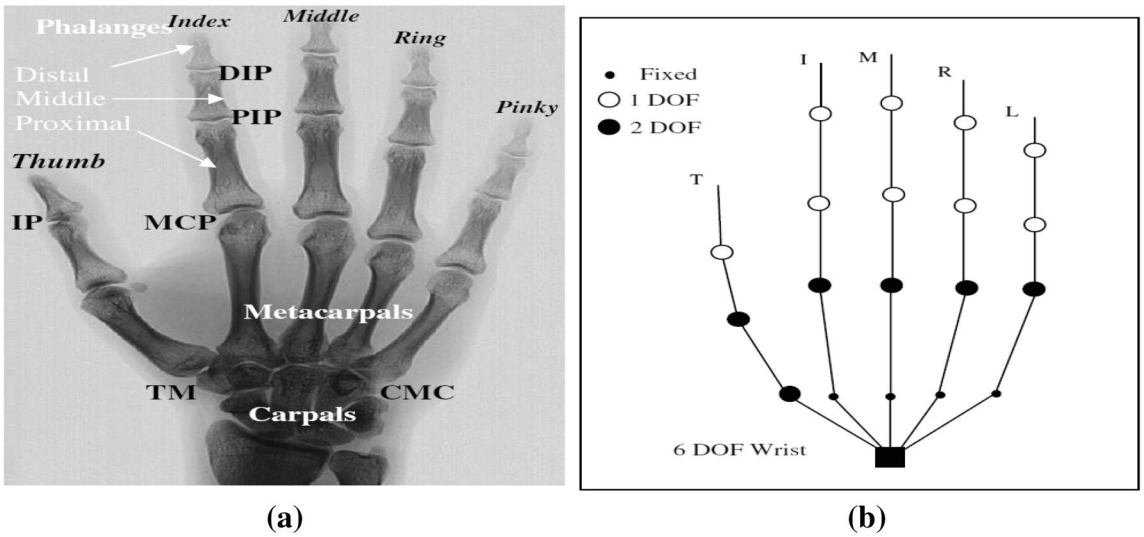

Fig. 10 Skeletal hand model: **a** hand anatomy [48], **b** the kinematic model [123]

model, assuming that the wrist has 6 DOFs (Fig. 10b). As evident from Fig. 10, the hand is an articulated object with more than 20 DOF. Now, because of the interdependencies between the fingers, the effective number of DOF reduces to approximately six. Their estimation—in addition to the location and orientation of the hand—results in a large number of parameters to be estimated. Estimation of hand configuration is extremely difficult because of occlusion and the high degrees of freedom. Even data gloves are not able to acquire the hand state perfectly. Compared with sensors for glove-based recognition, computer vision methods are generally at a disadvantage. To get rid of these constraints, [150] has tracked air-written gestures only through finger-tip detection. But it has the limitation that the detection of sign language is not possible. For monocular vision, it is impossible to know the full state of the hand unambiguously for all hand configurations, as several joints and finger parts may be hidden from the view of the camera. Applications in vision-based interfaces need to keep these limitations in mind and focus on gestures that do not require full hand pose information. General hand detection in uncon-

strained settings is a largely unsolved problem. In view of this, systems often locate and track hands in images using color segmentation, motion flow, background subtraction, or a combination of these techniques.

- *Gesture spotting problem* Gesture spotting means locating the beginning and the end-points of a gesture in a nonstop stream of gestures. When gesture boundaries are resolved, the gesture can be extracted and grouped. In any case, spotting significant patterns from a stream of gestures is an exceptionally troublesome errand mainly because of two issues: segmentation ambiguity and spatiotemporal variability. For sign language recognition, the framework should uphold the natural gesturing of the user to empower unhindered collaboration with the entity. Prior to taking care of the video into the recognition framework, the non-gestural movements ought to be taken out from the video sequence since these movements regularly blend a motion grouping. Instances of non-gestural movements incorporate "movement epenthesis" and "gesture co-articulation" (appeared in Fig. 11). Movement epenthesis occurs between two gestures and the current gesture is affected by the preceding or the following gesture. Gesture co-articulation is an unwanted movement that occurs in the middle of performing a gesture. In some cases, a gesture could be similar to a sub-part of a longer gesture, referred to as the "sub-gesture problem" [7]. When a user tries to repeat the same gesture, spatiotemporal variations in the shape and speed of the hands will occur. The system must accommodate these variations while maintaining an accurate representation of the gestures. Though static hand gesture recognition problem [52, 59, 60, 156, 174] is almost a solved one, but to date, there are only a handful of works are there dealing with these three problems of continuous hand gesture recognition system [16–18, 95, 133, 211, 240].
- *Problems related to two-handed gesture recognition* The inclusion of two-handed gestures in a gesture vocabulary can make HCI more natural and expressive for the user. It can greatly increase the size of the vocabulary because of the different combinations of left and right-hand gestures. Previously proposed methods include template-based gesture recognition with motion estimation [78] and two-hand tracking with colored gloves [10]. Despite its advantages, two-handed gesture recognition faces some major difficulties:
  - *Computational complexity* The inclusion of two-handed gestures can be computationally expensive because of their complicated nature.

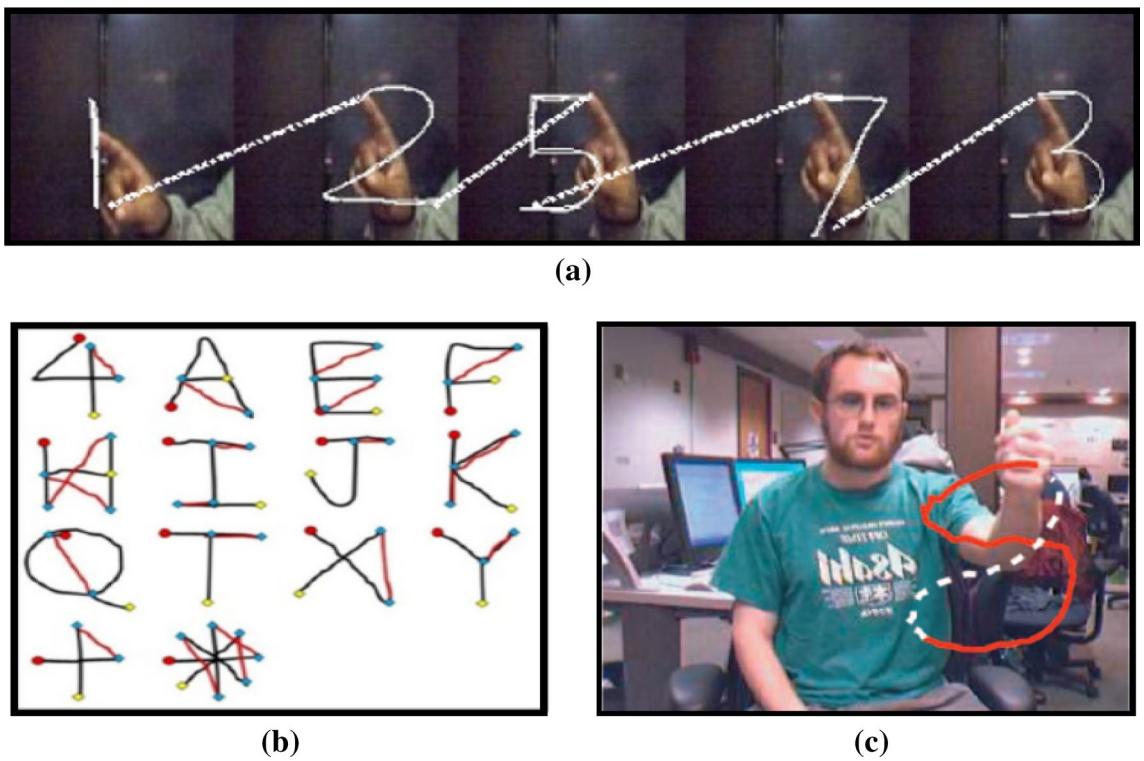

Fig. 11 **a** Movement epenthesis problem [18] **b** Gesture co-articulation (marked with redline) [202] **c** sub-gesture problem (here gesture '5' is a sub-gesture of gesture '8') [7]

  - *Inter-hand overlapping* The hands can overlap or occlude each other, thus impeding recognition of the gestures.
  - *Simultaneous tracking of both hands* The accurate tracking of two interacting hands in a real environment is still an unsolved problem. If the two hands are clearly separated, the problem can be solved as two instances of the single-hand tracking problem. However, if the hands interact with each other, it is no longer possible to use the same method to solve the problem because of overlapping hand surfaces [160].

- *Hand gestures with facial expressions* Incorporating facial expressions into the hand gesture vocabulary can make it more expressive as it can enhance the discrimination of different gestures with similar hand movements. A major application of hand and face gesture recognition is sign language. Little work has been reported in this research direction. Von Agris et al. used facial and hand gesture features to recognize sign language automatically [2].

  This approach also has the following challenges:

  - The simultaneous tracking of both hand and face.
  - Higher computational complexity compared with the recognition of only hand gestures.

- *Difficulties associated with extracted features* It is generally not recommended to consider all the image pixel values in a gesture video as the feature vector. This will not only be time-consuming but also it would take a great many examples to span the space variation, particularly if multiple viewing conditions and multiple users are considered. The standard approach is to compute some features from each image and concatenate these as a feature vector to the gesture model. Both the spatial and temporal movements of the hand along with its characteristics should be considered by a gesture model. No two samples of the same gesture will bring about the very same hand and arm movements or similar arrangement of visual pictures, i.e., gestures experience the ill effects of spatio-transient variety. Spatio-temporal variety exists in any event when the same user plays out a similar gesture on various occasions. Each time the user performs a gesture, the shape, position of the hand and speed of the motion normally change. Accordingly, extracted features ought to be rotation-scaling-translation (RST) invariant. Yet, different image processing strategies have their own imperatives to deliver RST-invariant features. Another limitation is that the processing of a lot of image information is tedious, and thus a real-time application might be troublesome.

Overview of Vision-Based Hand Gesture Recognition System

The essential part of vision-based frameworks is to identify and perceive visual signs for correspondence. A vision-based plan is more helpful than a glove-based one on account of its natural methodology. It tends to be utilized any place inside a camera's field of view and simple to convey. The fundamental undertaking of vision-based gesture recognition is to get visual data in a specific scene and attempt to separate the vital motions. This methodology should be acted in a progression of succession, in particular, acquisition, detection and pre-processing; gesture representation and feature extraction; and recognition (Fig. 12).

1. *Acquisition, detection and pre-processing* The acquisition and detection of the gesturing body part is vital for a productive VGR framework. The procurement incorporates capturing gestures utilizing imaging gadgets. The fundamental assignment of discovery and pre-processing is essentially the segmentation of the gesturing body part from images or videos as precisely as could really be expected.
2. *Gesture representation and feature extraction* The assignment of the following subsystem in a hand gesture recognition system is to model or represent the gesture. The performance of a gestural interface is directly related to the proper representation of hand gestures. After gesture modeling, a bunch of features should be extricated for gesture recognition. Diverse sorts of features have been distinguished for addressing specific sorts of gestures [25].
3. *Recognition* The last subsystem of an recognition framework has the assignment of recognition or classification

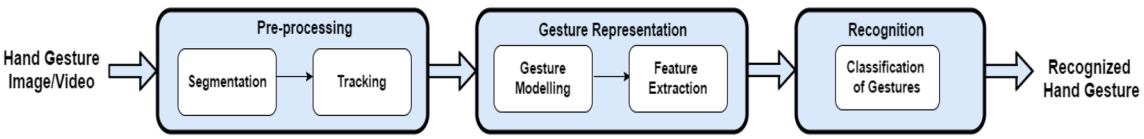

Fig. 12 The basic architecture of a typical gesture recognition system

of gestures. A reasonable classifier perceives the incoming gesture parameters or features and gathers them into either predefined classes (supervised) or by their closeness (unsupervised) [146]. There are numerous classifiers utilized for both static and dynamic gestures, every one with its own benefits and constraints.

Acquisition, Detection and Pre-processing

Gesture acquisition involves capturing images or videos using imaging gadgets. The detection and classification of moving objects present in a scene is key research in the field of action/gesture recognition. The most important research challenges are segmentation, detection, and tracking of moving objects from a video sequence. The detection and preprocessing stage mainly deals with localizing gesturing body parts in images or videos. Since dynamic gesture analysis consists of all these subtasks, so this very portion can be subdivided into segmentation and tracking or combining both of them together. Moreover, in static gestures also segmentation is a vital step.

1. *Segmentation* Segmentation is the way toward partitioning an image into various distinct parts and in this way discovering the region of interest (ROI), which is hand for our situation. Precise segmentation of the hand or the body parts from the captured images actually stays a challenge for some engrossed limitations in computer vision like illumination variation, background complexity, and occlusion. A large portion of the segmentation strategies can be extensively delegated as follows (Fig. 13): (a) skin color-based segmentation, (b) region based, (c) edge based, (d) Otsu thresholding and so on. The simplest method to recognize skin districts of a picture is through an explicit boundary specification for skin tone in a particular color space, e.g., RGB [69], HSV [205], YCbCr [28] or CMYK [193]. Numerous analysts drop the luminance segment and have utilized just the chrominance segment since chrominance signals contain skin color information. This is on the grounds that the hue-separation space is less sensitive to illumination changes when contrasted with RGB shading space [190]. Also, color cues show variations in the skin color in different illumination conditions, and also skin color changes with the change in human races, and so segmentation is more constrained in the presence of skin-colored objects in the background. Occlusion also leads to many issues in the segmentation process.

Recently published works of literature show that the performance of the model-based approaches (parametric and non-parametric) is better than explicit boundary specification-based methods [97]. To improve the detection accuracy, many researchers have used parametric and non-parametric model-based approaches for skin detection. For example, Yang et al. [237] used a single multivariate Gaussian to model skin color distribution. But, skin color distribution possesses multiple co-existing modes. So, the Gaussian mixture model (GMM) [238] is more appropriate than a single Gaussian function. Lee and Yoo [124] proposed an elliptical modeling-based approach for skin detection. The elliptical modeling has less computational complexity as compared to GMM modeling. However, many true skin pixels may get rejected if the ellipse is small. Whereas if the ellipse is larger, many non-skin pixels may be detected as skin pixels. Out of different non-parametric model-based approaches for skin detection, Bayes skin probability map (Bayes SPM) [88], self-organizing map (SOM) [22], k-means clustering [154], artificial neural network (ANN) [33], support vector machine (SVM)

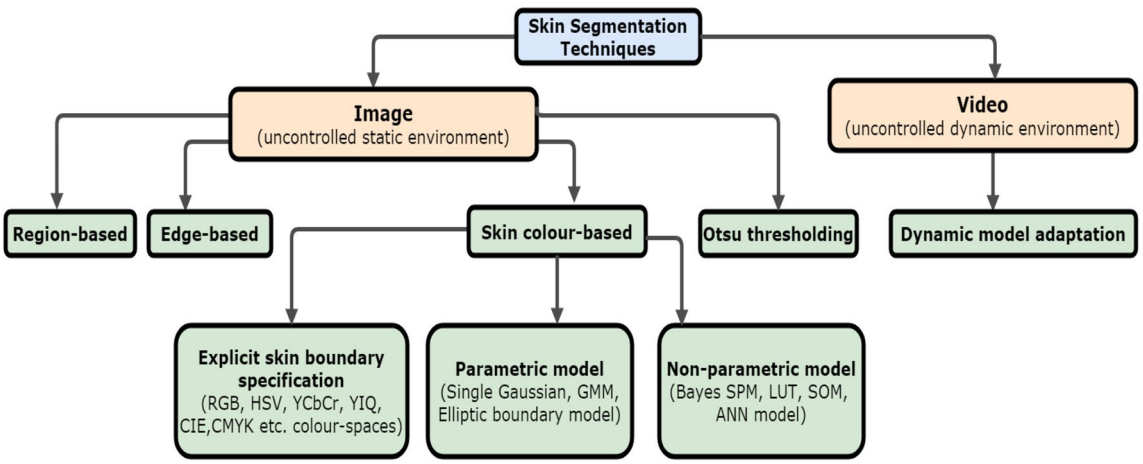

Fig. 13 Different skin segmentation techniques

[69], random forest [99] are noteworthy. The region-based approach involves region growing techniques, region splitting and region merging techniques. Rotem et al. [184] combined patch-based information with edge cues under a probabilistic framework. In an edge-based technique, basic edge-detecting approaches like Prewitt filter, Canny edge detector, Hough transforms are used. Otsu thresholding is a clustering-based image thresholding method that converts a gray-level image to a binary image using any edge detecting or tracking technique so that we have only two objects, i.e., one is hand and the other is background [145]. In the case of videos, all these methods can be applied with dynamic adaptation.

2. *Tracking* Tracking can also be considered as a part of pre-processing in the hand detection process as both tracking and segmentation together help to extract the hand from the background. Despite the fact that skin segmentation is perhaps the most favored technique for segmentation or detection, still, it is not so viable for different imperatives like scene illumination variation, background complexity, and occlusion [190]. Fundamentally, when earlier information on moving objects like appearance and shape is not known, pixel-level change can, in any case, give viable motion-based cues for detecting and localizing objects. Different methodologies for moving item discovery utilizing pixel-level change can be background subtraction, inter-frame difference, or three-frame difference [241]. Stabilized background detection consistently is an expensive matter making it defenseless for long and fluctuated video groupings [241]. Aside from this, the choice of temporal distance between frames is a tricky question. It essentially relies upon the size and speed of the moving object. Despite the fact that interframe difference methods can easily detect motion, it shows terrible performance in localizing the object. The three-frame difference [92] approach uses previous, current and future frames to localize the object in the current frame. The utilization of future frames presents a slack in the global positioning framework, and this slack is adequate just if the object is far away from the camera or moves slowly comparative with the high catch pace of the camera.

