Analysis of the hands in egocentric vision: A survey

Andrea Bandini, *Member, IEEE,* and José Zariffa, *Senior Member, IEEE*

Abstract—Egocentric vision (a.k.a. first-person vision – FPV) applications have thrived over the past few years, thanks to the availability of affordable wearable cameras and large annotated datasets. The position of the wearable camera (usually mounted on the head) allows recording exactly what the camera wearers have in front of them, in particular hands and manipulated objects. This intrinsic advantage enables the study of the hands from multiple perspectives: localizing hands and their parts within the images; understanding what actions and activities the hands are involved in; and developing human-computer interfaces that rely on hand gestures. In this survey, we review the literature that focuses on the hands using egocentric vision, categorizing the existing approaches into: localization (where are the hands or parts of them?); interpretation (what are the hands doing?); and application (e.g., systems that used egocentric hand cues for solving a specific problem). Moreover, a list of the most prominent datasets with hand-based annotations is provided.

Index Terms—Egocentric vision, Computer vision, Hand detection, Hand segmentation, Hand pose estimation, Hand gesture recognition, Grasp, Action recognition, Activity recognition, Human computer interaction.

1 Introduction

The hands are of primary importance for human beings, as they allow us to interact with objects and environments, communicate with other people, and perform activities of daily living (ADLs) such as eating, bathing, and dressing. It is not a surprise that in individuals with impaired or reduced hand functionality (e.g., after a stroke or cervical spinal cord injury – cSCI) the top recovery priority is to regain the function of the hands [1]. Given their importance, computer vision researchers have tried to analyze the hands from multiple perspectives: localizing them in the images [2], inferring the types of actions they are involved in [3], as well as enabling interactions with computers and robots [4], [5]. Wearable cameras (e.g., cameras mounted on the head or chest) have allowed studying the hands from a point of view (POV) that provides a first-person perspective of the world. This field of research in computer vision is known as egocentric or first-person vision (FPV). Although some studies were published as early as the 1990s [6], FPV gained more importance after 2012 with the emergence of smart glasses and action cameras (i.e., Google Glass and GoPro cameras). For an overview of the evolution of FPV methods, the reader is referred to the survey published by Betancourt et al. [7].

Egocentric vision presents many advantages when compared with third person vision, where the camera position is usually stable and disjointed from the user. For example: the device is recording exactly what the users have in front of them; camera movement is driven by the camera-wearer's activity and attention; hands and manipulated objects tend to appear at the center of the image and hand occlusions are minimized [8]. These advantages made the development of novel approaches for studying the hands very appealing. However, when working in FPV, researchers must also face an important issue: the camera is not stable, but is moving with the human body. This causes fast movements and sudden illumination changes that can significantly reduce the quality of the video recordings and make it more difficult to separate the hand and objects of interest from the background.

Betancourt et al. [9] clearly summarized the typical processing steps of hand-based methods in FPV. The authors proposed a unified and hierarchical framework where the lowest levels of the hierarchy concern the detection and segmentation of the hands, whereas the highest levels are related to interaction and activity recognition. Each level is devoted to a specific task and provides the results to higher levels (e.g., hand identification builds upon hand segmentation and hand detection, activity recognition builds upon the identification of interactions, etc.). Although clear and concise, this framework could not cover some of the recent developments in this field, made possible thanks to the availability of large amounts of annotated data and to the advent of deep learning [10], [11], [12]. Other good surveys closely related to the topics discussed in our paper were published in the past few years [4], [8], [13], [14], [15], [16]. The reader should refer to the work of Del Molino et al. [15] for an introduction into video summarization in FPV, to the survey of Nguyen et al. [8] for the recognition of ADLs from egocentric vision, and to the work of Bolaños et al. [16] for a review on visual lifelogging. Hand pose estimation and hand gesture recognition methods are analyzed in [14] and [13], respectively.

In this survey we define a comprehensive taxonomy of hand-based methods in FPV expanding the categorization proposed in [9] and classifying the existing literature into three macro-areas: localization, interpretation, and application. For each macro-area we identify the main sub-areas of research, presenting the most prominent approaches published in the past 10 years and discussing advantages and disadvantages of each method. Moreover,

- *A. Bandini and J. Zariffa are with KITE – Toronto Rehab – University Health Network, Toronto, ON, CA.*
- *J. Zariffa is also with the Institute of Biomaterials and Biomedical Engineering (IBBME), University of Toronto, Toronto, ON, CA, the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, CA, and the Rehabilitation Sciences Institute, University of Toronto, Toronto, ON, CA.*
  E-mail: andrea.bandini@uhn.ca

we summarize the available datasets published in this field. Our focus in defining a comprehensive taxonomy and comparing different approaches is to propose an updated and general framework of hand-based methods in FPV, highlighting the current trends and summarizing the main findings, in order to provide guidelines to researchers who want to improve and expand this field of research. The remainder of the paper is organized as follows: Section 2 presents a taxonomy of hand-based methods in FPV following a novel categorization that divides these approaches into three macro-areas: localization, interpretation, and application; Section 3 describes the approaches developed for solving the localization problem; Section 4 summarizes the work focused on interpretation; Section 5 summarizes the most important applications of hand-based methods in FPV; Section 6 reviews the available datasets published so far; and, finally, Section 7 concludes with a discussion of the current trends in this field.