   Tracking of the hand can be restricted due to the fast movement of the hand and its appearance can alter immensely within a few frames. In such cases, model-based algorithms like mean-shift [56], Kalman filter [44], particle filter [30] are some of the methods used for tracking. The mean-shift is a purely non-parametric mode-seeking algorithm that iteratively shifts a data point to the average of data points in its neighborhood (similar to clustering). However, tracking often converges to an incorrect object when the object changes its position very quickly in the two neighboring frames. Because of this problem, a conventional mean-shift tracker fails to position a fast-moving object. [152, 185, 190] used a modified mean-shift algorithm called continuous adaptive mean-shift (CAMShift) where the window size is adjusted so as to fit the gesture area reflected by any variation in the distance between the camera and the hand. Though CAMShift performs well with objects that have a simple and consistent appearance, it is not powerful in more perplexing scenes. The movement model for the Kalman filter depends on the understanding that the speed is moderately little when items are moving, and thus, it is demonstrated by a zero mean and low variance white noise. One restriction of the Kalman filter is the supposition that the state variables depend on Gaussian distribution, and along these lines, the Kalman filter will give inaccurate assessments for state variables that do not follow a linear Gaussian environment. The particle filter is for the most part a preferred strategy over the Kalman filter since it can consider non-linearity and non-Gaussianity. The fundamental thought of the particle filter is to apply a weighted sample particle set to approximate the probability distribution, i.e., the necessary posterior density function is addressed by a bunch of arbitrary examples with related weights and estimation is done based on these samples and weights. Both Kalman filter and particle filter have the disadvantage of the requirement of previous knowledge in modeling the system. Kalman filter or particle filter can be combined with the mean shift tracker for precise tracking. In [224], authors have detected hand movement using Adaboost with the histogram of gradient (HOG) method.
3. *Combined segmentation and tracking* Here the first step is object labeling by segmentation and the second step is object tracking. Accordingly, an update for tracking is done by calculating the distribution model with various label values. Skin-segmentation and tracking together can give quite a good performance [68], but researchers have adopted other methods too where skin segmentation is not so efficient.

Gesture Representation and Feature Extraction

Based on spatio-temporal variation, gestures are mainly classified as static or dynamic. Static gestures are simply the pose or orientation of the gesturing part (e.g., hand pose) in the space and hence sometimes simply called posture. On the other hand, dynamic gestures are defined by trajectory or temporal deformation (e.g., shape, position, motion, etc.) of body parts. Again dynamic gestures can be either single isolated trajectory type or continuous type, occurring in a stream, one after another.

1. *Gesture representation* A gesture must be represented using a suitable model for its recognition. Based on feature extraction methods, the following are the types of gesture representations: model based and appearance based (Fig. 14).

   (a) *Model based* Here, gestures can be modeled utilizing either a 2D model or a 3D model. The 2D model essentially relies upon either different color-based models like RGB, HSV, YCbCr, and so forth, or silhouettes or contours obtained from 2D images. The deformable Gabarit model relies upon the arrangement of active deformable shaping. Then again, 3D models can be classified into mesh model [98], geometric model, volumetric models and skeletal models [198]. The volumetric model addresses hand motions with high exactness. The skeletal model diminishes the hand signals into a bunch of identical joint angle parameters with fragment length. For instance, Rehg and Kanade [179] utilized a 27-level degree-of-freedom (DOF) model of the human hand in their framework called 'Digiteyes'. Local image-based trackers are utilized to adjust the extended model lines to the finger edges against a solid background. Crafted by Goncalves et al. [61] advanced three-dimensional tracking of the human arm utilizing a two cone arm model and a single camera in a uniform background. One significant drawback of model-based portrayal utilizing a single camera is self-occlusion [61] that often happens in articulated objects like a hand. To stay away from it, a few frameworks utilize multiple/stereo cameras and restrict the motion to small regions [179]. But it also has its own disadvantages like precision, accuracy, etc. [32].

   (b) *Appearance based* The appearance-based model attempts to distinguish gestures either straightforwardly from visual images/videos or from the features derived from the raw data. Highlights of such models might be either the image sequences or a few features obtained from the images which can be utilized for hand-tracking or classification purposes. For instance, Wilson and Bobick [228] introduced results utilizing activities, generally hand motions, where the genuine gray-scale images (with no background) are utilized in real-life portrayal. Rather than utilizing raw gray-scale images, Yamato et al. [234] utilized body silhouettes, and Akita [5] utilized body shapes/edges. Yamato et al. [234] used low-level silhouettes of human activities in a hidden Markov model (HMM) system, where binary silhouettes of background-subtracted images are vector quantized and used as input to the HMMs. In Akita's work [5], the utilization of edges and some straightforward two-dimensional body setup information were utilized to decide the body parts in a progressive way (first, discover legs, then the head, arms, trunk) in light of steadiness. While utilizing two or three-dimensional primary data, there is a prerequisite of individual features or properties to be extracted and tracked from each frame of the video sequence. Consequently, movement understanding is truly cultivated by perceiving an arrangement of static setups that require previous detection and segmentation of the item. Furthermore, since the good old days, sequential state-space models like generative hidden Markov models (HMMs) [122] or discriminative conditional random fields (CRFs) [19] have been proposed to demonstrate elements of activity/gesture recordings. Temporal ordering models like dynamic time warping (DTW) [7] have likewise been applied with regards to dynamic activity/gesture recog-

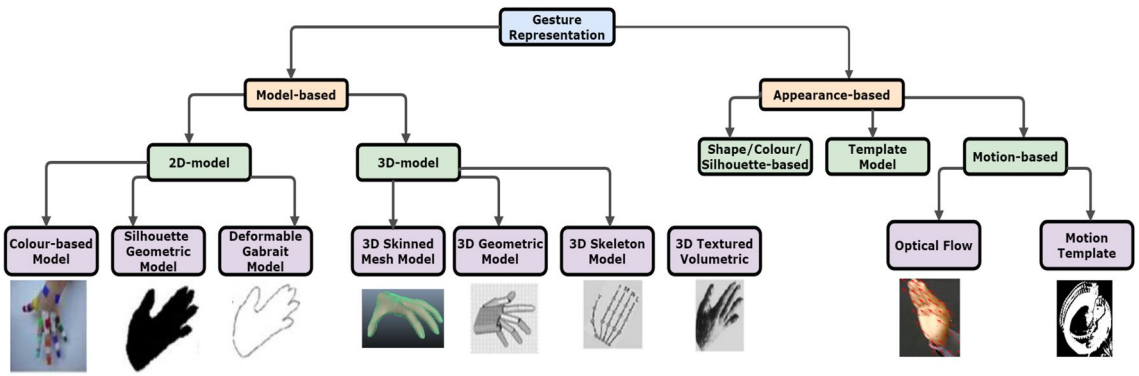

Fig. 14 Different hand models for hand gesture representation

nition where matching of an incoming gesture is done to a set of pre-defined representations.

In most literature, e.g., [165], it is mentioned that gestures are represented by either model-based or appearance-based model. The motion-based methods are also generally included in the appearance-based methods (shown in Fig. 14). But, here, we want to discuss the motion-based methods separately. This is because the shape and appearance of the body/body-part depend on many factors, e.g., illumination variation, image resolution, skin color, clothing, etc. But motion estimation should be independent of the shape and appearance of the gesturing hand (at least in theory). Optical flow and motion templates are the two major motion-based representation schemes and can be used directly to describe human gesture/action [191]. There are also a few examples like [191, 192, 232] where these two methods are combined together.

(a) *O*ptical flow Optical flow is the apparent movement or displacement of items/pixels as seen by a spectator. Optical flow shows the adjustment in speed of a point moving in the scene, likewise called a movement field. Here the objective is to assess the motion field (velocity vector) which can be figured from horizontal and vertical flow fields. Preferably, the motion field addresses the 3*D* movement of the points of an article across 2*D* image frame for a specific frame interval. Out of various optical stream procedures found in the literature, the most well-known strategies are: (a) Lucas–Kanade [134], (b) Horn–Schunk [76], (c) Brox 04 [23] and (5) Brox 11 [24], and (d) Farneback [51]. The choice of the optical flow technique principally relies upon the power of generating a histogram of optical flow (HOF) or motion boundary histogram (MBH) descriptor. HOF gives the optical flow vectors in horizontal and vertical directions. The natural thought of MBH is to address the oriented gradients computed over the vertical and horizontal optical flow components. When horizontal and vertical optical flow segments are acquired, histograms of oriented gradients are computed on each image component. The result of this interaction is a couple of horizontal (MBHx) and vertical (MBHy) descriptors. Laptev et al. [119] executed a blend of HOG-HOF for taking insensible human activity from motion pictures. [39] additionally proposed to ascertain changes of optical flow that focus on optical flow differences between frames (motion boundaries). Yacoob and Davis [233] utilized optical flow estimations to follow predefined polygonal patches set on interest areas for facial expression recognition. [229] introduced an incorporated methodology where the optical flow is coordinated frame-by-frame over time by considering the consistency of direction. In [135], the optical flow was used to detect the direction of motion along with the RANSAC algorithm which in turn helped to further localize the motion points. In [95], authors have used optical flow guided trajectory images for dynamic hand gesture recognition using deep learning-based classifier.

(b) *M*otion templates Basically, motion templates are the compact representation of a gesture video where the dynamics of motion of a gesture video is encoded into an image. These templates are compact representations of videos where a single image illustrates the motion information of the whole video useful for video analysis. Hence, these images are named motion fused images or temporal templates or motion templates. There are three widely used motion fusion strategies namely motion energy image(MEI) and motion history image (MHI) [3, 21], dynamic images (DI) [20] and methods based on PCA [49]. We will not go into the details of these methods, but the same can be found in [191] by the same authors.

2. *Feature extraction* After modeling a gesture, the next step is to extract a bunch of features for gesture recognition. For static gestures, features are obtained from image data like color and texture or posture data like direction, orientation, shape, and so forth. There are three basic features for spatio-temporal patterns of dynamic gestures namely location, orientation and velocity [242], based on which different features or descriptors are utilized in the cutting edge techniques. For instance, a few features depend on movement and additionally disfigurement data like position, skewness, and the speed of hands. Features for dynamic hand signals are spatial-transient examples. A static hand gesture might be seen as a special instance of a dynamic gesture with no temporal variation of the hand position as well as shape. A gesture model ought to think about both spatial and temporal changes of the hand and its motion. Generally, no two examples of the same gesture will bring about the very same hand and arm movements or generate a similar arrangement of visual information, i.e., motions experience the ill effects of spatial-transient variety. There exists spatial-transient variety when a user plays out the same gesture on various occasions. Each time the user plays out a motion, the shape and the speed of the motion for the most part shift. Regardless of

whether a similar individual attempt to play out a similar sign twice, a little variety in speed and position of the hands may happen. Subsequently, separated features ought to be rotation–scaling–translation (RST) invariant. Different features or descriptors are utilized in the classification stage of VGR frameworks. These features can be comprehensively classified depending on their technique for extraction, for example, spatial domain features, transform domain features, curve fitting-based features, histogram-based descriptors, and interest point-based descriptors. Also, the classifier ought to have the capacity to deal with spatio-temporal variations. As of late, feature extraction procedures based on deep learning have frequently been applied for various applications. Kong et al. [109] proposed a view-invariant feature extraction technique utilizing deep learning for multi-view activity acknowledgment. Table 1 gives a short review of the properties of various features utilized for both static and dynamic motion acknowledgment.

Recognition

The last subsystem of a gesture framework has the assignment of recognition where a reasonable classifier perceives the incoming gesture parameters or features and gathers them into either predefined classes (supervised) or by their closeness (unsupervised). Here, the hand gesture recognition techniques have been tried to classify into some categories for easy understanding. And based on the type of input data and the method, the hand gesture recognition process can be broadly categorized into three sections:

- Conventional methods on RGB data
- Depth-based methods on RGB-D data
- Deep networks—a new era in computer vision

Conventional Methods on RGB Data

Vision-based gesture recognition generally depends on three stages where the third module consists of a classifier, which typically classifies the input gestures. However, each classifier has its own advantages as well as limitations. Here, we discuss the conventional methods of classification for static and dynamic gestures on RGB data.

- *Static gesture recognition* Static gestures are basically finger-spelled signs in still images without any time frame. Unsupervised *k*-means and supervised *k*-NN, SVM, ANN are the major classifiers for static gesture recognition.
  - *k-means* It is an unsupervised classifier that evaluates *k* center points to minimize error in the clustering defined by the sum of the distances of all data points to their respective cluster centers. For a set of observations $(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n)$, in a *d*-dimensional real vector space, *k*-means clustering partitions the *n* observations into a set of *k* clusters or groups $S = \{S_1, S_2, \ldots, S_k\}$ $(k \leq n)$ and their centers are given by

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \left\| \mathbf{x}_j - \boldsymbol{\mu}_i \right\|^2. \quad (1)$$

The classifier arbitrarily finds *k* cluster centers in the feature space. Each point in the information dataset is assigned to the closest cluster center, and their locations are refreshed to the average location value for each group. This cycle is then rehashed until a halting condition is met. The halting condition could be either a user indicated of maximum number of cycles or a distance edge for the development of the group communities. Ghosh and Ari [59] utilized a *k* means clustering-based radial basis function neural network (RBFNN) for static hand gesture recognition. In this work, *k* means grouping is utilized to decide the RBFNN centers.
  - *k-nearest neighbors (k-NN)* *k*-NN is a non-parametric algorithm where information in the feature space can be multidimensional. It is a supervised learning scheme with a bunch of labeled vectors as training data. The number *k* essentially decides the number of neighbors (close feature vectors) that impact the characterization. Commonly, an odd estimation of *k* is picked for two-class characterization. Each neighbor might be given a similar weight or more weight might be given to those nearest to the input information by applying a Gaussian distribution. In uniform voting, a new feature vector is allocated to the class to which the majority of its neighbors belongs. Hall et al. expected two statistical distributions (Poisson and binomial) for the sample data to get the ideal estimation of *k* [67]. The *k*-NN can be utilized in various applications, for example, hand gesture-based media player control [138], sign language recognition [64], and so on.
  - *Support vector machine (SVM)* An SVM is a supervised classifier for both linearly separable and non-separable data. When it is not possible to linearly separate the input data in the current feature space, then SVM maps this non-linear data to some higher dimensional space where the data can be linearly separated. This mapping from lower to higher dimensional space makes the order of the information more straightforward and recognition more precise. On several occasions SVM has been utilized for

Table 1 Major features used in gesture recognition

Feature type	Examples	Static	Dynamic	Advantages	Limitations
Spatial domain (2D)	Fingertips location, finger direction, and silhouette [156]	✓	✓	• Easy to extract	• Unreliable under occlusion or varying illumination
				• Rotation invariant	• Object view dependent
				• Distorted hand trajectory distorts MCC also	
	Motion chain code (MCC) [19, 122]		✓		
Spatial domain (3D)	Joint angles, hand location, surface texture and surface illumination [118]	✓	✓	• 3D modeling can most accurately represent the state of a hand, and thus can give higher recognition accuracy	• Difficult to accurately estimate 3D shape information of a hand
Transform domain	Fourier descriptor [70], DCT descriptor [4], Wavelet descriptor [79]	✓	✓	• RST invariant	• Not able to perfectly distinguish different gestures
Moments	Geometric moments, orthogonal moments [174]	✓	✓	• Moments can be used to derive RST invariant global features	• Moments are in general global features. So, moments cannot effectively represent an occluded hand
Curve fitting based	Curvature scale space [231]		✓	• RST invariant	• Sensitive to distortion in the boundary
				• Resistant to noise	
Histogram based	Histogram of gradient (HoG) features [52]	✓	✓	• Invariant to geometry and illumination changes	• Performance is not so satisfactory for images with a complex background and noise
Interest point based	Scale-invariant feature transform (SIFT) [40], Speeded up robust features (SURF) [247]	✓	✓	• RST and illumination invariant	• They are not the best choice for real-time applications because they are computationally expensive
Mixture of features	Combined features [60]	✓	✓	• Incorporates the advantages of different types of features	• Classification performance may degrade due to curse of dimensionality

hand gesture recognition [41, 98, 132, 183]. SVMs were initially intended for two-class grouping, and an expansion for multi-class arrangement is vital for many instances. Dardas et al. [41] applied SVM along with bag-of-visual-words for hand gesture recognition. Weston and Watkins [226] proposed an SVM design to settle a multi-class pattern recognition problem using a single optimization stage. Be that as it may, their optimization procedure found to be extremely convoluted to be executed for real-life pattern recognition problems [77]. Rather than utilizing a single optimization method, various paired classifiers can be utilized to take care of multi-class grouping issues, for example, "one-against-all" and "one-against-one" techniques. Murugeswari and Veluchamy [151] utilized "one-against-one" multi-class SVM for gesture recognition. It was tracked down that the "one-against-one" strategy performs better compared to the remainder of the strategies [77].

– *Artificial neural network (ANN)* ANN is a statistical learning algorithm utilized for different errands like functional approximation, pattern recognition and classification. ANNs can be used as a biologically inspired supervised classifier for gesture recognition where training is performed utilizing a bunch of marked input data. The trained ANN arranges new input data into the labeled classes. ANNs can be utilized to perceive both static [59] as well as dynamic hand gestures [157, 163]. [157] applied ANN to classify gesture motions utilizing a 3D articulated hand model. A dataset collected using Kinect® sensor [163] was used for this. Obtaining info from data glove, Kim et al. [102] applied ANNs to perceive Korean sign language from the movement of hand and fingers. A restriction of traditional ANN design is its failure to deal with temporal arrangements of features proficiently and successfully [165]. Primarily, it cannot make up for changes in transient moves and scales, particularly in real-time applications [177]. Out of a few altered structures, multi-state time-delay neural networks [239] can deal with such changes somewhat utilizing dynamic programming. Fuzzy-based neural networks have likewise been utilized to perceive gestures [220].

- *Dynamic gesture recognition* Dynamic gestures or trajectory-based gestures are gestures having trajectories with temporal information in terms of video frames. Dynamic gestures can be either a single isolated trajectory type or continuous type occurring one after another in a stream. Recognition performance of dynamic gestures, especially the continuous gestures, is basically dependent on gesture spotting schemes. Dynamic gesture recognition schemes can be categorized into direct or indirect methods [7]. The approaches in direct method first detect the boundaries in time for the performed gestures and then apply standard techniques same as isolated gesture recognition. Typically, motion cues like speed, acceleration and trajectory curvature [242]) or specific starting–ending marks [7], an open/closed palm can be applied for boundary detection. Whereas, in the indirect approach temporal segmentation is intertwined with recognition. In indirect methods, typically gesture boundaries are detected by finding time intervals that give good scores when matched with one of the gesture classes in the input sequence. Such procedures are too vulnerable to false positives and recognition errors as they have to deal with two vital constraints of dynamic gesture recognition [146]: 1) spatiotemporal variability, i.e., a user cannot reproduce the same gesture at the exact same shape and duration and 2) segmentation ambiguity, i.e., problems faced due to erroneous boundary detection. Through indirect methods, we try to minimize these problems as much as possible. Indirect methods can be of two types (Fig. 15): non-probabilistic, i.e., (a) dynamic programming/dynamic time warping, (b) ANN; and probabilistic, i.e., (c) HMM and other statistical methods, (d) CRF and

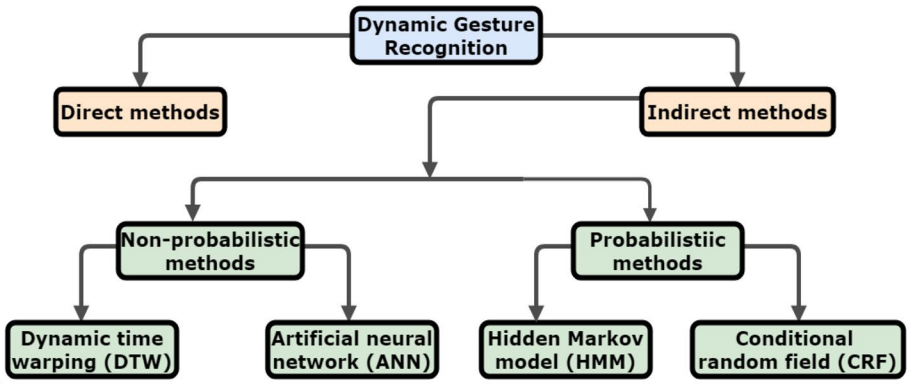

Fig. 15 Conventional dynamic gesture recognition techniques

its variants. Some other common techniques are eigenspace-based methods [164], curve fitting [196], finite-state machine (FSM) [16, 19] and graph-based methods [194].

– *Dynamic programming/dynamic time warping (DTW)* A template matching approach of dynamic programming is dynamic time warping (DTW) and it has been extensively used in isolated gesture recognition. It can find the optimal alignment of two signals in the time domain. Each element in a time series is represented by a feature vector. So, the DTW algorithm calculates the distance between each possible pair of points in two time series in terms of their feature vectors. The steps in a DTW are as follows:

- Two time series P and Q:

$$\mathbf{P} = \mathbf{p_1}, \mathbf{p_2}, \cdots, \mathbf{p_M}$$
$$\mathbf{Q} = \mathbf{q_1}, \mathbf{q_2}, \cdots, \mathbf{q_N},$$

  where $\mathbf{q}_i$, $\mathbf{p}_i$ are feature vectors for the ith element of the corresponding time sequences.
- Construct $N \times M$ matrix D with distances $D_{ij} = d(\mathbf{p}_i, \mathbf{q}_j)$.
- Warping path W is a contiguous set of matrix elements $w_k = (i,j)_k$
- Define warping between **P** and **Q**

$$W = w_1, w_2, ..., w_K,$$

  where $\max(M,N) \leq K \leq M + N - 1$
- Find:

$$DTW(P,Q) = \min \sqrt{\sum w_k}.$$

DTW has been applied for gesture classification by several authors [7, 80, 127, 211]. Alon et al. [7] proposed a DTW-based approach that can handle the sub-gesture problem. Lichtenauer et al. [127] introduced a hybrid method by applying statistical DTW (SDTW) only for time warping and another classifier on the warped features.

– *Hidden Markov model (HMM)* Though HMM originally emerged in the field of speech recognition, now, it is one of the most widely used techniques for gesture recognition with its numerous variants. HMM is extensively used because it can be applied for modeling the spatiotemporal variability of the gesture videos. Since trajectory-based gesture is a series of images, so there is a need for past knowledge to help the system to recognize gestures and an HMM can help us in this. Before we elaborate on HMM, let us understand a traditional Markov process. A stochastic process has the nth order Markov property if the current event's conditional probability density is dependent only on the n most recent events. For $n = 1$, the process is called a first-order Markov process, where the current event depends only on the previous event. This is a useful assumption for hand gestures, where the positions and orientations of the hands are treated as events. HMM has two special properties for encoding hand gestures—a) it assumes a first-order model, i.e., it encodes the present time (t) in terms of the previous time ($t-1$)—the Markov property of underlying unobservable finite-state Markov process and b) a set of random functions, each associated with a state, that produces an observable output at discrete intervals. In this way, an HMM is a "doubly stochastic" process [176]. The states in the hidden stochastic layers are governed by a set of probabilities:

i. The state transition probability distribution **A**, which gives the probability of transition from the current state to the next possible state.
ii. The observation symbol probability distribution **B**, which gives the probability of observation for the present state of the model.
iii. The initial state distribution **Π**, which gives the probability of a state being an initial state.

An HMM is expressed as $\boldsymbol{\lambda} = (\mathbf{A}, \mathbf{B}, \boldsymbol{\Pi})$ and is described as follows:

- Let there be a set of N states $\{s_1, \ldots, s_N\}$; with a sequence of states $Q = \{q_1, \ldots, q_T\}$, where $t = 1, ..., T$. For a gesture with M observable states, the set of observed symbol or feature is given by $O = \{o_1, \ldots, o_T\}$.
- The state-transition matrix is $\mathbf{A} = \{a_{ij}\}$, where a_{ij} is the state-transition probability from state $q_t = s_i$ at time t to state $q_t = s_j$ at time $t+1$.

$$\mathbf{A} = \{a_{ij}\} = P(q_{t+1} = s_j | q_t = s_i), \quad \text{for } 1 \leq i, j \leq N.$$

- The observation symbol probability matrix $\mathbf{B} = \{b_{jk}\}$, where b_{jk} is the probability of symbol o_k at state s_j.

$$b_j(k) = P[o_k \text{ at t} | q_t = s_j], \quad \text{for } 1 \leq j \leq N, 1 \leq k \leq M.$$

- The initial probability distribution $\Pi = \{\pi_j\}$, where

$$\pi_j = P\left[q_1 = s_j\right], \quad \text{for} \quad 1 \leq j \leq N.$$

The modeling of a gesture involves two phases—feature extraction and HMM training. In the first phase, a particular gesture sequence is represented by a set of feature vectors. Each of these feature vectors describes the trajectory of the hand corresponding to a particular state of the gesture. The number of such states depends on the nature and complexity of a gesture. In the second phase, the vector set is used as an input to HMM. The global HMM structure is formed by connecting in parallel the trained HMMs $(\lambda_1, \lambda_2, \ldots, \lambda_G)$, where G is number of gestures to be recognized. For dynamic gestures, temporal components like the start state, the end state, and the set of observation sequences (e.g., position) are mapped by an HMM classifier using a set of boundary conditions.

For a given observation sequence, the key issues of HMM are,

- *Evaluation* Given the model $\lambda = (\mathbf{A}, \mathbf{B}, \mathbf{\Pi})$. What is the probability of occurrence of a particular observation sequence $O = \{o_1, \ldots, o_T\} = P(O|\lambda)$? This is the heart of the classification/recognition problem. Determination of the probability that a particular model will generate the observed sequence when there is a trained model for each of a set of classes (forward–backward algorithm).
- *Decoding* Optimal state sequence to produce an observation sequence $O = \{o_1, \ldots, o_T\}$ Determination of the optimal state sequence that produces the observation sequence (Viterbi algorithm).
- *Learning* Determine model λ, given a training set of observations, i.e., find λ, such that $P(O|\lambda)$ is maximal. Train and adjust the model to maximize the observation sequence probability such that HMM should identify a similar observation sequence in the future (Baum–Welch algorithm).

HMMs are frequently applied for trajectory-based gesture recognition [72, 117, 122, 166]. But the main disadvantage of HMM is that every gesture model has to be represented and trained separately considering it as a new class, independent of anything else already learned.

– *Conditional random field (CRF)* CRF is basically a variant of the Markov model with some added advantages. HMM requires strict independence assumptions across multivariate features and conditional independence between observations. This is generally violated in continuous gestures where observations are not only dependent on the state, but also on the past observations. Another disadvantage of using HMM is that the estimation of the observation parameters needs a huge amount of training data. The distinction between HMM and CRF is that HMM is a generative model that defines a joint probability distribution to solve a conditional problem thus focusing on modeling the observation to compute the conditional probability. Moreover, one HMM is constructed per label or pattern where HMM assumes that all the observations are independent. On the other hand, CRF is a discriminative model that uses a single model of the joint probability of the label sequence to find conditional densities from the given observation sequence. CRFs can effortlessly address contextual dependencies and have computationally alluring properties. CRFs support proficient recognition utilizing dynamic programming, and their parameters can be learned utilizing convex optimization.

Both HMM and CRF can be used for labeling sequential data. For this, we define a statement for a given observation sequence x that, we want to choose a label sequence y^* such that the conditional probability $P(y|x)$ is maximized, that is:

$$y^* = argmax_y P(y|x). \tag{2}$$

Maximum entropy Markov models (MEMMs) are discriminative models, where each state has an exponential model that takes the observation sequence as input and outputs a probability distribution over the next possible states.

$$P(y|x) = \prod_{t=1}^{T} P(y_t|y_{t-1}, x). \tag{3}$$

Each of the $P(y_t|y_{t-1}, x)$, is an exponential model of the form:

$$P(y|x) = \frac{1}{Z(x_t, y_{t-1})} \exp\left(\sum_a \lambda_a f_a(x_t, y_t)\right), \tag{4}$$

where Z is a normalization constant and the summation is overall features. But MEMM suffers from Label Bias Problem, i.e., the transition probabilities of leaving a given state are normalized for only that state (local normalization). MEMMs have a non-linear decision surface in light of the fact that the current observation is simply ready to choose what successor state has chosen; however, the probability mass do not move to that state. To stay away from this impact, a CRF utilizes an undirected graphical model that characterizes a single

log-linear distribution over the joint vector of a whole class label sequence given a specific observation sequence and accordingly the model has a linear decision surface. Let $G = (V, E)$ be a graph such that $Y = (Y_v), v \epsilon V$ so that Y is indexed by vertices of G. Then (X, Y) is a conditional random field, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph: $p(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$, where $w \sim v$ means that w and v are neighbors in G. Given by Hammersley and Clifford, it states that the probability distribution of x satisfies the Markov property with respect to graph $G(V, E)$ if and only if, it can be factored according to G:

$$P(x) = \frac{1}{Z}\prod_C \psi_C, \tag{5}$$

where Z is the normalization constant and ψ_C is the potential function over clique C.

$$P(x) = \frac{1}{Z}\prod_C \exp(\lambda_C^T f(C)) = \frac{1}{Z}\exp(\sum_C \lambda_C^T f(C)), \tag{6}$$

where $f(.)$ is the feature vector defined over the clique and λ is the corresponding weight vector for those features. Bhuyan et al. [19] proposed a recognition method applying CRF through a novel set of motion chain code features. Sminchisescu et al. [203] have compared performance analysis applying algorithms based on CRF and MEMM for discerning human motion in video sequences. Undirected conditional model CRF and directed conditional model MEMM with different windows of observations are compared with HMM. Both MEMM and HMM have trouble in perceiving long-range observation dependencies that become useful in discriminating among various gestures. It is seen that CRFs have better recognition performance compared to MEMMs, which in turn, typically outperformed traditional HMMs. This is because CRF applies an undirected graphical model to overcome the problem of label bias present in maximum entropy Markov models (MEMMs) where states with low-entropy transition distributions effectively ignore their observations. The main constraint of CRF is that training is more time-consuming ranging from several minutes to several hours for models having longer windows of observations (as compared to seconds for HMMs, or minutes for MEMMs), on a standard desktop PC.

– *Some other classification methods* Here, we discuss some other classification techniques that have also been used in the classification of gestures. Patwardhan and Roy [164] presented an eigenspace-based methodology to represent trajectory-based hand gestures containing both shape and trajectory information which are rotation, scale and translation (RST) invariant. Shin et al. [196] presented a curve-fitting based geometric framework utilizing Bezier curves by fitting the curve to the 3D motion trajectory of the hand. The gesture velocity is interlinked in the algorithm to enable trajectory analysis and classification of dynamic gestures having variations in velocity. Bhuyan et al. [16, 19] represented the keyframes of a gesture trajectory as a sequence of states ordered in the spatial-transient space, which constitutes a finite state machine (FSM) that classifies the input. Graph-based frameworks are also applied as a powerful scheme for pattern recognition problems but have been practically left unused for a long period of time due to their high computational cost. [194] used graphs for gestures matching in an eigenspace to handle hand occlusion (Fig. 16).

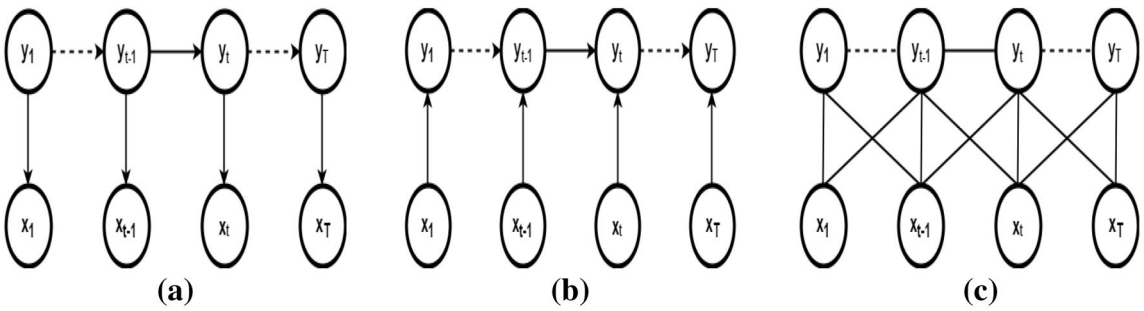

Fig. 16 **a** HMM **b** a directed conditional model or MEMM **c** a conditional random field accommodates arbitrary overlapping features or long-term dependency of observation sequence [203]

Depth-Based Methods on RGB-D Data

Depth information is largely invariant to illumination variation and skin colors and offers a quite clear segmentation from the background. So, the major problems in segmentation like illumination variation and occlusion can be handled nicely with the help of depth information to a great extent. Due to these advantages, depth measuring cameras have been used in the field of computer vision for many years. However, the applicability of depth cameras was restricted because of their excessive cost and low quality. With the introduction of low-cost color-depth (RGB-D) cameras like Kinect® by Microsoft, Leap Motion Controller (LMC) by Leap Motion, Intel RealSense®, Senz3D® by Creative and DVS128® by iniLabs, a new revolution was evolved in gesture recognition by providing high-quality depth images that can handle issues like complex background and variation in illumination. Out of all these, hand gesture recognition on Kinect®-based dataset and 'one-shot learning' with RGB-D data, are the prominent methods mostly discussed in depth-based hand gesture recognition.