2 Hand-based methods in FPV – An updated framework

Starting from the raw frames, the first processing step is dedicated to the localization of the hands or parts of them within the observed scene. This allows restricting the processing to small regions of interest (ROIs), excluding unnecessary information from the background, or reducing the dimensionality of the problem, by extracting the articulated hand pose. Once the positions of the hands and/or their joints have been determined, higher-level information can be inferred to understand what the hands are doing (e.g. gesture and posture recognition, action and activity recognition). This information can be used for building applications such as human-computer interaction (HCI) and human-robot interaction (HRI) [4], [5]. Therefore, we categorize the existing studies that made use of hand-based methods in FPV into three macro-areas:

- **Localization** – approaches that answer the question: **where** are the hands (or parts of them)?
- **Interpretation** – approaches that answer the question: **what** are the hands doing?
- **Application** – approaches that use methods from the above areas to build real-world applications.

For each area we define sub-areas according to the aims and nature of the proposed methods.

2.1 Localization – Where are the hands (or parts of them)?

The localization area encloses all the approaches that aim at localizing hands (or parts of them) within the images. The sub-areas are:

- **Hand segmentation** – detecting the hand regions with pixel-level detail.
- **Hand detection** – defined both as binary classification problem (does the image contain a hand?) and object localization problem (is there a hand? Where is it located?). The generalization of hand detection over time is **hand tracking**.
- **Hand identification** – classification between left and right hand, as well as other hands present in the scene.
- **Hand pose estimation** – estimation of hand joint positions. A simplified version of the hand pose estimation problem is **fingertip detection**, where only the fingertips of one or more fingers are identified.

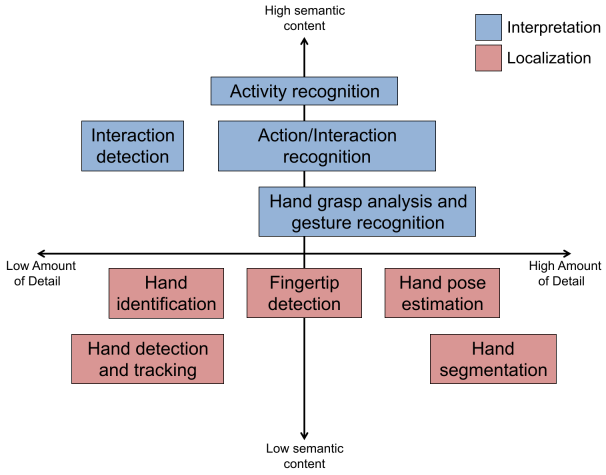


Fig. 1. Hand-based approaches in FPV categorized by amount of detail and semantic content.

From the above sub-areas it is possible to highlight two dimensions in the localization problem. The first one is the amount of detail of the information extracted with a method. For example, hand detection results in low-detail information (i.e., binary label or coordinates of a bounding box), whereas hand segmentation produces high-detail information (i.e., pixel-level silhouette). The second dimension is the meaning of the obtained information [8], [17], hereafter called semantic content. Hand detection and segmentation, although producing different amounts of detail, have the same semantic content, namely the global position of the hand. By contrast, hand pose estimation has higher semantic content than hand detection, as the position of the fingers and hand joints add more information to the global hand location. This categorization is shown in figure 1.

2.2 Interpretation – What are the hands doing?

The interpretation area includes those approaches that, starting from lower level information (i.e., detection, segmentation, pose estimation, etc.), try to infer information with higher semantic content. The main sub-areas are:

- **Hand grasp analysis** – Detection of the dominant hand postures during hand-object interactions.
- **Hand gesture recognition** – Classification of hand gestures, usually as input for virtual reality (VR) and augmented reality (AR) systems, as will be discussed in Section 5.
- **Action/Interaction recognition** – Predicting what type of action or interaction the hands are involved in. Following the taxonomy of Tekin et al. [18], an action is defined as a verb (e.g. "pour"), whereas an interaction as a verb-noun pair (e.g. "pour water"). This task is called *interaction detection* if the problem is reduced to a binary classification task (i.e., predicting whether or not the hands are interacting).
- **Activity recognition** – Identification of the activities, defined as set of temporally-consistent actions [3]. For example, preparing a meal is an activity composed of several actions and interactions, such as cutting vegetables, pouring water, opening jars, etc.

We can qualitatively compare these sub-areas according to the two dimensions described above (i.e., amount of detail and semantic content). Hand grasp analysis and gesture recognition have lower

semantic content than action/interaction recognition that, in turn has lower semantic content than activity recognition. Activity recognition, although with higher semantic content than action recognition, produces results with lower detail. This is because the information is summarized towards the upper end of the semantic content dimension. Following these considerations, we represent the localization and interpretation areas of this framework on a two-dimensional plot whose axes are the amount of detail and the semantic content (see Figure 1).

2.3 Application

The application area includes all the FPV approaches and systems that make use of hand-based methods for achieving certain objectives. The main applications are:

- Healthcare application, for example the remote assessment of hand function and the development of ambient assisted living (AAL) systems.
- HCI and HRI, for example VR and AR applications, or HRI systems that rely on the recognition of hand gestures.

Some egocentric vision applications were already covered by other surveys [8], [15], [16], [4]. Thus, we will summarizing novel aspects related to hand-based methods in FPV not covered in the previous articles.