- *Kinect®-based methods* Kinect® has a combined RGB and IR camera along with depth sensor [248]. It uses the infrared projector and sensor for depth computation and an RGB camera for capturing RGB data only. The infrared projector projects a predefined pattern on the items and a CMOS sensor captures the deformations in the reflected pattern. Depth information is then calculated by mapping a three-dimensional view of the scene obtained from the deformation information. Kinect® acquire RGB-D information by consolidating organized light with two exemplary computer vision strategies: depth from focus and depth from the stereo. The skeletal information got from these RGB-D sensors is changed over to more significant and undeniable features, and algorithms are created for the robust gesture classification. Classification of hand gestures is particularly difficult because of the complex articulation and relatively smaller area of the hand region. Kinect® is helpful in tending to these central issues in computer vision [163, 181, 212]. It has also diverse applications ranging from gaming to classroom [71, 116].
- *Other depth sensor-based methods* Leap motion controller (LMC) and Intel RealSense® are the most used RGB-D sensor for HCI applications apart from Kinect®. RealSense® is more robust to self-occlusions and it can capture pinching gestures. LMC is another RGB-D sensor and its purpose is to locate 3D fingertip positions instead of the whole-body depth information as the case with Kinect® sensor. It can detect only fingertips lying parallel to the sensor plane, but with high accuracy. In [133] feature vector with depth information is computed using a leap motion sensor and fed into the hidden conditional neural field (HCNF) to classify dynamic hand gestures. Leap motion sensors can also be applied in different utilization, e.g., virtual environments [178] and sign language recognition [172].
- *One-shot learning methods on RGB-D data* Using Deep Learning, human-level performance has become achieveable on complex image classification tasks. However, these models rely on a supervised training paradigm and their achievement is heavily dependent on the availability of labeled training data. Also, the classes that the models can recognize are limited to those they were trained on. This makes these models less useful in realistic scenarios for the classes where enough labeled data is not available during training. Also, since it is practically not possible to train on images of all possible objects, so the model is expected to recognize images from classes with a limited amount of data in the training phase or precisely with a single example. So, in the case of a small dataset, 'one-shot learning' may be very useful. Various researchers [108, 221, 230] have used one-shot learning in both deep learning and non-deep learning paradigm for recognition of hand gestures, especially with RGB-D data. Wu et al. [230] presented a framework to learn gestures from just one learning sample for each class, in particular, 'one-shot learning'. Features are obtained depending on extended motion history image (Extended MHI) and the gestures are recognized based on the maximum correlation coefficient. The extended MHI is used to improve the presentation of MHI by making up for the immobile regions and repeated activities. A multi-view spectral embedding (MSE) scheme is utilized to meld the RGB and depth information in an actually significant way. The MSE calculation finds the natural connection among RGB and depth features, improving the recognition rate of the algorithm. In [136], authors used a methodology consolidating MHI with statistical measures and frequency domain transformation on depth images for one-shot-learning hand gesture recognition. Due to the availability of the depth information, the background-subtracted silhouette images were obtained using a simple mask threshold.

Deep Networks: A New Era in Computer Vision

Though the idea of artificial intelligence (AI) is quite ancient, modern AI first came into the picture around the mid-twentieth century. The AI aims at developing intelligence in machines so as to make them work and respond like humans. This can be achieved when the machines are made to have certain traits, e.g., reasoning, problem solving, perception, learning, etc. Machine learning (ML) is one of the

cores of AI. There are a large number of applications of ML in many aspects of modern human society. Consumer products like cameras and smartphones are the best examples where ML techniques are being employed increasingly. In the field of computer vision, ML techniques have been vigorously used in different applications like object detection, image classification, face recognition, gesture and activity recognition, semantic segmentation, and many more. In conventional ML, engineers and data scientists have to identify useful features and they have to handcraft the feature extractor manually which requires considerable engineering skills and domain knowledge. To identify important and powerful features, they must have considerable domain expertise. The issue of "handcrafting features" can be addressed if good features can be learned automatically. This automatic learning of features can be done by a learning method called "representation learning". These are methods that enables a machine to automatically learn the representations that are crucial for detection or classification.

Recently, deep learning has shown outstanding performance outperforming "non-deep" state-of-the-art methods in action and gesture recognition fields. Deep learning, a subfield of ML, is based on representation learning methods having multiple levels of representation. Deep learning is a part of ML algorithms, in which extraction of multiple levels of features is possible. In several fields, such as computer vision, deep learning methods have been proved to have much better performance than conventional ML methods. The main reason for deep learning having an upper hand over ML is the fact that the feature learning mechanism at these different levels of representation is fully automatic, thereby allowing the computational model to implicitly capture intricate structures embedded in the data. The deep learning methods are said to have deep architecture because of the non-uniform processing of information at different levels of abstraction where higher-level features are interpreted in the form of lower-level features. This has propelled the advancement of learning powerful and successful portrayals straightforwardly from raw data and deep learning gives a conceivable method of naturally learning different levels of image specific features by utilizing different layers. Deep networks are fit for discovering remarkable dormant constructions inside unlabeled and unstructured raw data and can be utilized for both feature extraction as well as classification [110]. The recent popular deep learning methods like convolutional neural network (CNN), recurrent neural network (RNN) and long short-term memory (LSTM) have demonstrated competitive performance in both image/video representation as well as classification. But deep learning approaches have mainly two inherent requirements: huge data for training purposes and expensive computation. But in this modern era, the abundance of high quality, easily available labeled datasets from different sources along with parallel graphics processing unit (GPU) computing, also played a vital role in the success of deep learning by fulfilling its requirements. We will see all these methods one by one, but before that let's talk about one major problem of deep learning which is the requirement of huge data and how various researchers have tried to overcome it through the data augmentation process when the database is limited.

- *The need for data augmentation in deep learning methods* Contrary to the hand-crafted features, there is developing interest towards feature learned and represented by deep neural networks [12, 29, 37, 43, 58, 85, 94, 101, 110, 121, 129, 148, 149, 153, 169, 201, 215, 217, 223, 249, 250]. But the fundamental necessity in deep learning methods is loads of data set examples. Various researchers have stressed the significance of utilizing diverse training samples for CNNs/RNNs [110]. For datasets with restricted variety, they have proposed data augmentation techniques in the training stage to forestall CNNs/RNNs from overfitting. Krizhevsky et al. [110] utilized different data augmentation procedures in the preparation of the recognition problem of 1000 groups. Simonyan and Zisserman [201] utilized some spatial augmentation on every image frame to prepare CNNs for video-based human action classification. Notwithstanding, these data augmentation strategies were restricted to only spatial varieties. Pigou et al. [169] transiently deciphered video outlines apart from applying spatial changes to add varieties to the video sequences containing dynamic movement. Molchanov et al. [148] applied space-time video augmentation methods to keep away 3D-CNN from overfitting.
- *Convolutional neural networks (CNN)* In 1962, D.H. Hubel and T.N. Weisel proposed the prototype of Cat's visual cortex, which later on helped in the development of CNNs. The first neural network architecture for visual pattern recognition was presented by K. Fukushima in 1980 and was given the nickname "neocognitron" [57]. This network was based on unsupervised learning. Finally, in the late 90s, Yann LeCunn and his collaborators developed CNN which showed exciting results in various recognition tasks [121]. But till 2012, CNN was not that much evolved due to the requirements of deep learning methods mentioned above. After the work of Krizhevsky et al. [110], various researchers applied CNN in various domains for classification as well as other purposes. Generally, 2D-CNN is used in the case of images that can access only spatial information, whereas, for video processing, 3D-CNN (C3D) is quite effective which can extract both spatial as well as temporal information. A fusion-based approach with CNN as trajectory shape extractor of a gesture video and CRF as temporal feature extractor is proposed by [235]. In [190], the

authors used CNN for recognition of hand gestures using trajectory-to-contour-based images obtained through skin segmentation and tracking method. In [245], the authors used pseudo-color-based MHI images as input to convolutional networks. [96] proposed a model for isolated gesture recognition using optical flow where the trajectory-contour of the moving hand with varied shape, size and color is detected and the hand gesture is classified through a VGG16 CNN framework.

- *3D-CNN (C3D) model* 2D-CNN can handle 2D images for various tasks like recognition acting on the raw data directly. Whereas 3D-CNN models, also called C3D, act on videos for gesture or action detection. The framework obtains features from spatial as well as temporal dimensions by acting convolutions in 3D, thereby capturing the spatial as well as movement data present in the video sequence. [85] introduced a C3D network for human action recognition. To examine the progression of short video clips and normalize the framework's reactions for all the clips, Tran et al. [215] employed a C3D to learn the spatio-temporal features from sliced video clips and then fuse these features to make the final classification. [223] used a temporal segment network that works on video segments called snippets for spatio-temporal evaluation in action recognition. 3D-CNN (C3D) is quite effective which can extract both spatial as well as a piece of temporal information at less expense of both data and processing computation compared to RNN/LSTM [101, 192].
- *Two-stream model* Ciregan et al. [37] explained the advantage of utilizing multiple CNNs in parallel in improving the performance of the whole network by 30–80% for different image grouping errands. Also, for large-scale video arrangement, Karpathy et al. [94] found that the best outcomes can be obtained by joining two separate layers of CNNs trained with original and spatially trimmed video clips. Simonyan and Zisserman [201] proposed different streams of CNNs for spatial and transient data extraction which are later intertwined in the late-fusion scheme. Here in one stream optical flow is used for activity acknowledgment. To perceive sign language gestures, Neverova et al. [153] utilized CNNs to consolidate tone and depth information from hand areas and upper-body skeletons. Two stream model with two C3D layers that takes RGB and optical flow computed from the RGB stream as inputs were used by [101] for action recognition. [250] used a hidden two-stream CNN model where input is a crude video sequence that can explicitly detect the activity class without computing optical flow directly. Here the network predicts the motion information from consecutive frames through a temporal stream CNN that makes the network 10× faster [250], without computing optical flow which is time-consuming.
- *Long-term video prediction—RNN/LSTM/GRU* CNN can handle restricted local temporal data, and consequently, people have moved towards RNN, which can deal with worldly information utilizing repetitive associations in hidden layers [12]. Be that as it may, the major disadvantage of RNN is its short-term memory, which is inadequate for genuine real-life varieties in gestures or actions. To take care of this issue, long short-term memory (LSTM) [58] was presented which can handle longer-range temporal structures. Gestures or actions, in a video sequence, can be considered as a sequential temporal evaluation of body/body-part in a space-time representation. So, 3D-CNN/RNN/LSTM is the network generally applied in video-based action/gesture recognition. In addition to 3D-CNNs, recurrent neural networks have also been applied for dynamic hand gesture classification [149, 249]. [29] has extracted hand trajectory and hand posture features from RGB-D data and then a two-stream recurrent neural network (2S-RNN) is used to fuse multi-modal features. The spatio-temporal graphs are good for representing long-range spatio-temporal variations. Hence, a combination of high-level spatio-temporal graphs and RNN can also be applied to resolve the issue of spatio-temporal representation in RNN [84]. The long short-term memory problem and vanishing/exploding problem of RNN can be handled to some extent by adding 'gates' in LSTM. Hence networks based on LSTM can be efficiently utilized for the representation of dynamic gestures [43, 129, 217]. However, in both RNN and LSTM, the problem of vanishing/expanding gradient is much acute compared to CNN and they become more data-hungry. Gated recurrent units (GRU) are simplified LSTM units with adaptive gate parameters with fewer parameters which makes the training process faster. [197] presented a skeleton-based dynamic hand gesture acknowledgment technique that separates geometric features into various parts and uses a gated recurrent unit-recurrent neural network (GRU-RNN) for each featuring part. Since each divided feature component has fewer dimensions than the whole element, the number of hidden units needed for optimization is decreased. Subsequently, the plan accomplished improved recognition performance with fewer parameters.

Thus, more or less, deep learning procedures can give exceptional execution in both feature extraction and recognition tasks due to their inherent feature learning ability. The powerful and effective algorithms of deep networks are fit for tackling complex pattern recognition and optimization tasks.

Hand Gesture Databases

The advancement of standard hand gesture datasets is an essential requirement for the dependable analysis and verification of hand gesture recognition techniques. There are a few freely accessible hand gesture databases that are created with the end goal of hand motion investigation and similar examinations. Several authors have come out with such lists of databases [11, 12, 35, 170]. But most of them have not given a detailed analysis of the same in a concise way, though all of them tried to include the most-used hand and human activity databases. In this work, we have tried to collate a comprehensible list of the 50 most used freely accessible hand gesture databases with their brief description in two tables. Table 2 mainly gives the content and description, whereas Table 3 gives the link of the publicly available sources.

Applications, Recent Advancements and Future Scopes of VGR Systems

Gesture Interface for Vision-Based HCI System

The approach of vision-based hand gesture is more intrinsic and suitable compared to other glove-based approaches used in HCI since it can be used in the field of vision of a camera anywhere and at any time. The operator does not need to master any special hardware and, thus, it is easier to deploy. A vision-based approach also enables a variety of gestures to be used that can be updated in the software. Computer vision methods can enable HCI that is difficult or impossible to achieve with other modalities. Visual information is important in human–human communication because meaning is conveyed through identity, facial expression, posture, gestures, and other visually observable attributes. Therefore, intuitively it is possible to have natural HCI by sensing and perceiving these visual cues from video cameras placed appropriately in the environment. The major benefit of VGR is that it requires modest gadgets in terms of cost as input devices. Even an advanced camera can be incorporated with a solitary chip. Large-scale manufacturing is thus a lot simpler in contrast to other info gadgets like data gloves with mechanical components. Furthermore, the expense of image processing equipment can be minimized since most computers now have a central processing unit and graphics processing unit fast enough to perform these computer vision tasks. While other information gadgets like a mouse, joystick, and trackpad are restricted to a particular capacity; camera-based computer vision techniques are flexible enough to offer an entire scope of conceivable future applications in a human–computer association as well in user validation, video conferencing, and distance schooling. Another significant benefit of computer vision is that it is non-intrusive. Cameras are open information gadgets that do not need direct contact with the user to detect activities. The user can communicate with the computer without wires and without controlling mediator gadgets. Moreover, humans are more comfortable in communicating with body postures or gestures as compared to using some mechanical techniques like clicking the mouse or pressing the keyboard, or touching a touch-sensitive screen and thus experience more comfortable and better natural interactions than with traditional interaction techniques. These are the major advantages of a VGR system, including a natural, contact-free method of interaction. However, vision-based gesture interfaces also have many disadvantages, including user fatigue, cultural differences, the requirement of high-speed processing, and noise sensitivity. Nevertheless, it is more difficult to use because current computer vision schemes are still limited in processing such highly articulated, non-convex, and flexible objects like the human hand. Vision-based recognition is amazingly difficult not just in light of its assorted settings, different translations, and spatio-transient varieties yet additionally as a result of the complex non-unbending properties of the human hand. The current classifiers utilized for vision-based motion recognition are not prepared to handle all the motion characterization issues at the same time. Every one of them has at least one downside restricting the general execution of the motion recognition strategies.

Applications and Recent Advancements

Despite all the drawbacks, the number of VGR systems is assumed to increase more in daily life; and as such, interactive technology needs to be designed effectively to provide a more natural way of communication. Therefore, currently, vision-based gesture recognition has become a major research field in HCI and there is a various real-life implementation of VGR. More specifically hand gestures-based VGR systems can provide a noncontact input modality. The widespread use of gesture-based interfaces for vision-based HCI is possible due to the advantages mentioned above. One of the forward leaps in VGR is the presentation of Microsoft Kinect® as a contact-less interface [248]. The Kinect has huge potential in different applications, for example, medical care [82], educational training [116], and so on. Be that as it may, its poor open-air execution and depth resolution limits its convenience. As of late, SoftKinetic's Gesture Control Technology is consolidated in BMW vehicles to permit drivers to explore the in-vehicle infotainment framework easily [1]. Most as of late executed and some proposed utilization of VGR incorporate sign language recognition [127], virtual reality (VR) [187], virtual game [112], augmented reality

Table 2 Summary of hand gesture databases with brief description

Sl. No.	Dataset	Contents	Description
RGB/grayscale dataset			
1	NUS hand posture dataset-I, 2010 [114]	10 classes, 1 subject, 240 samples	Both color and gray scale
2	NUS hand posture dataset-II, 2012 [171]	10 classes, 40 subjects, 2750 samples	Complex natural background
3	UNIGEhands Dataset, 2015 [15]	37.21 and 37.63 minutes of positives and negative video sequencs	Egocentric videos in 5 natural locations (Office, Bar, Kitchen, Bench, Street)
4	OUHANDS hand gesture dataset, 2016 [141]	2150 training and 1000 test images	Different background, contains body gesture, collected by Intel RealSense
5	Cambridge hand gesture dataset, 2007 [103]	9 classes, 2 subjects, 900 image sequences	Different illumination conditions
6	Gesture dataset by Shen et al. [195], 2012	10 classes, 15 subjects, 1050 samples	Different poses of thumb, fist, all fingers extended
7	Sebastien Marcel hand posture and gesture datasets, 2001 [216]	Three static datasets, with 10 (gray scale), 12 (color), and 6 (gray scale) classes. One dynamic dataset with 4 classes	Both simple and complex background
8	Aalborg Video Database, 2004 [75]	9 static and 4 dynamic classes	Hand gestures over a wooden table
9	Sebastien Marcel interact play database, 2004 [91]	16 classes, 22 subjects, 50 samples/ subject	Single and both hand dataset
10	Gesture dataset by Yoon et al. [242]	48 classes, 20 subjects, 9600 samples	Alphabetical gestures containing sequences of xy coordinates
11	Keck gesture dataset, 2009 [128]	Keck gesture dataset, 2009	Military signals with training set in simple background and testing set in complex background
12	Massey gesture dataset, 2005 [38]	6 classes, 5 subjects, about 1500 frames	Different image frames of gestures in different illumination
13	IDIAP two-handed gesture dataset, 2005 [139]	7 classes, 7 subjects	Special color-glove to differentiate between right and left hand
14	FABO gesture dataset, 2006 [62]	21 classes divided into two sets	Face and body gesture dataset in fixed background
15	IBGHT dataset, 2015 [11]	36 classes, 60 video sequences	0–9 numeric and A–Z alphabetic color dataset
16	10 Palm Graffiti Digits dataset, 2009 [7]	10 classes, 30 examples per class	0–9 digits in continuous stream, colored glove in training set, both easy and hard test set
17	NITS hand gesture dataset, 2015 [145]	40 classes, 20 subjects, divided into 7 sets	Gestures collected in lab environment with colored fingertip
18	The 20BN-jester dataset, 2019 [216]	148,092 videos in total: 118,562 for training, 14,787 for validation and 14,743 for testing	Densely labeled video clips that show humans performing predefined hand gestures in front of a laptop camera or webcam
RGB-D dataset			
19	NTU posture dataset by Ren et al., 2011 [182]	10 classes, 10 subjects, 1000 samples	Color as well as depth maps, cluttered background, recorded with Kinect
20	ColorTip dataset, 2013 [210]	7 subjects, 9 classes, 7 training sequences of between 600 and 2000 depth frames	Fingertips are covered with colored glove for automatic annotation
21	NYU hand pose dataset, 2014 [214]	72,757 and 8252 frames in training and test sets	2 users, data from 3 Kinects (frontal and 2 sides)
22	General-HANDS data-set, 2014	22 sequences	Different view-points, scales, poses, and occlusions
23	VPU Hand Gesture dataset (HGds), 2008 [107]	12 classes, 11 subjects	One static pose video per gesture (252 grayscale frames); collected by time-of-flight camera