3 Localization

The localization of hands (or parts of them) is the first and most important processing step of many hand-based methods in FPV. A good hand localization algorithm allows estimating the accurate position of the hands within the image, boosting the performance of higher-level inference [19]. For this reason, hand localization has been the main focus of researchers in egocentric vision. Although many hand detection, pose-estimation, and segmentation algorithms were developed in third person vision [14], the egocentric POV presents notable challenges that do not allow a direct translation of these methods. Rogez et al. [20] demonstrated that egocentric hand detection is considerably harder in FPV, and methods developed specifically for third person POV may fail when applied to egocentric videos.

Hand segmentation and detection are certainly the two most extensively studied sub-areas. They are often used in combination, for example to classify as "hand" or "not hand" previously segmented regions [21], [22], or to segment ROIs previously obtained with a hand detector [19]. However, considering the extensive research behind these two sub-areas, we summarize them separately.

3.1 Hand segmentation

Hand segmentation is the process of identifying the hand regions at pixel-level (see Figure 2). This step allows extracting the silhouette of the hands and has been extensively used as a preprocessing step for hand pose estimation, hand-gesture recognition, action/interaction recognition, and activity recognition. One of the most straightforward approaches is to use the color as discriminative feature to identify skin-like pixels [23]. Although very simple and fast, color-based segmentation fails whenever background objects have similar skin color (e.g., wooden objects) and it is robust only if the user wears colored gloves or patches to simplify the processing [24], [25]. However, this might not be feasible in real-world applications, where the hand segmentation algorithm is supposed to work without external cues, thus mimicking human vision. Illumination changes due to different environments also negatively affect the segmentation performance. Moreover, the availability of large datasets with pixel-level ground truth annotations is another issue when working with hand segmentation. This type of annotation requires a lot of manual work and the size of these datasets is much smaller than those with less detailed annotations (e.g., bounding boxes). Thus, several approaches were proposed to face the above issues.

3.1.1 Discriminating hands from objects and background

Traditional hand segmentation approaches (i.e., not based on deep learning) rely on the extraction of features from an image patch, classifying the central pixel or the entire patch as skin or no-skin using a binary classifier or regression model. The vast majority of approaches combined color with gradient and/or texture features, whereas random forest has been the most popular classification algorithm [26]. The use of texture and gradient features allows capturing salient patterns and contours of the hands that, combined with the color features, help discriminate them from background and objects with similar color.

Pixel-based classification. Li and Kitani [2] tested different combinations of color (HSV, RGB, and LAB color spaces) and local appearance features (Gabor filters [27], HOG [28], SIFT [29], BRIEF [30], and ORB [31] descriptors) to capture local contours and gradients of the hand regions. Each pixel was classified as skin or no-skin using a random forest regression. When using color features alone, the LAB color space provided the best performance, whereas gradient and texture features, such as HOG and BRIEF, improved the segmentation performance when combined with the color information [2]. Zariffa and Popovic [32] used a mixture of Gaussian skin model with dilation and erosion morphological operators to detect a coarse estimate of the hand regions. The initial region was refined by removing small isolated blobs with texture different from the skin, by computing the Laplacian of the image within each blob. Lastly, pixel-level segmentation was achieved by backprojecting using an adaptively selected region in the colour space. In [33], the coarse segmentation obtained with a mixture of Gaussian skin model [23], [32] was refined by using a structured forest edge detection [34], specifically trained on available datasets [22], [35].

Patch-based classification. Other authors classified image patches instead of single pixels, in order to produce segmentation masks more robust to pixel-level noise [36], [37], [38], [39], [40]. Serra et al. [36] classified clusters of pixels (i.e., super-pixels) obtained with the simple linear iterative clustering (SLIC) algorithm [41]. For each super-pixel, they used a combination of color (HSV and LAB color spaces) and gradient features (Gabor filters and histogram of gradients) to train a random forest classifier. The segmentation was refined by assuming temporal coherence between consecutive frames and spatial consistency among groups of super-pixels. Similarly, Singh et al. [37] computed the hand binary mask by extracting texture and color features (Gabor filters with RGB, HSV, and LAB color features) from the super-pixels, whereas Urabe et al. [38] used the same features in conjunction with the centroid location of each super-pixel to train a support vector machine (SVM) for segmenting the skin regions. Instead of classifying the whole patch from which color, gradient and texture features are extracted, Zhu et al. [39], [40] learned the segmentation mask within the image patch, by using a random forest framework (i.e., shape-aware structured forest).