Table 2 (continued)

Sl. No.	Dataset	Contents	Description
24	ChaLearn gesture data, 2011 [65]	62,000 samples	Hand gestures including body gestures; recorded with Kinect
25	MSRC-12 Kinect gesture dataset, 2012 [54]	12 classes, 30 subjects, 6244 samples	Human movement including body gestures; recorded with Kinect
26	ChaLearn multi-modal gesture dataset, 2013 [50]	20 classes, 27 subjects, 13,858 samples	Including body gestures
27	NATOPS aircraft handling signals database, 2011 [207]	24 classes, 20 subjects, 9600 samples	Including body gestures
28	ChAirGest multi-modal dataset , 2013 [186]	10 classes, 10 subjects, 1200 samples	Recorded with Kinect and inertial motion units
29	Sheffield KInect Gesture (SKIG) dataset, 2013 [131]	10 classes, 6 subjects, 2160 samples	Two illumination condition, recorded with Kinect and RGB cameras
30	Full Body Gesture (FBG) database, 2006 [81]	14 normal gesture of daily life, 10 abnormal gesture classes, 20 subjects	Full body 3D dataset
31	10 3D digit dataset by Berman et al., 2013 [55]	10 classes, 8 subjects	0–9 in continuous stream, dataset collected using PrimeSense 3D camera
32	6D Motion Gesture (6DMG) dataset, 2012 [34]	10 digit classes, 26 upper and lower alphabet classes each	Dataset is recorded by Wii device with trajectories in space, includes some body gestures also
33	Hand gesture datasets, University of Padova, 2014 [140]	10 ASL classes, 14 subjects	Dataset is collected with both leap motion controller and Kinect. First of its kind dataset collected by both.
34	Hand gesture datasets, University of Padova, 2015 [143]	Several static gestures	Collected with Senz3D device
35	Hand gesture datasets, University of Polytechnique, Madrid, 2015 [137]	10 classes, divided into 2 sets with 5 gestures each	Collected with Senz3D device
36	SP-EMD dataset, 2015 [222]	10 gestures with 20 different poses, 5 subjects	In two different illumination, collected using Kinect
37	DHG-14/28, 2016 [42]	14 classes, 20 subjects	Gestures are collected using Kinect in two ways: using one finger and the whole hand
38	DVS128 gesture dataset, 2017 [8]	11 classes, 29 subjects	3 illumination condition, collected with DSV128
39	BigHand2.2M hand posture dataset, 2017 [243]	2.2 million depth maps	Collected with Intel RealSense, some are egocentric images
40	EgoGesture Dataset, 2017 [26]	83 classes, 50 subjects, 6 scenes, 24161 RGB-D video samples	First-person view gestures, collected using Intel RealSense SR300
41	VIVA dataset, 2014 [159]	19 classes, 8 subjects, 885 RGB-D video samples	Driver hand gestures in single scene, collected using Microsoft Kinect
42	NVIDIA Gesture (nvGesture) dataset, 2016 [149]	25 classes, 20 subjects, 1532 RGB-D video samples	Driver hand gestures collected using SoftKinetic DS325 and a top-mounted DUO 3D sensor to record a pair of stereo-IR streams
Sign language dataset			
43	Dataset by Kawulok et al., 2014 [97]	32 classes, 18 subjects	Gestures from Polish Sign Language and American Sign Language (ASL)
44	ASL Finger Spelling Dataset, 2011 [175]	24 classes, 9 subjects, 65,000 samples	Alphabet depth dataset
45	Massey 2D Static ASL dataset, 2011 [14]	2425 gestures, 5 subjects	Color ASL dataset
46	Purdue RVL-SLLL ASL Database, 2006 [227]	Different ASL gestures by 14 subjects	Alphanumeric dataset

Table 2 (continued)

Sl. No.	Dataset	Contents	Description
47	RWTH-BOSTON-104 Database, 2007 [46]	104 signs, 201 videos, about 15000 image frames	Grayscale ASL dataset
48	RWTH-BOSTON-400, 2008 [45]	406 signs; extended upon 2007 dataset	Color ASL dataset
49	MSR/MSRA Gesture 3D dataset, 2011 [126]	12 ASL gesture classes, 10 subjects	Hand tracking ASL dataset. Some are daily gestures
50	Kaggle Sign Language dataset, 2017	24 classes A–Z excluding J, Z and 10 classes of digits 0–9, mimics EMNIST	ASL image dataset

(AR) [180], smart video conferencing [142], smart home and office [155], medical services and clinical help (MRI navigation) [82], robotic surgery [213], wheelchair control [104], driver observing [204], vehicle control [168], interactive presentation module [244], virtual study hall [71], web-based business (e-commerce) [9], etc. A portion of the significant applications (see Fig. 17) of hand gesture-based HCI applications are illustrated below:

- *Augmented reality and virtual reality* Hand gestures can be very useful for realistic manipulations of virtual objects in virtual environments [187] and as an interface for virtual gaming [112]. Many problems like detection, registration and tracking can be solved using augmented reality techniques [180].
- *Sign language recognition* Hand gestures are useful for sign language recognition for the deaf–mute community [127]. The system mainly acts as an interpreter between the deaf/mute and others.
- *Vehicle monitoring and vehicle control* Gesture-based interfaces may be used to operate a vehicle [168], and also for driver monitoring [204].
- *Healthcare and medical assistance* Gesture-based interfaces have many applications in healthcare and medicine, for example, MRI navigation in the operating room [82], and medical volume visualization tasks, browsing radiology images may be some of the possible applications. Gestures can also be used to train physicians in robotic surgery [213] and medical assistance for physically disabled persons, including hand gesture-based wheelchair control [104].
- *Information retrieval* Gesture-based interfaces can also be used for day-to-day information retrieval from the internet [166].
- *Education* Gesture interfaces for controlling presentations (e.g., powerpoint®) is helpful for teachers [244]. Gesture-based interfaces can be used for window menu activation.
- *Desktop, television control and tablet PC applications* Gesture interfaces can be useful in controlling desktop, television, etc. and also for tablet PC applications [155].

Future Scope

Gestures can be made universal and users can apply user-friendly gestures in place of multi-step interactions for communication. With a worldwide focus on reducing the risk of spreading bacteria and viruses, this sort of solution would undoubtedly be welcomed by all. Moreover, as the world adapts to the new changes after the COVID-19 pandemic, touch-less technology can be the 'new normal' in such situations to minimize the risk of a global health crisis. For instance, in airports, if cameras and hardware

Table 3 Publicly available hand gesture databases with their sources

Sl. No.	Dataset	Static(S) and/or Dynamic(D)	Source
RGB/grayscale dataset			
1	NUS hand posture dataset-I, 2010	S	http://www.ece.nus.edu.sg/stfpage/elepv/NUS-HandSet/
2	NUS hand posture dataset-II, 2012	S	http://www.ece.nus.edu.sg/stfpage/elepv/NUS-HandSet/
3	UNIGEhands Dataset, 2015	S	http://alejobetancourt.com/resume/dataset?id=1
4	OUHANDS hand gesture dataset, 2016	S	http://www.ouhands.oulu.fi
5	Cambridge hand gesture dataset, 2007	S and D	http://www.iis.ee.ic.ac.uk/tkkim/ges/_db.htm
6	Gesture dataset by Shen et al. 2012	S and D	http://users.eecs.northwestern.edu/xsh835/GestureDataset.zip
7	Sebastien Marcel hand posture and gesture datasets, 2001	S and D	http://www.idiap.ch/resource/gestures/
8	Aalborg Video Database, 2004	S and D	http://www-prima.inrialpes.fr/FGnet/data/03-Pointing/index.html
9	Sebastien Marcel interact play database, 2004	D	https://www.idiap.ch/resource/interactplay/
10	Gesture dataset by Yoon et al. 2001	D	Available on e-mail request to yoonhs@etri.re.kr
11	Keck gesture dataset, 2009	D	http://users.umiacs.umd.edu/zhuolin/Keckgesturedataset.html
12	Massey gesture dataset, 2005	D	https://www.massey.ac.nz/albarcza/gesture_dataset2012.html
13	IDIAP two-Handed Gesture Dataset, 2005	D	https://www.idiap.ch/resource/twohanded/
14	FABO gesture dataset, 2006	D	https://mmv.eecs.qmul.ac.uk/fabo/
15	IBGHT dataset, 2015	D	http://ibg-usm.org
16	10 Palm Graffiti Digits dataset, 2009	D	http://vlm1.uta.edu/athitsos/projects/digits/
17	NITS dataset, 2015	D	https://joyeetasingha26.wixsite.com/nits-database
18	The 20BN-jester dataset, 2019	D	https://20bn.com/datasets/jester
RGB-D dataset			
19	NTU posture dataset by Ren et al. 2011	S	*http : //rose1.ntu.edu.sg/datasets/actionrecognition.asp*
20	ColorTip dataset, 2013	S	*https : //imatge.upc.edu/web/res/colortip*
21	NYU hand pose dataset, 2014	S	https://jonathantompson.github.io/NYU_Hand_Pose_Dataset.htm
22	General-HANDS data-set, 2014	S	*http : //homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htmgesture*
23	VPU Hand Gesture dataset (HGds), 2008	S	http://www-vpu.eps.uam.es/DS/HGds/
24	ChaLearn gesture data, 2011	D	*http : //gesture.chalearn.org/data*
25	MSRC-12 Kinect gesture dataset, 2012	D	http://research.microsoft.com/en-us/um/cambridge/projects/msrc12/
26	ChaLearn multi-modal gesture data, 2013	D	*http : //sunai.uoc.edu/chalearn/*
27	NATOPS aircraft handling signals database, 2011	D	*http : //groups.csail.mit.edu/mug/natops/*
28	ChAirGest multi-modal dataset , 2013	D	https://project.heia-fr.ch/chairgest/Pages/Download.aspx
29	Sheffield KInect Gesture (SKIG) dataset, 2013	D	*http : //lshao.staff.shef.ac.uk/data/SheffieldKinectGesture.htm*
30	Full Body Gesture (FBG) Database, 2006	D	*http : //gesturedb.korea.ac.kr/*
31	10 3D digit dataset by Berman et al. 2013	D	Available on e-mail request to sigalbe@bgu.ac.il
32	6D Motion Gesture (6DMG) dataset, 2012	D	http://web.cs.wpi.edu/~claypool/mmsys-dataset/2012/6dmg/

Table 3 (continued)

Sl. No.	Dataset	Static(S) and/or Dynamic(D)	Source
33	Hand gesture datasets, University of Padova, 2014	S	*http* : *//lttm.dei.unipd.it/downloads/gesture*
34	Hand gesture datasets, University of Padova, 2015	D	*http* : *//lttm.dei.unipd.it/downloads/gesture*
35	Hand gesture datasets, University of Polytechnique, Madrid, 2015	S and D	*https* : *//www.gti.ssr.upm.es/data/HandGesture*$_d$*atabase.html*
36	SP-EMD dataset, 2015	D	*https* : *//sites.google.com/site/spemdKinect*
37	DHG-14/28, 2016	D	*http* : *//www* − *rech.telecom* − *lille.fr/DHGdataset/*
38	DVS128 gesture dataset, 2017	D	*http* : *//research.ibm.com/dvsgesture/*
39	BigHand2.2M hand posture dataset, 2017	S	Available on e-mail request to hands.iccv17@outlook.com
40	EgoGesture dataset, 2017	D	*http* : *//www.nlpr.ia.ac.cn/iva/yfzhang/datasets/egogesture.html*
41	VIVA datase, 2014	D	http://www.site.uottawa.ca/research/viva/projects/hand_detection
42	NVIDIA Gesture (nvGesture) dataset, 2016	D	*https* : *//research.nvidia.com/publications*
Sign language dataset			
43	Dataset by Kawulok et al. 2014	S	http://sun.aei.polsl.pl/mkawulok/gestures/
44	ASL Finger Spelling Dataset, 2011	S	http://lifeprint.com
45	Massey 2D Static ASL dataset, 2011	S	http://iims.massey.ac.nz/research/letters/
46	Purdue RVL-SLLL ASL Database, 2006	D	Available on e-mail request to wilbur@purdue.edu
47	RWTH-BOSTON-104 Database, 2007	D	http://www-i6.informatik.rwth-aachen.de/dreuw/database.php
48	RWTH-BOSTON-400, 2008	D	http://www-i6.informatik.rwth-aachen.de/aslr/
49	MSR/MSRA Gesture 3D dataset, 2011	D	https://www.microsoft.com/en-us/research/people/zliu/
50	Kaggle Sign Language dataset, 2017	S	https://www.kaggle.com/datamunge/sign-language-mnist

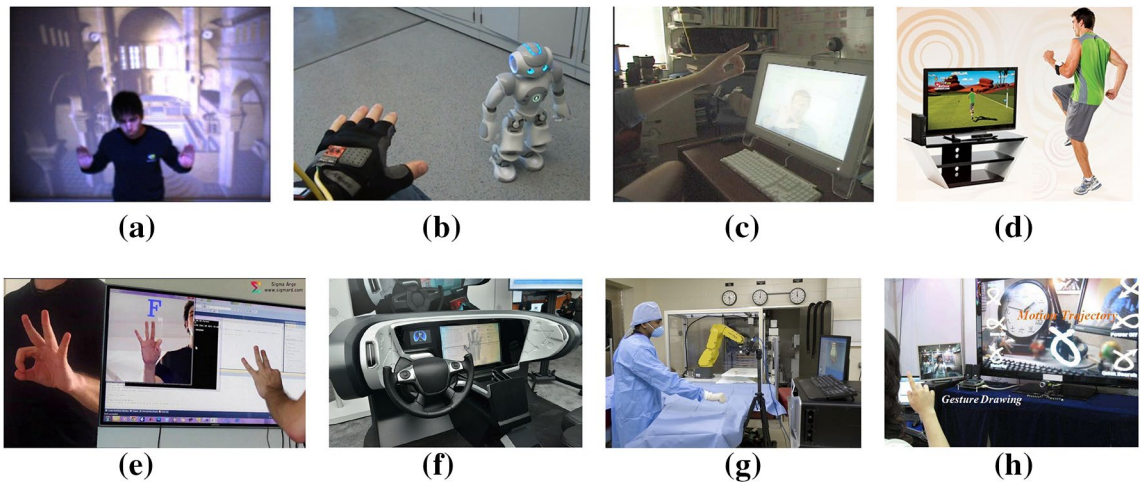

Fig. 17 Applications of hand gesture recognition systems: **a** virtual reality, **b** gesture-based interaction with robots (Picture courtesy http://www.robots-dreams.com/pc-based-robosapien-control-project), **c** desktop computing application, **d** virtual computer games using gesture, **e** sign language recognition, **f** vehicle control (picture courtesy: http://www.automotiveworld.com/news-releases/3D-gesture-recognition-virtual-touch-screen-bring-new-meaning-vehicle-controls/), **g** gesture controlled robotic surgery (Pic. courtesy: http://www.purdueexponent.org/campus/collection_daa8e8c2-3e15-11e0-bb90-0017a4a78c22.html) and **h** television and desktop controlling

are already embedded, passengers can take benefit from hand tracking and gesture recognition to control menus without physically touching a platform. Though there are some other touch-less technologies such as voice recognition, language and pronunciation become a barrier in many instances. Moreover, people are focusing on using smartphones to minimize contact when it comes to aspects such as check-in. However, with smartphones, passengers still often have to touch a screen, which gives a chance of risk. Additionally, at airport border control, it is often forbidden to use a smartphone. So, there are further limits to these existing features. In addition, on roads drivers can control auto navigation through simple in-air movements. In such cases, hand-tracking and gesture recognition technology can provide a hardware-agnostic solution to these problems.

Another major challenge to overcome for a gesture recognizer system is the implementation of an efficient real-time application. A good gesture recognizer should fulfill the following requirements, and the most important aspect is computational efficiency for real-time implementation:

- *Robustness* The system should be robust to real-world conditions like noisy visual information, changing illumination, cluttered and dynamic backgrounds, occlusion, and so on.
- *Scalability* The core of the system should be adaptive to different scales of applications like sign language recognition, robot navigation, virtual environments, and so on.
- *Computational efficiency* The system should be computationally efficient.
- *User's tolerance* The system should detect the mistakes performed by the user and ask the user to repeat them until the mistake is corrected.

As shown in Fig. 18, the real-time implementation of gesture recognition algorithms can be made by using graphics processing units (GPUs) alone or in combination with general-purpose CPUs to increase the processing speed.

Conclusion

Hand gesture recognition is a significant field of exploration in computer vision with different applications in HCI. Applications incorporate desktop tools, computer games, healthcare, medical assistance, robotics, sign language, vehicle monitoring, and virtual reality environments. Interfaces support unimodal or multimodal connection by utilizing computer vision, speech recognition, wearable sensors, or a mix of these and different advancements. Utilizing more than one modality can make the interaction more natural and accurate, but it also increases the system complexity. Both static and dynamic gestures give a helpful and normal human–computer interface. Dynamic gestures can be grouped depending on their implications and appearances. They can be obtained primarily from vision-based frameworks or from wearable-sensor-based gloves. In principle, vision-based gesture interfaces should be preferred to data gloves because of their simplicity and low cost. While glove-based gesture recognition is almost a tackled issue, vision-based gesture recognition is yet in

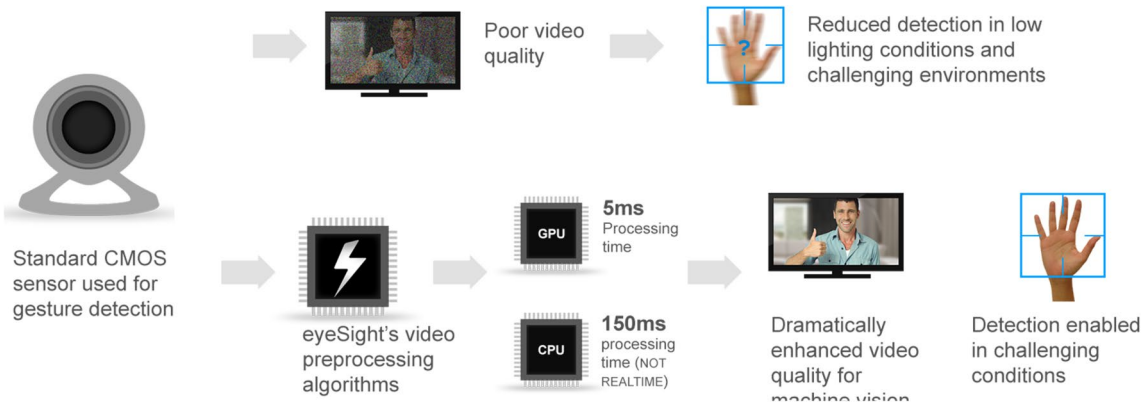

Fig. 18 Use of GPU in gesture recognition (courtesy: http://community.arm.com/groups/arm-mali-graphics/blog/2013/10/06/improved-gesture-detection-with-mali-gpu-compute)

its developing stage. Vision-based gesture recognition typically depends on the proper segmentation of the gesturing body parts. Image segmentation is altogether influenced by factors including physical movement, variations in illumination and shadows, and background complexity. The complex enunciated state of the hand makes it difficult to represent the appearance of gestures. Moreover, the variety of gesture boundaries due to spatial-transient differences of hand gestures makes the spotting and recognition process more troublesome. Recognition of static, as well as dynamic gestures, becomes more difficult if there is occlusion. Occlusion estimation is a challenging problem in its own right and an active area of research. Impediment assessment is a difficult issue by its own doing and a functioning space of exploration. Occlusions can be estimated using multiple cameras or tracking-based methods. The inclusion of depth information in gesture recognition can make the recognition process more accurate. Deep learning techniques have acquired another point of view for different applications of computer vision. Deep learning strategies can be used in both feature extraction and recognition inferable from their underlying feature learning ability in finding salient latent structures within unlabeled and unstructured raw data.

This paper surveyed the main approaches in vision-based hand gesture recognition for HCI. Major topics were different classes of gestures and their acquisition; gesture system architectures; and applications and recent advances of gesture-based human–computer interfaces in HCI. A detailed discussion was provided on the features and major classifiers in current use. Also, a brief description of different hand gesture databases is listed with their available source links. The scope of gesture naturalness and expressiveness can be enhanced by including facial expressions or allowing the use of both hands. However, this increases the size of the gesture vocabulary, inherently increasing the complexity.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. 'softkinetic's gesture control technology rolls out in additional car model; 2017.
2. von Agris U, Knorr M, Kraiss KF. The significance of facial features for automatic sign language recognition. In: Proceedings of 8th IEEE international conference automatic face gesture recognition, 2008; FG '08. pp. 1–6.
3. Ahad MAR, Tan JK, Kim H, Ishikawa S. Motion history image: its variants and applications. Mach Vis Appl. 2012;23(2):255–81.
4. Akhter I, Sheikh Y, Khan S, Kanade T. Trajectory space: a dual representation for nonrigid structure from motion. IEEE Trans Pattern Anal Mach Intell. 2011;33(7):1442–56.
5. Akita K. Image sequence analysis of real world human motion. Pattern Recognit. 1984;17(1):73–83.
6. Alberola C, Juan F, Ruiz J, Socas R. Human hand postures and gestures recognition: towards a human-gesture communication interface. In: Proceedings of international conference image processing (ICIP), 1999, vol. 4. pp. 222–6.
7. Alon J, Athitsos V, Yuan Q, Sclaroff S. A unified framework for gesture recognition and spatiotemporal gesture segmentation. IEEE Trans Pattern Anal Mach Intell. 2009;31(9):1685–99.
8. Amir A, Taba B, Berg D, Melano T, McKinstry J, Di Nolfo C, Nayak T, Andreopoulos A, Garreau G, Mendoza M, et al. A low power, fully event-based gesture recognition system. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. pp. 7243–52.
9. Arafa Y, Mamdani A. Building multi-modal personal sales agents as interfaces to e-commerce applications. In: International computer science conference on active media technology. Springer; 2001. pp. 113–133.
10. Aran O, Akarun L. Recognizing two handed gestures with generative, discriminative and ensemble methods via fisher kernels. In: International workshop on multimedia content representation, classification and security, 2006. Springer. pp. 159–66.

11. Asaari MSM, Rosdi BA, Suandi SA. Intelligent biometric group hand tracking (IBGHT) database for visual hand tracking research and development. Multim Tools Appl. 2014;70(3):1869–98.
12. Asadi-Aghbolaghi M, Clapes A, Bellantonio M, Escalante HJ, Ponce-López V, Baró X, Guyon I, Kasaei S, Escalera S. A survey on deep learning based approaches for action and gesture recognition in image sequences. In: Automatic face & gesture recognition (FG 2017), 2017 12th IEEE international conference on, 2017. IEEE. pp. 476–83.
13. Avinash B, Ghosh D, Ari S. Color hand gesture segmentation for images with complex background. In: Proceedings of international conference circuits, power and computing technologies (ICCPCT), 2013. pp. 1127–31.
14. Barczak A, Reyes N, Abastillas M, Piccio A, Susnjak T. A new 2D static hand gesture colour image dataset for ASL gestures; 2011.
15. Betancourt A, Morerio P, Barakova EI, Marcenaro L, Rauterberg M, Regazzoni CS. A dynamic approach and a new dataset for hand-detection in first person vision. In: International conference on computer analysis of images and patterns, 2015. Springer. pp. 274–87.
16. Bhuyan M. FSM-based recognition of dynamic hand gestures via gesture summarization using key video object planes. Int J Comput Commun Eng. 2012;6:248–59.
17. Bhuyan M, Ghosh D, Bora P. Continuous hand gesture segmentation and co-articulation detection. In: Computer vision, graphics and image processing, 2006. Springer. pp. 564–75.
18. Bhuyan M, Ghosh D, Bora P. In: Feature extraction from 2d gesture trajectory in dynamic hand gesture recognition. In: Cybernetics and intelligent systems, 2006 IEEE conference on, 2006. IEEE. pp. 1–6.
19. Bhuyan MK, Kumar DA, MacDorman KF, Iwahori Y. A novel set of features for continuous hand gesture recognition. J Multimodal User Interfaces. 2014;8(4):333–43.
20. Bilen H, Fernando B, Gavves E, Vedaldi A, Gould S. Dynamic image networks for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016. pp. 3034–42.
21. Bobick AF, Davis JW. The recognition of human movement using temporal templates. IEEE Trans Pattern Anal Machine Intell. 2001;23(3):257–67.
22. Brown DA, Craw I, Lewthwaite J. A SOM based approach to skin detection with application in real time systems. BMVC Citeseer. 2001;1:491–500.
23. Brox T, Bruhn A, Papenberg N, Weickert J. High accuracy optical flow estimation based on a theory for warping. In: European conference on computer vision, 2004. Springer. pp. 25–36.
24. Brox T, Malik J. Large displacement optical flow: descriptor matching in variational motion estimation. IEEE Trans Pattern Anal Machine Intell. 2011;33(3):500–13.
25. Campbell LW, Becker DA, Azarbayejani A, Bobick AF, Pentland A. Invariant features for 3-D gesture recognition. In: Automatic face and gesture recognition, 1996. Proceedings of the second international conference on, 1996. IEEE. pp. 157–62.
26. Cao C, Zhang Y, Wu Y, Lu H, Cheng J. Egocentric gesture recognition using recurrent 3D convolutional neural networks with spatiotemporal transformer modules. In: Proceedings of the IEEE international conference on computer vision, 2017. pp. 3763–71.
27. Chai D, Ngan K. Face segmentation using skin-color map in videophone applications. IEEE Trans Circuits Syst Video Technol. 1999;9(4):551–64.
28. Chai D, Ngan KN. Face segmentation using skin-color map in videophone applications. IEEE Trans Circuits Syst Video Technol. 1999;9(4):551–64.
29. Chai X, Liu Z, Yin F, Liu Z, Chen X. Two streams recurrent neural networks for large-scale continuous gesture recognition. In: Pattern recognition (ICPR), 2016 23rd international conference on, 2016. IEEE. pp. 31–6.
30. Chai Y, Shin S, Chang K, Kim T. Real-time user interface using particle filter with integral histogram. IEEE Trans Consum Electron. 2010;56(2):510–5.
31. Chakraborty BK, Bhuyan M, Kumar S. Combining image and global pixel distribution model for skin colour segmentation. Pattern Recognit Lett. 2017;88:33–40.
32. Chakraborty BK, Sarma D, Bhuyan M, MacDorman KF. Review of constraints on vision-based gesture recognition for human–computer interaction. IET Comput Vis. 2017;12(1):3–15.
33. Chen L, Zhou J, Liu Z, Chen W, Xiong G. A skin detector based on neural network. In: Communications, circuits and systems and West Sino expositions, IEEE 2002 international conference on, vol. 1, 2002. IEEE. pp. 615–19.
34. Chen M, AlRegib G, Juang BH. 6dmg: a new 6D motion gesture database. In: Proceedings of the 3rd multimedia systems conference, 2012. ACM. pp. 83–8.
35. Cheng H, Yang L, Liu Z. Survey on 3D hand gesture recognition. IEEE Trans Circuits Syst Video Technol. 2016;26(9):1659–73.
36. Cheok MJ, Omar Z, Jaward MH. A review of hand gesture and sign language recognition techniques. Int J Mach Learn Cybern. 2017;10(1):1–23.
37. Ciregan D, Meier U, Schmidhuber J. Multi-column deep neural networks for image classification. In: Computer vision and pattern recognition (CVPR), 2012 IEEE conference on, 2012. IEEE. pp. 3642–9.
38. Dadgostar F, Barczak ALC, Sarrafzadeh A. A color hand gesture database for evaluating and improving algorithms on hand gesture and posture recognition. Massey University; 2005.
39. Dalal N, Triggs B, Schmid C. Human detection using oriented histograms of flow and appearance. In: European conference on computer vision, 2006. Springer. pp. 428–41.
40. Dardas N, Chen Q, Georganas ND, Petriu EM. Hand gesture recognition using bag-of-features and multi-class support vector machine. In: Haptic audio–visual environments and games (HAVE), 2010 IEEE international symposium on, 2010. IEEE. pp. 1–5.
41. Dardas NH, Georganas ND. Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans Instrum Meas. 2011;60(11):3592–607.
42. De Smedt Q, Wannous H, Vandeborre JP. Skeleton-based dynamic hand gesture recognition. In: Computer vision and pattern recognition workshops (CVPRW), 2016 IEEE conference on, 2016. IEEE. pp. 1206–14.
43. Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015. pp. 2625–34.
44. Dondi P, Lombardi L, Porta M. Development of gesture-based human–computer interaction applications by fusion of depth and colour video streams. IET Comput Vis. 2014;8(6):568–78.
45. Dreuw P, Neidle C, Athitsos V, Sclaroff S, Ney H. Benchmark databases for video-based automatic sign language recognition. In: LREC; 2008.
46. Dreuw P, Rybach D, Deselaers T, Zahedi M, Ney H. Speech recognition techniques for a sign language recognition system. In: Eighth annual conference of the international speech communication association, 2007.
47. El Ayadi M, Kamel MS, Karray F. Survey on speech emotion recognition: features, classification schemes, and databases. Pattern Recognit. 2011;44(3):572–87.