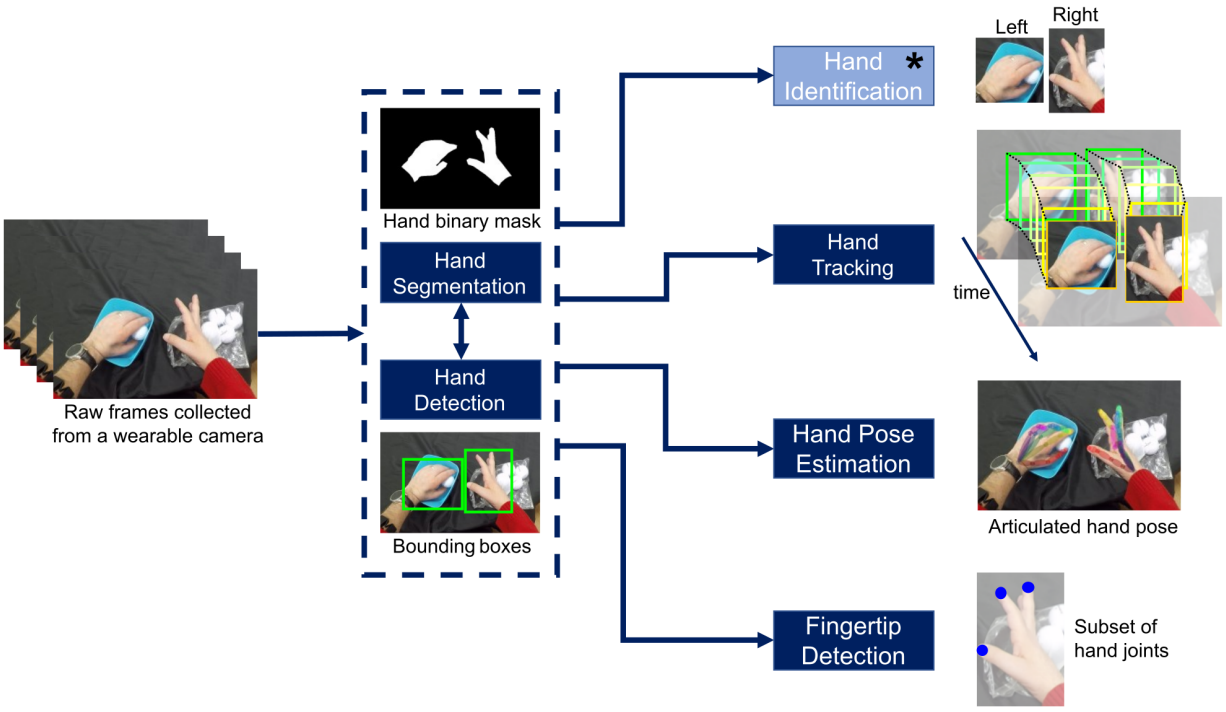


Fig. 2. Diagram of hand localization tasks in egocentric vision. Hand detection and segmentation have often been used in combination, for example to segment ROIs previously obtained with a hand detector, or to classify as "hand" or "not hand" previously segmented regions. Since they provide the global position of the hands within the frame, they are chosen as the basis for other localization approaches, such as hand identification, hand tracking, hand pose estimation, fingertip detection (*: Hand identification is now typically incorporated within the hand detection step).

Deep learning may help solve hand segmentation problems in FPV. However, its use is still hampered by the lack of large annotated datasets with pixel-level annotations. Some deep learning approaches for hand segmentation [42], [43] tackled this issue by using the available annotations in combination with other image segmentation techniques (e.g., super-pixels or GrabCut [41], [44], [45], [46]) to generate new hand segmentation masks for expanding the dataset and fine-tuning pre-trained networks (see Section 3.1.3 for more details). The availability of pre-trained convolutional neural networks (CNNs) for semantic object segmentation [47], [48] was exploited in [49], [50]. Wang et al. [51], [11] tackled the hand segmentation problem in a recurrent manner by using a recurrent U-NET architecture [52]. The rationale behind this strategy is to imitate the saccadic movements of the eyes that allow refining the perception of a scene. The computational cost can be another issue in CNN-based hand segmentation. To reduce this cost, while achieving good segmentation accuracy, Li et al. [53] implemented the deep feature flow (DFF) [54] with an extra branch to make the approach more robust against occlusions and distortion caused by DFF.

3.1.2 Robustness to illumination changes

The problem of variable illumination can be partially alleviated by choosing the right color-space for feature extraction (e.g., LAB [2]) and increasing the size of the training set. However, the latter strategy may reduce the separability of the color space and increase the number misclassified examples [55]. Thus, a popular solution has been to use a collection of segmentation models, adaptively selecting the most appropriate one for the current test conditions [2], [36], [37], [55], [56]. Li and Kitani [2] proposed an adaptive approach that selects the nearest segmentation model, namely the one trained in a similar environment. To learn different global appearance models, they clustered the HSV histogram of the training images using k-means and learned a separate random tree regressor for each cluster. They further extended this concept in [56] where they formulated hand segmentation as a model recommendation task. For each test image, the system was able to propose the best hand segmentation model given the color and structure (HSV histogram and HOG features) of the observed scene and the relative performance between two segmentation models. Similarly, Betancourt et al. [55] trained binary random forests to classify each pixel as skin or not skin using the LAB values. For each frame they trained a separate segmentation model storing it along with the HSV histogram, as a proxy to summarize the illumination condition of that frame. K-nearest neighbors (k-NN) classification was performed on the global features to select the k most suitable segmentation models. These models were applied to the test frame and their segmentation results combined together to obtain the final hand mask.

3.1.3 Lack of pixel-level annotations

Annotating images at pixel-level is a very laborious and costly work that refrains many authors from publishing large annotated datasets. Thus, the ideal solution to the hand segmentation problem would be a self-supervised approach able to learn the appearance of the hands on-the-fly, or a weakly supervised method that relies on the available training data to produce new hand masks.

Usually, methods for online hand segmentation made assumptions on the hand motion [57], [58], [59], [60] and/or required the user to perform a calibration with pre-defined hand movements [61]. In this way, the combination of color and motion

features facilitates the detection of hand pixels, in order to train segmentation models online. Kumar et al. [61] proposed an on-the-fly hand segmentation, where the user calibrated the systems by waving the hands in front of the camera. The combination of color and motion segmentation (Horn–Schunck optical flow [62]) and region growing, allowed locating the hand regions for training a GMM-based hand segmentation model. Region growing was also used by Huang et al. [57], [58]. The authors segmented the frames in super-pixels [41] and extracted ORB descriptors [31] from each super-pixel to find correspondences between regions of consecutive frames, which reflect the motion between two frames. Hand-related matches were distinguished from camera-related matches based on the assumption that camera-related matches play a dominant role in the video. These matches were estimated using RANSAC [63] and after being removed, those left were assumed to belong to the hands and used to locate the seed point for region growing. Zhao et al. [59], [60] based their approach on the typical motion pattern during actions involving the hands: preparatory phase (i.e., the hands move from the lower part of the frame to the image center) and interaction phase. During the preparatory phase they used a motion-based segmentation, computing the TV-L1 optical flow [64]. As the preparatory phase ends, the motion decreases and the appearance becomes more important. A super-pixel segmentation [41] was then performed and a super-pixel classifier, based on the initial motion mask, was trained using color and gradient features.

Transfer learning has also been used to deal with the paucity of pixel-level annotations. The idea is to exploit the available pixel-level annotations in combination with other image segmentation techniques (e.g., super-pixels or GrabCut [41], [44], [46]) to generate new hand segmentation masks and fine-tune pre-trained networks. Zhou et al. [42] trained a hand segmentation network using a large amount of bounding box annotations and a small amount of hand segmentation maps [65]. They adopted a DeconvNet architecture [66] made up of two mirrored VGG-16 networks [67] initialized with 1,500 pixel-level annotated frames from [65]. Their approach iteratively selected and added good segmentation proposals to gradually refine the hand map. The hand segmentation proposals were augmented by applying super-pixel segmentation [41] and Grabcut [46] to generate the hand probability map within the ground truth bounding boxes. DeconvNet was trained in an Expectation-Maximization manner: 1) keeping the network parameter fixed, they generated a set of hand masks and selected the best segmentation proposals (i.e., those with largest match with the ground truth mask); 2) they updated the network weights by using the best segmentation hypotheses. Similarly, Li et al. [43] relied on the few available pixel-level annotations to train Deeplab-VGG16 [68]. Their training procedure was composed of multiple steps: 1) Pre-segmentation – the CNN, pre-trained using the available pixel-level annotations, was applied on the target images to generate pre-segmentation maps; 2) Noisy mask generation – the pre-segmentation map was combined with a super-pixel segmentation [44]; and 3) Model retraining – the new masks were used as ground truth to fine tune the pre-trained Deeplab-VGG16.

3.1.4 Depth and 3D segmentation

The use of depth sensors or stereo cameras helps alleviate some of the aforementioned issues, in particular the robustness to illumination changes and lack of training data. However, the use of devices not specifically developed for wearable applications and their high power consumption [55] has limited their FPV application only to research studies.

Some authors used the depth information to perform a background/foreground segmentation followed by hand/object segmentation within the foreground region by using appearance information [69], [70], [71]. Wan et al. [69] used a time-of-flight (ToF) camera to capture the scene during hand-object interactions. They observed that the foreground (i.e., arm, hands, and manipulated objects) is usually close to the camera and well distinguishable, in terms of distance, from the background. Thus, after thresholding the histogram of depth values to isolate the foreground, hand pixels were detected by combining color and texture features (e.g., RGB thresholds and Gabor filters). The same ToF camera (Creative® Senz3D™) was used by Rogez et al. [70]. The authors trained a multi-class classifier on synthetic depth maps of 1,500 different hand poses, in order to recognize one of these poses in the test depth images, thus producing a coarse segmentation mask. This mask was then processed in a probabilistic manner to find the binary map corresponding to the hand pixels. Color cues were also used by computing RGB-based super-pixels on the test image. Yamazaki et al. [71] reconstructed the colored point cloud of the scene recorded with a Microsoft® Kinect™ v2. The foreground was isolated by detecting and removing large plane structures (i.e., likely belonging to the background) using RANSAC [63]. Afterwards, color segmentation was performed using a person-specific skin color model calibrated on the user's skin. Ren et al. [72] used a stereo camera to reconstruct the depth map of the scene. Specifically, the depth map was reconstructed using the scanline-based stereo matching and the hand was segmented only using depth information.

3.1.5 Remarks on hand segmentation

Because of the high amount of detail obtained with hand segmentation algorithms, this task is the hardest one among hand-based methods in FPV. The pixel- or super-pixel-level accuracy required for this task, combined with the intrinsic problems of egocentric vision, made this sub-area the most challenging and debated of this field of research. The effort of many researchers in finding novel and powerful approaches to obtain better results is justified by the possibility to improve not only the localization accuracy, but also to boost the performance of higher-level inference. In fact, it was demonstrated that a good hand segmentation mask can be sufficient for recognizing actions and activities involving the hands with high accuracy [19], [73]. For this reason, pixel-level segmentation has often been used as basis of higher-inference methods.