48. Erol A, Bebis G, Nicolescu M, Boyle RD, Twombly X. Vision-based hand pose estimation: a review. Comput Vis Image Underst. 2007;108(1–2):52–73.
49. Escalante HJ, Guyon I, Athitsos V, Jangyodsuk P, Wan J. Principal motion components for one-shot gesture recognition. Pattern Anal Appl. 2017;20(1):167–82.
50. Escalera S, Gonzàlez J, Baró X, Reyes M, Lopes O, Guyon I, Athitsos V, Escalante, H. Multi-modal gesture recognition challenge 2013: dataset and results. In: Proceedings of the 15th ACM on international conference on multimodal interaction, 2013. ACM. pp. 445–52.
51. Farnebäck G. Two-frame motion estimation based on polynomial expansion. In: Scandinavian conference on Image analysis, 2003. Springer. pp. 363–70.
52. Feng KP, Yuan F. Static hand gesture recognition based on hog characters and support vector machines. In: Instrumentation and measurement, sensor network and automation (IMSNA), 2013 2nd international symposium on, 2013. IEEE. pp. 936–38.
53. Finlayson G, Drew M, Lu C. Intrinsic images by entropy minimization. In: Pajdla T, Matas J (eds) Computer vision—ECCV 2004, Lecture notes in computer science, 2004, vol. 3023. Berlin: Springer. pp. 582–95.
54. Fothergill S, Mentis H, Kohli P, Nowozin S. Instructing people for training gestural interactive systems. In: Proceedings of the SIGCHI conference on human factors in computing systems, 2012. ACM. pp. 1737–46.
55. Frolova D, Stern H, Berman S. Most probable longest common subsequence for recognition of gesture character input. IEEE Trans Cybern. 2013;43(3):871–80.
56. Fukunaga K, Hostetler L. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans Inf Theory. 1975;21(1):32–40.
57. Fukushima K, Miyake S. Neocognitron: a self-organizing neural network model for a mechanism of visual pattern recognition. In: Competition and cooperation in neural nets, 1982. Springer. pp. 267–85.
58. Gers FA, Schraudolph NN, Schmidhuber J. Learning precise timing with LSTM recurrent networks. J Mach Learn Res. 2002;3(Aug):115–43.
59. Ghosh DK, Ari S. A static hand gesture recognition algorithm using k-mean based radial basis function neural network. In: Information, communications and signal processing (ICICS) 2011 8th international conference on, 2011. IEEE. pp. 1–5.
60. Ghosh DK, Ari S. Static hand gesture recognition using mixture of features and SVM classifier. In: Communication systems and network technologies (CSNT), 2015 fifth international conference on. IEEE, 2015. pp. 1094–99.
61. Goncalves L, Di Bernardo E, Ursella E, Perona P. Monocular tracking of the human arm in 3D. In: Proceedings of IEEE International Conference on Computer Vision. IEEE; 1995. pp. 764–70.
62. Gunes H, Piccardi M. A bimodal face and body gesture database for automatic analysis of human nonverbal affective behavior. In: Pattern recognition, 2006. ICPR 2006. 18th international conference on, vol. 1, 2006. IEEE. pp. 1148–53.
63. Gupta A, Mittal A, Davis L. Constraint integration for efficient multiview pose estimation with self-occlusions. IEEE Trans Pattern Anal Mach Intell. 2008;30(3):493–506.
64. Gupta B, Shukla P, Mittal A. K-nearest correlated neighbor classification for Indian sign language gesture recognition using feature fusion. In: Computer communication and informatics (ICCCI), 2016 international conference on, 2016. IEEE. pp. 1–5.
65. Guyon I, Athitsos V, Jangyodsuk P, Hamner B, Escalante HJ. Chalearn gesture challenge: design and first results. In: Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on. IEEE, 2012. pp. 1–6.
66. Habili N, Lim CC, Moini A. Segmentation of the face and hands in sign language video sequences using color and motion cues. IEEE Trans Circuits Syst Video Technol. 2004;14(8):1086–97.
67. Hall P, Park BU, Samworth RJ. Choice of neighbor order in nearest-neighbor classification. Ann Stat. 2008;36(5):2135–52.
68. Han J, Awad G, Sutherland A. Automatic skin segmentation and tracking in sign language recognition. IET Comput Vis. 2009;3(1):24–35.
69. Han J, Award G, Sutherland A, Wu H. Automatic skin segmentation for gesture recognition combining region and support vector machine active learning. In: Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on, 2006. IEEE. pp. 237–42.
70. Harding PR, Ellis T. Recognizing hand gesture using Fourier descriptors. In: Pattern recognition, 2004. ICPR 2004. Proceedings of the 17th international conference on, vol. 3, 2004. IEEE. pp. 286–89.
71. Hariharan B, Padmini S, Gopalakrishnan U. Gesture recognition using kinect in a virtual classroom environment. In: Digital information and communication technology and it's applications (DICTAP), 2014 fourth international conference on, 2014. IEEE. pp. 118–24.
72. Heracleous P, Aboutabit N, Beautemps D. Lip shape and hand position fusion for automatic vowel recognition in cued speech for French. IEEE Signal Process Lett. 2009;16(5):339–42.
73. Hewett TT, Baecker R, Card S, Carey T, Gasen J, Mantei M, Perlman G, Strong G, Verplank W. ACM SIGCHI curricula for human–computer interaction. ACM; 1992.
74. Hollister A, Buford WL, Myers LM, Giurintano DJ, Novick A. The axes of rotation of the thumb carpometacarpal joint. J Orthop Res. 1992;10(3):454–60.
75. Holte M, Stórring M. Documentation of pointing and command gestures under mixed illumination conditions: video sequence database; 2004. http://www-prima.inrialpes.fr/FGnet/data/03-Pointing/index.html.
76. Horn BK, Schunck BG. Determining optical flow. Artif Intell. 1981;17(1–3):185–203.
77. Hsu CW, Lin CJ. A comparison of methods for multiclass support vector machines. IEEE Trans Neural Netw. 2002;13(2):415–25.
78. Huang Y, Huang TS, Niemann H. Two-handed gesture tracking incorporating template warping with static segmentation. In: Proceedings of fifth IEEE international conference on automatic face gesture recognition. IEEE, 2002. pp. 275–80.
79. Hung KC. The generalized uniqueness wavelet descriptor for planar closed curves. IEEE Trans Image Process. 2000;9(5):834–45.
80. Hussain SMA, Rashid, AHU. User independent hand gesture recognition by accelerated DTW. In: Informatics, electronics & vision (ICIEV), 2012 international conference on, 2012. IEEE. pp. 1033–7.
81. Hwang BW, Kim S, Lee SW. A full-body gesture database for automatic gesture recognition. In: Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on, 2006. IEEE. pp. 243–48.
82. Jacob M, Cange C, Packer R, Wachs JP. Intention, context and gesture recognition for sterile MRI navigation in the operating room. In: Iberoamerican congress on pattern recognition, 2012. Springer. pp. 220–27.
83. Jaimes A, Sebe N. Multimodal human–computer interaction: a survey. Comput Vis Image Underst. 2007;108(1–2):116–34.
84. Jain A, Zamir AR, Savarese S, Saxena A. Structural-RNN: deep learning on spatio-temporal graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016. pp. 5308–317.
85. Ji S, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. IEEE Trans Pattern Anal Mach Intell. 2012;35(1):221–31.

86. Jiang RM, Sadka AH, Crookes D. Multimodal biometric human recognition for perceptual human–computer interaction. IEEE Trans Syst Man Cybern Part C (Appl Rev). 2010;40(6):676–81.
87. Jones MJ, Rehg J. Statistical color models with application to skin detection. Int J Comput Vis. 2002;46(1):81–96.
88. Jones MJ, Rehg JM. Statistical color models with application to skin detection. Int J Comput Vis. 2002;46(1):81–96.
89. Juang CF, Chang CM, Wu JR, Lee D. Computer vision-based human body segmentation and posture estimation. IEEE Trans Syst Man Cybern Part A Syst Hum. 2009;39(1):119–33.
90. Juang CF, Chiu SH, Shiu SJ. Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation. IEEE Trans Syst Man Cybern Part A Syst Hum. 2007;37(6):1077–87.
91. Just A, Bernier O, Marcel S. HMM and IOHMM for the recognition of mono-and bi-manual 3D hand gestures. IDIAP: technical report; 2004.
92. Kameda Y, Minoh M. A human motion estimation method using 3-successive video frames. In: International conference on virtual systems and multimedia, 1996. pp. 135–40.
93. Karam M. Ph.D. thesis: a framework for research and design of gesture-based human–computer interactions. University of Southampton; 2006. (Ph.D. thesis).
94. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L. Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014. pp. 1725–32.
95. Kavyasree V, Sarma D, Gupta P, Bhuyan M. Deep network-based hand gesture recognition using optical flow guided trajectory images. In: 2020 IEEE applied signal processing conference (ASPCON), 2020. IEEE. pp. 252–56.
96. Kavyasree V, Sarma D, Gupta P, Bhuyan MK. Deep network-based hand gesture recognition using optical flow guided trajectory images. In: Proceedings of the 2nd IEEE conference on applied signal processing (ASPCON), 2020.
97. Kawulok M, Kawulok J, Nalepa J. Spatial-based skin detection using discriminative skin-presence features. Pattern Recognit Lett. 2014;41:3–13.
98. Keskin C, Kıraç F, Kara YE, Akarun L. Real time hand pose estimation using depth sensors. In: Consumer depth cameras for computer vision. Springer; 2013. pp. 119–37.
99. Khan R, Hanbury A, Stoettinger J. Skin detection: a random forest approach. In: Image processing (ICIP), 2010 17th IEEE international conference on, 2010. IEEE. pp. 4613–16.
100. Khanal B, Sidibé D. Efficient skin detection under severe illumination changes and shadows. In: Jeschke S, Liu H, Schilberg D, editors. Intelligent robotics and applications. Lecture notes in computer Science, vol. 7102. Berlin: Springer; 2011. pp. 609–18.
101. Khong VM, Tran TH, Improving human action recognition with two-stream 3D convolutional neural network. In: 2018 1st international conference on multimedia analysis and pattern recognition (MAPR), 2018. IEEE. pp. 1–6.
102. Kim JS, Jang W, Bien Z. A dynamic gesture recognition system for the Korean sign language (KSL). IEEE Trans Syst Man Cybern Part B (Cybern). 1996;26(2):354–9.
103. Kim TK, Wong SF, Cipolla R. Tensor canonical correlation analysis for action classification. In: Computer vision and pattern recognition, 2007. CVPR'07. IEEE conference on, 2007. IEEE. pp. 1–8.
104. Kim-Tien N, Truong-Thinh N, Cuong TD. A method for controlling wheelchair using hand gesture recognition. In: Robot intelligence technology and applications 2012, 2013. Springer. pp. 961–70.
105. Kinnunen T, Li H. An overview of text-independent speaker recognition: from features to supervectors. Speech Commun. 2010;52(1):12–40.
106. Kobayashi Y, Kinpara Y, Shibusawa T, Kuno Y. Robotic wheelchair based on observations of people using integrated sensors. In: 2009 IEEE/RSJ international conference on intelligent robots and systems, 2009. IEEE. pp. 2013–8.
107. Kollorz E, Penne J, Hornegger J, Barke A. Gesture recognition with a time-of-flight camera. Int J Intell Syst Technol Appl. 2008;5(3–4):334–43.
108. Konecnỳ J, Hagara M. One-shot-learning gesture recognition using HOG-HOF. J Mach Learn Res. 2014;15:2513–32.
109. Kong Y, Ding Z, Li J, Fu Y. Deeply learned view-invariant features for cross-view action recognition. IEEE Trans Image Process. 2017;26(6):3028–37.
110. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, 2012. pp. 1097–105.
111. Kuiaski D, Neto H, Borba G, Gamba H. A study of the effect of illumination conditions and color spaces on skin segmentation. In: Proceedings of 22nd Brazilian symposium on computer graphics and image processing. 2009;SIBGRAPI. pp. 245–52.
112. Kulshreshth A, Pfeil K, LaViola JJ. Enhancing the gaming experience using 3D spatial user interface technologies. IEEE Comput Graphics Appl. 2017;38(3):16–23.
113. Kumar P, Rautaray SS, Agrawal A. Hand data glove: A new generation real-time mouse for human-computer interaction. In: Recent advances in information technology (RAIT), 2012 1st international conference on, 2012. IEEE. pp. 750–5.
114. Kumar PP, Vadakkepat P, Loh AP. Hand posture and face recognition using a fuzzy-rough approach. Int J Humanoid Robot. 2010;7(03):331–56.
115. Kumar S, Bhuyan M, Chakraborty BK. Extraction of informative regions of a face for facial expression recognition. IET Comput Vis. 2016;10(6):567–76.
116. Kumara W, Wattanachote K, Battulga B, Shih TK, Hwang WY. A kinect-based assessment system for smart classroom. Int J Dist Educ Technol. 2015;13(2):34–53.
117. Kwon J, Park FC. Natural movement generation using hidden Markov models and principal components. IEEE Trans Syst Man Cybern Part B (Cybern). 2008;38(5):1184–94.
118. de La Gorce M, Paragios N. A variational approach to monocular hand-pose estimation. Comput Vis Image Underst. 2010;114(3):363–72.
119. Laptev I, Marszalek M, Schmid C, Rozenfeld B. Learning realistic human actions from movies. In: Computer vision and pattern recognition, 2008. CVPR 2008. IEEE conference on, 2008. IEEE. pp. 1–8.
120. Lathuiliere F, Herve JY. Visual tracking of hand posture with occlusion handling. In: Proceedings of 15th international conference pattern recognition, 2000, vol. 3. pp. 1129–33.
121. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE. 1998;86(11):2278–324.
122. Lee HK, Kim JH. An hmm-based threshold model approach for gesture recognition. IEEE Trans Pattern Anal Mach Intell. 1999;21(10):961–73.
123. Lee J, Kunii T. Constraint-based hand animation. In: Thalmann N, Thalmann D (eds) Models and technology in computer animation, computer animation series, 1993. Springer. pp. 110–27. https://doi.org/10.1007/978-4-431-66911-1_11.
124. Lee JY, Yoo SI. An elliptical boundary model for skin color detection. In: Proceedings of the 2002 international conference on imaging science, systems, and technology, 2002.

125. Li J, Deng L, Gong Y, Haeb-Umbach R. An overview of noise-robust automatic speech recognition. IEEE/ACM Trans Audio Speech Lang Process. 2014;22(4):745–77.
126. Li W, Zhang Z, Liu Z. Action recognition based on a bag of 3D points. In: Computer vision and pattern recognition workshops (CVPRW), 2010 IEEE computer society conference on, 2010. IEEE. pp. 9–14.
127. Lichtenauer JF, Hendriks EA, Reinders MJ. Sign language recognition by combining statistical DTW and independent classification. IEEE Trans Pattern Anal Mach Intell. 2008;30(11):2040–6.
128. Lin Z, Jiang Z, Davis LS. Recognizing actions by shape-motion prototype trees. In: Computer vision, 2009 IEEE 12th international conference on, 2009. IEEE. pp. 444–51.
129. Liu J, Shahroudy A, Xu D, Wang G. Spatio-temporal LSTM with trust gates for 3D human action recognition. Spatio-temporal LSTM with trust gates for 3D human action recognition. In: European conference on computer vision, 2016. Springer. pp. 816–33.
130. Liu L, Sang N, Yang S, Huang R. Real-time skin color detection under rapidly changing illumination conditions. IEEE Trans Consum Electron. 2011;57(3):1295–302.
131. Liu L, Shao L. Learning discriminative representations from RGB-D video data. IJCAI. 2013;1:3.
132. Liu L, Xing J, Ai H, Ruan X. Hand posture recognition using finger geometric feature. In: Pattern recognition (ICPR), 2012 21st international conference on. IEEE, 2012. pp. 565–68.
133. Lu W, Tong Z, Chu J. Dynamic hand gesture recognition with leap motion controller. IEEE Signal Process Lett. 2016;23(9):1188–92.
134. Lucas BD, Kanade T, et al. An iterative image registration technique with an application to stereo vision; 1981.
135. Mahbub U, Imtiaz H, Ahad MAR. An optical flow based approach for action recognition. In: 14th international conference on computer and information technology (ICCIT 2011), 2011. IEEE. pp. 646–51.
136. Mahbub U, Imtiaz H, Roy T, Rahman MS, Ahad MAR. A template matching approach of one-shot-learning gesture recognition. Pattern Recognit Lett. 2013;34(15):1780–8.
137. Maqueda AI, del Blanco CR, Jaureguizar F, García N. Human–computer interaction based on visual hand-gesture recognition using volumetric spatiograms of local binary patterns. Comput Vis Image Underst. 2015;141:126–37.
138. Marasović T, Papić V. Feature weighted nearest neighbour classification for accelerometer-based gesture recognition. In: Software, telecommunications and computer networks (SoftCOM), 2012 20th international conference on, 2012. IEEE. pp. 1–5.
139. Marcel S, Just A. Idiap two handed gesture dataset. Switzerland: IDIAP Research Institute; 2005.
140. Marin G, Dominio F, Zanuttigh P. Hand gesture recognition with leap motion and kinect devices. In: Image processing (ICIP), 2014 IEEE international conference on, 2014. IEEE. pp. 1565–9.
141. Matilainen M, Sangi P, Holappa J, Silvén O. Ouhands database for hand detection and pose recognition. In: Image processing theory tools and applications (IPTA), 2016 6th international conference on, 2016. IEEE. pp. 1–5.
142. McCowan L, Gatica-Perez D, Bengio S, Lathoud G, Barnard M, Zhang D. Automatic analysis of multimodal group actions in meetings. IEEE Trans Pattern Anal Mach Intell. 2005;27(3):305–17.
143. Memo A, Minto L, Zanuttigh P. Exploiting silhouette descriptors and synthetic data for hand gesture recognition. In: Smart tools and apps or graphics. 2015.
144. Meyer S, Rakotonirainy A. A survey of research on context-aware homes. In: Proceedings of Australasian information security workshop conference. ACSW frontiers 2003—volume 21, ACSW frontiers '03. Australian Computer Society, Inc., Darlinghurst, Australia, Australia 2003. pp. 159–68.
145. Misra S, Singha J, Laskar R. Vision-based hand gesture recognition of alphabets, numbers, arithmetic operators and ascii characters in order to develop a virtual text-entry interface system. Neural Comput Appl. 2018;29(8):117–35.
146. Mitra S, Acharya T. Gesture recognition: a survey. IEEE Trans Syst Man Cybern Part C (Appl Rev). 2007;37(3):311–24.
147. Moeslund TB, Hilton A, Krüger V. A survey of advances in vision-based human motion capture and analysis. Comput Vis Image Underst. 2006;104(2–3):90–126.
148. Molchanov P, Gupta S, Kim K, Kautz J. Hand gesture recognition with 3D convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2015. pp. 1–7.
149. Molchanov P, Yang X, Gupta S, Kim K, Tyree S, Kautz J. Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016. pp. 4207–15.
150. Mukherjee S, Ahmed SA, Dogra DP, Kar S, Roy PP. Fingertip detection and tracking for recognition of air-writing in videos. Expert Syst Appl. 2019;136:217–29.
151. Murugeswari M, Veluchamy S. Hand gesture recognition system for real-time application. In: Advanced communication control and computing technologies (ICACCCT), 2014 international conference on, 2014. IEEE. pp. 1220–5.
152. Nadgeri SM, Sawarkar S, Gawande AD. Hand gesture recognition using Camshift algorithm. In: Emerging trends in engineering and technology (ICETET), 2010 3rd international conference on, 2010. IEEE. pp. 37–41.
153. Neverova N, Wolf C, Taylor G, Nebout F. Moddrop: adaptive multi-modal gesture recognition. IEEE Trans Pattern Anal Mach Intell. 2015;38(8):1692–706.
154. Ng P, Pun CM. Skin color segmentation by texture feature extraction and k-mean clustering. In: Computational intelligence, communication systems and networks (CICSyN), 2011 third international conference on, 2011. IEEE. pp. 213–8.
155. Ng WL, Ng CK, Noordin NK, Ali BM. Gesture based automating household appliances. In: International conference on human–computer interaction, 2011. Springer. pp. 285–93.
156. Nguyen TN, Vo DH, Huynh HH, Meunier J. Geometry-based static hand gesture recognition using support vector machine. In: Control automation robotics & vision (ICARCV), 2014 13th international conference on, 2014. IEEE. pp. 769–74.
157. Nolker C, Ritter H. Visual recognition of continuous hand postures. IEEE Trans Neural Netw. 2002;13(4):983–94.
158. Ogawara K, Takamatsu J, Hashimoto K, Ikeuchi K. Grasp recognition using a 3D articulated model and infrared images. In: Proceedings of IEEE/RSJ international conference intelligent robotics and systems (IROS), 2003, vol. 2. pp. 1590–5.
159. Ohn-Bar E, Trivedi MM. Hand gesture recognition in real time for automotive interfaces: a multimodal vision-based approach and evaluations. IEEE Trans Intell Transp Syst. 2014;15(6):2368–77.
160. Oikonomidis I, Kyriazis N, Argyros AA. Tracking the articulated motion of two strongly interacting hands. In: 2012 IEEE conference on computer vision and pattern recognition, 2012. IEEE. pp. 1862–9.
161. Oviatt S. Multimodal interfaces. The human–computer interaction handbook: fundamentals, evolving technologies and emerging applications, vol. 14. Mahwah, NJ: Lawrence Erlbaum Assoc.; 2003. pp. 286–304.
162. Pantic M, Rothkrantz LJ. Toward an affect-sensitive multimodal human–computer interaction. Proc IEEE. 2003;91(9):1370–90.