RGB-D information can certainly improve and simplify the hand segmentation task. However, these methods are a minority with respect to the 2D counterpart, since no depth cameras have been developed for specific egocentric applications. With the recent miniaturization of depth sensors (e.g., iPhone® X and 11) the 3D segmentation is still an area worth exploring and expanding within the next few years.

Many authors considered detection and segmentation as two steps of the same task. We preferred to split these two sub-areas given the large amount of work produced in the past few years. However, as it will be illustrated in the next section, many hand detection approaches, especially those using region-based CNNs, used the segmentation mask for generating region proposals. Perhaps, with the possibility to re-train powerful object detectors, this process has become inefficient and instead of having

a "detection over segmentation", it will be more convenient to have a "segmentation over detection", unless the specific problem calls for a pixel-level segmentation of the entire frame. Following the great success of mask R-CNN [74], an interesting approach in this direction would be to address hand segmentation as an instance segmentation task, embedding bounding box detection and semantic segmentation of the hands in a single model.

3.2 Hand detection and tracking

Hand detection is the process of localizing the global position of the hands at frame level. This task is usually performed by fitting a bounding box around the area where the hand has been detected (see Figure 2). Hand detection allows extracting coarser information than hand segmentation, although this lower detail is counterbalanced by higher robustness to noise. If the application does not require very detailed information, this is the most popular choice as basis for hand-based higher inference. In the literature we can distinguish two main approaches: hand detection as image classification task; and hand detection as object detection task. Furthermore, hand detection generalized over time is referred to as hand tracking.

3.2.1 Hand detection as image classification

Pixel-level segmentation of hand regions, if performed on the entire image, may be prone to high occurrence of false positives [9], [35]. In these cases, a pre-filtering step that prevents from processing frames without any hands is necessary. This approach allows determining whether an image contains hands and it is usually followed by a hand segmentation step responsible for locating the hand region [9], [32], [35], [59], [60].

In [32], the authors back-projected the frame using a histogram obtained from a mixture of Gaussian skin model [23], predicting the presence of hands within the image by thresholding the back-projected values. Betancourt et al. [35] proposed an approach based on HOG features and SVM classifier to predict the presence of hands at frame-level, reducing the number of false positives. However, this frame-by-frame filtering increased the risk of removing frames with barely visible hands, thus increasing the false negatives [35]. To solve this issue, the authors proposed a dynamic Bayesian network (DBN) to smooth the classification results of the SVM and improve the prediction performance [75]. Zhao et al. [59], [60] detected the presence of hands within each frame exploiting the typical interaction cycle of the hands (i.e., preparatory phase - interaction - hands out of the frame). Based on this observation, they defined an ego-saliency metric related to the probability of having hands within a frame. This metric was derived from the optical flow map calculated using [76] and was composed of two terms: spatial cue, which gives more weight to motion within the lower part of the image; and temporal cue, which takes into account whether the motion is increasing or decreasing between adjacent frames.

3.2.2 Hand detection as object detection

Hand detection performed within an object localization framework presents notable challenges. Given the flexibility, the continuous variation of poses, and the high number of degrees of freedom, the hand appearance is highly variable and classical object detection algorithms (e.g., Haar like features with adaboost classification) may work only in constrained situations, such as detection of hands in a specific pose [77]. For these reasons, and thanks to the availability of large annotated datasets with bounding boxes, this is the area that most benefited from the advent of deep learning.

Region-based approaches. Many authors proposed region-based CNNs to detect the hands, exploiting segmentation approaches summarized in Section 3.1 to generate region proposals. Bambach et al. [19], [73] proposed a probabilistic approach for region proposal generation that combined spatial biases (e.g., reasoning on the position of the shape of the hands from training data) and appearance models (e.g., non-parametric modeling of skin color in the YUV color space). To guarantee high coverage, they generated 2,500 regions for each frame that were classified using CaffeNet [78]. Afterwards, they obtained the hand segmentation mask within the bounding box, by applying GrabCut [46]. Zhu et al. [21] used a structured random forest to propose pixel-level hand probability maps. These proposals were passed to a multitask CNN to locate the hand bounding box, the shape of the hand within the bounding box, and the position of wrist and palm. In [22], the authors generated region proposals by segmenting skin regions with [2] and determining if the set of segmented blobs correspond to one or two arms. This estimation was performed by thresholding the fitting error of a straight line. K-means clustering, with $k = 2$ if two arms are detected, was applied to split the blobs into two separate structures. The hand proposals were selected as the top part of a rectangular bounding box fitted to the arm regions and passed to CaffeNet for the final prediction. To generate hand region proposals, Cruz et al. [79] used a deformable part model (DPM) to make the approach robust to different gestures. DPM learns the hand shape by considering the whole structure and its parts (i.e., the fingers) using HOG features. CaffeNet [78] was used for classifying the proposals. Faster R-CNN was used in [33], [80], [81]. In particular, Likitlersuang et al. [33] fine-tuned the network on videos from individuals with cSCI performing ADLs. False positives were removed based on the arm angle information computed by applying a Haar-like feature rotated 360 degrees around the bounding box centroid. The resulting histogram was classified with a random forest to determine whether the bounding box actually included a hand. Furthermore, they combined color and edge segmentation to re-center the bounding box, in order to promote maximum coverage of the hands while excluding parts of the forearm.

Regression-based approaches were also used for detecting the hands. Mueller et al. [82] proposed a depth-based approach for hand detection, implemented using the Intel® RealSense™ SR300 camera. A Hand Localization Network (HALNet – architecture derived from ResNet50 [83] and trained on synthesized data) was used to regress the position of the center of the hand. The ROI was then cropped around this point based on its distance from the camera (i.e., the higher the depth, the smaller the bounding box). Recently, the *You Only Look Once* (YOLO) detector [84] was applied for localizing hands in FPV [85], [86], [87], demonstrating better trade-off between computational cost and localization accuracy than Faster R-CNN and single-shot detector (SSD) [85], [86], [88].

3.2.3 Hand tracking

Hand tracking allows estimating the position of the hands across multiple frames, reconstructing their trajectories in time. Theoretically, every hand detection and segmentation approach seen above, with the exception of the binary classification algorithms of section 3.2.1, can be used as tracker as well, by performing a frame-by-frame detection. This is the most widely used choice for

tracking the hand position over time. However, some authors tried to combine the localization results with temporal models to predict the future hand positions. This strategy has several advantages, such as decreasing the computational cost by avoiding to run the hand detection every frame [85], disambiguate overlapping hands by exploiting their previous locations [89], [90], [87], and refining the hand location [91].

Lee et al. [89] studied the child-parent social interaction from the child's POV, by using a graphical model to localize the body parts (i.e., hands of child and parent, head of the parent). The model was composed of inter-frame links to enforce temporal smoothness of the hand positions over time, shift links to model the global shifts in the field of view caused by the camera motion, and intra-frame constraints based on the spatial configuration of the body parts. Skin color segmentation in the YUV color space was exploited to locate the hands and define intra-frame constraints on their position. This formulation forced the body parts to remain in the neighborhood of the same position between two consecutive frames, while allowing for large displacement due to global motion (i.e., caused by head movements) if this displacement is consistent with all parts. Liu et al. [91] demonstrated that the hand detection is more accurate in the central part of the image due to a center bias (i.e., higher number of training examples with hands in the center of the frame). To correct this bias and obtain homogeneous detection accuracy in the whole frame, they proposed an attention-based tracker (AHT). For each frame, they estimated the target location of the hand by exploiting the result at the previous frame. Then, the estimated hand region was translated to the image center, where a CNN fine-tuned on frames with centralized hands was applied. After segmenting the hand regions using [2], Cai et al. [90] used the temporal tracking method [92] to discriminate them in case of overlap.

Regression-based CNNs in conjunction with object tracking algorithms were used in [87], [85]. Kapidis et al. [87] fine-tuned YOLOv3 [93] on multiple datasets to perform the hand detection, discriminating the right and left hand trajectories over time using the simple online real-time tracking (SORT) [94]. For each detected bounding box, this algorithm allowed predicting its next position, also assigning it to existing tracks or to new ones. Visée et al. [85] combined hand detection and tracking to design an approach for fast and reliable hand localization in FPV. Motivated by the slow detection performance of YOLOv2 without GPU, they proposed to combine YOLOv2 with the Kernelized Correlation Filter (KCF) [95] as a trade-off between speed and accuracy. The authors used the detector to automatically initialize and reset the tracker in case of failure or after a pre-defined number of frames.

3.2.4 Remarks on hand detection and tracking

Hand detection and segmentation are two closely related tasks that can be combined together. If hand detection is performed using a region-based approach (e.g., Faster R-CNN), hand segmentation can be seen as the pre-processing step of the localization pipeline, whereas in case of regression-based CNNs (e.g., YOLO) hand segmentation may follow the bounding box detection. The higher performance of regression-based methods with respect to region-based CNNs [86], [85] makes the latter approach more appealing in view of optimizing the hand localization pipeline. If there is no need of segmenting the hands at pixel level, the segmentation can just be skipped, whereas in problems where detailed hand silhouettes are needed, hand segmentation can be applied only within the detected the ROI, avoiding unnecessary computation.

The combination of detection and tracking algorithm may help to speed-up the localization performance with the possibility of translating these approaches into real-world application where low resource hardware is the only available option [85]. Moreover, as we will show in Section 4, hand tracking is an important step for the characterization and recognition of dynamic hand gestures [96], [97].

3.3 Hand identification

Hand identification is the process of disambiguating the left and right hands of the camera wearer, as well as the hands of other persons in the scene. The egocentric POV has intrinsic advantages that allow discriminating the hands by using simple spatial and geometrical constraints [33], [89], [98]. Usually, the user's hands appear in the lower part of the image, with the right hand to the right of the user's left hand, and vice versa. By contrast, other people's hands tend to appear in the upper part of the frame [89]. The orientation of the arm regions was used in [33], [98] to distinguish the left from the right user's hand. To estimate the angle, the authors rotated a Haar-like feature around the segmented hand region, making this approach robust to the presence of sleeves and different skin colors, since it did not require any calibrations [98]. To identify the hands, they split the frame into four quadrants. The quadrant with the highest sum of the Haar-like feature vector determined the hand type: "user's right" if right lower quadrant; "user's left" if left lower quadrant; "other hands" if upper quadrants [33]. The angle of the forearm/hand regions was also used by Betancourt et al. [55], [99]. The authors fitted an ellipse around the segmented region, calculating the angle between the arm and the lower frame border and the normalized distance of the ellipse center from the left border. The final left/right prediction was the result of a likelihood ratio test between two Maxwell distributions. Although simple and effective, spatial and geometric constraints may fail in case of overlapping hands. In this case, the temporal information help disambiguate the hands [90], [87]. Cai et al. [90] were interested in studying the grasp of the right hand. After segmenting the hand regions [2], they implemented the temporal tracking method proposed in [92] to handle the case of overlapping hands, thus tracking the right hand. Kapidis et al. [87] used the SORT tracking algorithm [94]. This approach combines the Kalman filter to predict the future position of the hand and the Hungarian algorithm to assign the next detection to existing tracks (i.e., left/right) or new ones.