163. Patsadu O, Nukoolkit C, Watanapa B. Human gesture recognition using kinect camera. In: Computer science and software engineering (JCSSE), 2012 international joint conference on, 2012. IEEE. pp. 28–32.
164. Patwardhan KS, Roy SD. Hand gesture modelling and recognition involving changing shapes and trajectories, using a predictive Eigentracker. Pattern Recognit Lett. 2007;28(3):329–34.
165. Pavlovic VI, Sharma R, Huang TS. Visual interpretation of hand gestures for human–computer interaction: a review. IEEE Trans Pattern Anal Mach Intell. 1997;19(7):677–95.
166. Peng SY, Wattanachote K, Lin HJ, Li KC. A real-time hand gesture recognition system for daily information retrieval from internet. In: Ubi-media computing (U-Media), 2011 4th international conference on, 2011. IEEE. pp. 146–51.
167. Phung SL, Chai D, Bouzerdoum A. Adaptive skin segmentation in color images. In: Proceedings of IEEE international conference acoustics, speech, and signal processing (ICASSP '03), 2003, vol. 3, p. III-353-6.
168. Pickering C. The search for a safer driver interface: a review of gesture recognition human machine interface. Comput Control Eng J. 2005;16(1):34–40.
169. Pigou L, Dieleman S, Kindermans PJ, Schrauwen B. Sign language recognition using convolutional neural networks. In: Workshop at the European conference on computer vision, 2014. Springer. pp. 572–8.
170. Pisharady PK, Saerbeck M. Recent methods and databases in vision-based hand gesture recognition: a review. Comput Vis Image Underst. 2015;141:152–65.
171. Pisharady PK, Vadakkepat P, Loh AP. Attention based detection and recognition of hand postures against complex backgrounds. Int J Comput Vis. 2013;101(3):403–19.
172. Porfirio AJ, Wiggers KL, Oliveira LE, Weingaertner D. Libras sign language hand configuration recognition based on 3D meshes. In: Systems, man, and cybernetics (SMC), 2013 IEEE international conference on, 2013. IEEE. pp. 1588–93.
173. Powar V, Jahagirdar A, Sirsikar S. Skin detection in YCBCR color space. In: IJCA Proceedings of international conference on computational intelligence (ICCIA2012) ICCIA (5) (2012). Published by Foundations of Computer Science, New York, USA
174. Priyal SP, Bora PK. A study on static hand gesture recognition using moments. In: Signal processing and communications (SPCOM), 2010 international conference on, 2010. IEEE. pp. 1–5.
175. Pugeault N, Bowden R. Spelling it out: real-time ASL fingerspelling recognition. In: Computer vision workshops (ICCV workshops), 2011 IEEE international conference on, 2011. IEEE. pp. 1114–9.
176. Rabiner LR. A tutorial on hidden Markov models and selected applications in speech recognition. Proc IEEE. 1989;77(2):257–86.
177. Rautaray SS, Agrawal A. Vision based hand gesture recognition for human computer interaction: a survey. Artif Intell Rev. 2015;43(1):1–54.
178. Regenbrecht H, Collins J, Hoermann, S. A leap-supported, hybrid AR interface approach. In: Proceedings of the 25th Australian computer–human interaction conference: augmentation, application, innovation, collaboration, 2013. ACM. pp. 281–84.
179. Rehg JM, Kanade T. Model-based tracking of self-occluding articulated objects. In: Computer vision, 1995. Proceedings. Fifth international conference on, 1995. IEEE. pp. 612–7.
180. Reifinger S, Wallhoff F, Ablassmeier M, Poitschke T, Rigoll G. Static and dynamic hand-gesture recognition for augmented reality applications. In: International conference on human–computer interaction, 2007. Springer. pp. 728–37.
181. Ren Z, Yuan J, Meng J, Zhang Z. Robust part-based hand gesture recognition using kinect sensor. IEEE Trans Multim. 2013;15(5):1110–20.
182. Ren Z, Yuan J, Zhang, Z. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In: Proceedings of the 19th ACM international conference on multimedia, 2011. ACM. pp. 1093–6.
183. Rodriguez KO, Chavez GC. Finger spelling recognition from RGB-D information using kernel descriptor. In: Graphics, patterns and images (SIBGRAPI). IEEE; 2013. (2013 26th SIBGRAPI-conference on). pp. 1–7.
184. Rotem O, Greenspan H, Goldberger J. Combining region and edge cues for image segmentation in a probabilistic Gaussian mixture framework. In: Computer vision and pattern recognition, 2007. CVPR'07. IEEE conference on, 2007. IEEE. pp. 1–8.
185. Roy PP, Kumar P, Kim BG. An efficient sign language recognition (SLR) system using Camshift tracker and hidden Markov model (hmm). SN Comput Sci. 2021;2(2):1–15.
186. Ruffieux S, Lalanne D, Mugellini, E. Chairgest: a challenge for multimodal mid-air gesture recognition for close HCI. In: Proceedings of the 15th ACM on international conference on multimodal interaction, 2013. ACM. pp. 483–8.
187. Sagayam KM, Hemanth DJ. Hand posture and gesture recognition techniques for virtual reality applications: a survey. Virtual Real. 2017;21(2):91–107.
188. Salen K, Zimmerman E. Rules of play: game design fundamentals. Cambridge: The MIT Press; 2003.
189. Sandbach G, Zafeiriou S, Pantic M, Yin L. Static and dynamic 3D facial expression recognition: a comprehensive survey. Image Vis Comput. 2012;30(10):683–97.
190. Sarma D, Bhuyan MK. Hand gesture recognition using deep network through trajectory-to-contour based images. In: 15th IEEE India Council international conference (INDICON), 2018. pp. 1–6.
191. Sarma D, Bhuyan MK. Optical flow guided motion template for hand gesture recognition. In: Proceedings of the 2nd IEEE conference on applied signal processing (ASPCON), 2020.
192. Sarma D, Kavyasree V, Bhuyan M. Two-stream fusion model for dynamic hand gesture recognition using 3D-CNN and 2D-CNN optical flow guided motion template. 2020. arXiv:2007.08847
193. Sawicki DJ, Miziolek W. Human colour skin detection in CMVK colour space. IET Image Process. 2015;9(9):751–7.
194. Shamaie A, Sutherland A. Graph-based matching of occluded hand gestures. In: Applied imagery pattern recognition workshop. IEEE, 2001. pp. 67–73.
195. Shen X, Hua G, Williams L, Wu Y. Dynamic hand gesture recognition: an exemplar-based approach from motion divergence fields. Image Vis Comput. 2012;30(3):227–35.
196. Shin MC, Tsap LV, Goldgof DB. Gesture recognition using Bezier curves for visualization navigation from registered 3-D data. Pattern Recognit. 2004;37(5):1011–24.
197. Shin S, Kim WY. Skeleton-based dynamic hand gesture recognition using a part-based GRU-RNN for gesture-based interface. IEEE Access. 2020;8:50236–43.
198. Shotton J, Fitzgibbon A, Cook M, Sharp T, Finocchio M, Moore R, Kipman A, Blake A. Real-time human pose recognition in parts from single depth images. In: Computer vision and pattern recognition (CVPR), 2011 IEEE conference on, 2011. IEEE. pp. 1297–304.
199. Sigal L, Sclaroff S, Athitsos V. Estimation and prediction of evolving color distributions for skin segmentation under varying illumination. In: Proceedings of IEEE conference computer vision and pattern recognition, 2000, vol. 2, pp. 152–9.
200. Sigal L, Sclaroff S, Athitsos V. Skin color-based video segmentation under time-varying illumination. IEEE Trans Pattern Anal Mach Intell. 2004;26(7):862–77.

201. Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems, 2014. pp. 568–76.
202. Singha J, Laskar RH. Recognition of global hand gestures using self co-articulation information and classifier fusion. J Multimodal User Interfaces. 2016;10(1):77–93.
203. Sminchisescu C, Kanaujia A, Metaxas D. Conditional models for contextual human motion recognition. Comput Vis Image Underst. 2006;104(2–3):210–20.
204. Smith P, Shah M, da Vitoria Lobo N. Determining driver visual attention with one camera. IEEE Trans Intell Transp Syst. 2003;4(4):205–18.
205. Sobottka K, Pitas I. A novel method for automatic face segmentation, facial feature extraction and tracking. Signal Process Image Commun. 1998;12(3):263–81.
206. Song F, Tan X, Chen S, Zhou ZH. A literature survey on robust and efficient eye localization in real-life scenarios. Pattern Recognit. 2013;46(12):3157–73.
207. Song Y, Demirdjian D, Davis R. Tracking body and hands for gesture recognition: Natops aircraft handling signals database. In: Automatic face & gesture recognition and workshops (FG 2011), 2011 IEEE international conference on, 2011. IEEE. pp. 500–6.
208. Stern H, Efros B. Adaptive color space switching for tracking under varying illumination. Image Vis Comput. 2005;23(3):353–64. https://doi.org/10.1016/j.imavis.2004.09.005.
209. Störring M, Andersen HJ, Granum E. Skin colour detection under changing lighting conditions. In: Proceedings of 7th symposium intelligent & robotics systems, 1999. pp. 187–95.
210. Suau X, Alcoverro M, López-Méndez A, Ruiz-Hidalgo J, Casas JR. Real-time fingertip localization conditioned on hand gesture classification. Image Vis Comput. 2014;32(8):522–32.
211. Tan W, Wu C, Zhao S, Li J. Dynamic hand gesture recognition using motion trajectories and key frames. In: Advanced computer control (ICACC), 2010 2nd international conference on, 2010, vol. 3. IEEE. pp. 163–67.
212. Tang M. Recognizing hand gestures with microsoft's kinect. Palo Alto: Department of Electrical Engineering of Stanford University:[sn]; 2011.
213. Tao L, Zappella L, Hager GD, Vidal R. Surgical gesture segmentation and recognition. In: International conference on medical image computing and computer-assisted intervention, 2013. Springer. pp. 339–46.
214. Tompson J, Stein M, Lecun Y, Perlin K. Real-time continuous pose recovery of human hands using convolutional networks. ACM Trans Graphics. 2014;33(5):169.
215. Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE international conference on computer vision, 2015. pp. 4489–97.
216. Triesch J, Von Der Malsburg C. A system for person-independent hand posture recognition against complex backgrounds. IEEE Trans Pattern Anal Mach Intell. 2001;23(12):1449–53.
217. Tsironi E, Barros P, Wermter S. Gesture recognition with a convolutional long short-term memory recurrent neural network, vol. 2. Bruges, Belgium; 2016.
218. Utsumi A, Ohya J. Direct manipulation interface using multiple cameras for hand gesture recognition. In: Proceedings of IEEE international conference multimedia computer and systems, 1998. pp. 264–7.
219. Utsumi A, Tetsutani N, Igi S. Hand detection and tracking using pixel value distribution model for multiple-camera-based gesture interactions. In: Proceedings of IEEE workshop knowledge media network, 2002. pp. 31–36.
220. Várkonyi-Kóczy AR, Tusor B. Human-computer interaction for smart environment applications using fuzzy hand posture and gesture models. IEEE Trans Instrum Meas. 2011;60(5):1505–14.
221. Wan J, Ruan Q, Li W, Deng S. One-shot learning gesture recognition from RGB-D data using bag of features. J Mach Learn Res. 2013;14(1):2549–82.
222. Wang C, Liu Z, Chan SC. Superpixel-based hand gesture recognition with kinect depth camera. IEEE Trans Multim. 2015;17(1):29–39.
223. Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L, Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L. Temporal segment networks: towards good practices for deep action recognition. In: European conference on computer vision. Springer; 2016. pp. 20–36.
224. Wang X, Xia M, Cai H, Gao Y, Cattani C. Hidden-markov-models-based dynamic hand gesture recognition. Math Probl Eng. 2012;2012.
225. Wang X, Zhang X, Yao J. Skin color detection under complex background. In: Proceedings of international conference mechatronic science electric engineering and computing, 2011; MEC. pp. 1985–8.
226. Weston J, Watkins C. Multi-class support vector machines. Citeseer: technical report; 1998.
227. Wilbur R, Kak AC. Purdue RVL-SLLL American sign language database; 2006.
228. Wilson AD, Bobick AF. Learning visual behavior for gesture analysis. In: Computer vision, 1995. Proceedings. International symposium on, 1995. IEEE. pp. 229–34.
229. Wixson L. Detecting salient motion by accumulating directionally-consistent flow. IEEE Trans Pattern Anal Mach Intell. 2000;22(8):774–80.
230. Wu D, Zhu F, Shao L. One shot learning gesture recognition from RGBD images. In: Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on, 2012. IEEE. pp. 7–12.
231. Wu X, Mao X, Chen L, Xue Y, Rovetta A. Point context: an effective shape descriptor for RST-invariant trajectory recognition. J Math Imaging Vis. 2016;56(3):441–54.
232. Xu H, Li L, Fang M, Zhang F. Movement human actions recognition based on machine learning. Int J Online Biomed Eng. 2018;14(04):193–210.
233. Yacoob Y, Davis LS. Recognizing human facial expressions from long image sequences using optical flow. IEEE Trans Pattern Anal Mach Intell. 1996;18(6):636–42.
234. Yamato J, Ohya J, Ishii K. Recognizing human action in time-sequential images using hidden Markov model. In: Computer vision and pattern recognition, 1992. Proceedings CVPR'92. 1992 IEEE computer society conference on, 1992. IEEE. pp. 379–85.
235. Yang C, Han DK, Ko H. Continuous hand gesture recognition based on trajectory shape information. Pattern Recognit Lett. 2017;99:39–47.
236. Yang G, Li H, Zhang L, Cao Y. Research on a skin color detection algorithm based on self-adaptive skin color model. In: Proceedings of international conference communication intelligence information security (ICCIIS), 2010. pp. 266–70.
237. Yang J, Lu W, Waibel A. Skin-color modeling and adaptation. In: Asian conference on computer vision. Springer; 1998. pp. 687–94.
238. Yang MH, Ahuja N. Gaussian mixture model for human skin color and its applications in image and video databases. In: Storage and retrieval for image and video databases VII, 1998, vol. 3656. International Society for Optics and Photonics. pp. 458–67.

239. Yang MH, Ahuja N, Tabb M. Extraction of 2D motion trajectories and its application to hand gesture recognition. IEEE Trans Pattern Anal Mach Intell. 2002;24(8):1061–74.
240. Yang R, Sarkar S, Loeding B. Handling movement epenthesis and hand segmentation ambiguities in continuous sign language recognition using nested dynamic programming. IEEE Trans Pattern Anal Machine Intell. 2010;32(3):462–77.
241. Yin Z, Collins R. Moving object localization in thermal imagery by forward-backward MHI. In: 2006 conference on computer vision and pattern recognition workshop (CVPRW'06), 2006. IEEE. p. 133.
242. Yoon HS, Soh J, Bae YJ, Yang HS. Hand gesture recognition using combined features of location, angle and velocity. Pattern Recognit. 2001;34(7):1491–501.
243. Yuan S, Ye Q, Stenger B, Jain S, Kim TK. Bighand2. 2m benchmark: hand pose dataset and state of the art analysis. In: Computer vision and pattern recognition (CVPR), 2017 IEEE conference on, 2017. IEEE. pp. 2605–13.
244. Zeng B, Wang G, Lin X. A hand gesture based interactive presentation system utilizing heterogeneous cameras. Tsinghua Sci Technol. 2012;17(3):329–36.
245. Zhang E, Xue B, Cao F, Duan J, Lin G, Lei Y. Fusion of 2D CNN and 3D densenet for dynamic gesture recognition. Electronics. 2019;8(12):1511.
246. Zhang MJ, Gao W. An adaptive skin color detection algorithm with confusing backgrounds elimination. In: Proceedings of IEEE international conference image processing (ICIP), 2005'2. pp. II-390-3.
247. Zhang R, Ming Y, Sun J. Hand gesture recognition with surf-bof based on gray threshold segmentation. In: Signal processing (ICSP), 2016 IEEE 13th international conference on. IEEE, 2016. pp. 118–22.
248. Zhang Z. Microsoft kinect sensor and its effect. IEEE Multimed. 2012;19(2):4–10.
249. Zhao R, Ali H, Van der Smagt P. Two-stream RNN/CNN for action recognition in 3D videos. In: 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS), 2017. IEEE. pp. 4260–267.
250. Zhu Y, Lan Z, Newsam S, Hauptmann A. Hidden two-stream convolutional networks for action recognition. In: Asian Conference on Computer Vision, 2018;363–378. Springer