With the availability of powerful and accurate CNN-based detectors, the hand identification as separated processing step is deprecated, being incorporated within hand detection (see Section 3.2.2) [19], [80], [85], [86]. To this end, both region-based (e.g., Faster R-CNN) and regression-based methods (e.g., YOLO and SSD) have been used. These models were trained or fine-tuned to recognize two or more classes of hands, predicting the bounding box coordinates along with its label (i.e., left, right, and other hands) [85], [86].

3.4 Hand pose estimation and fingertip detection

Hand pose estimation consists in the localization of the hand parts (e.g., the hand joints) to reconstruct the articulated hand pose from the images (see Figure 2). The possibility to obtain the position of fingers, palm, and wrist, simplifies higher inference tasks such as grasp analysis and hand gesture recognition, since

the dimensionality of the problem is reduced yet keeping high-detail information. An important difficulty in hand pose estimation lies in object occlusions and self-occlusions that make it hard to localize hidden joints/parts of the hand. Some authors proposed the use of depth cameras in conjunction with sensor-based techniques to train hand pose estimators more robust to self-occlusions [20], [100], [82], [71], [12]. However, as discussed above, the use of RGB-D imaging techniques might not be easily translated to FPV. Thus, several attempts have also been made to estimate the hand pose using only color images [24], [40], [38], [101], [18]. In this section, we summarize the previous work distinguishing between hand pose estimation approaches with depth sensors and hand pose estimation using monocular color images. Moreover, we summarize approaches for fingertip detection, which can be seen as an intermediate step between hand detection and hand pose estimation.

3.4.1 Hand pose estimation using depth sensors

One of the advantages of using depth information for estimating the hand pose is that it is easier to synthesize depth maps that closely resemble the ones acquired by real sensors, when compared to real versus synthetic color images [20], [100]. In [20], the authors tackled hand pose estimation as a multiclass classification problem by using a hierarchical cascade architecture. The classifier was trained on synthesized depth maps by using HOG features and tested on depth maps obtained with a ToF sensor. Instead of estimating the joint coordinates independently, they predicted the hand pose as whole, in order to make the system robust to self-occlusions. Similarly, in [100], the authors predicted the upper arm and hand poses simultaneously, by using a multiclass linear SVM for recognizing K poses from depth data. However, instead of classifying scanning windows on the depth maps, they classified the whole egocentric work-space, defined as the 3D volume seen from the egocentric POV. Mueller et al. [82] proposed a CNN architecture (Joint Regression Net – JORNet) to regress the 3D locations of the hand joints within the cropped colored depth maps captured with a structured light sensor (Intel® RealSense™ SR300). Afterwards, a kinematic skeleton was fitted to the regressed joints, in order to refine the hand pose. Yamazaki et al. [71] estimated the hand pose from hand point clouds captured with the Kinect v2 sensor. The authors built a dataset by collecting pairs of hand point clouds and ground truth joint positions obtained with a motion capture system. The pose estimation was performed by aligning the test point cloud to the training examples and predicting its pose as the one that minimizes the alignment error. The sample consensus initial alignment [102] and iterative closest point algorithms [103] were used for aligning the point clouds. Garcia-Hernando et al. [12] evaluated a CNN-based hand pose estimator [104] for regressing the 3D hand joints from RGB-D images recorded with the Intel® RealSense™ SR300 camera. The authors demonstrated that state-of-the-art hand pose estimation performance can be reached by training the algorithms on datasets that include hand-object interactions, in order to improve its robustness to self-occlusions or hand-object occlusions.

3.4.2 Hand pose estimation from monocular color images

In general, hand pose estimation from monocular color images allows locating the parts of the hands either in the form of 2D joints or semantic sub-regions (e.g., fingers, palm, etc.). This estimation is performed within previously detected ROIs, obtained by either a hand detection or segmentation algorithm. Liang et al. [24] used a conditional regression forest (CRF) to estimate the hand pose from hand binary masks. Specifically, they trained a set of pose estimators separately, conditioned on different distances from the camera, since the hand appearance and size can change dramatically with the distance from the camera. Thus, they synthesized a dataset in which the images were sampled at discretized intervals. The authors also proposed an intermediate step for improving the joint localization, by segmenting the binary silhouette into twelve semantic parts corresponding to different hand regions. The semantic part segmentation was performed with a random forest for pixel-level classification exploiting binary context descriptors. Similarly, Zhu et al. [40] built a structured forest to segment the hand region into four semantic sub-regions: thumb, fingers, palm, and forearm. This semantic part segmentation was performed extending the structured regression forest framework already used for hand segmentation (as discussed in Section 3.1) to a multiclass problem [24].

Other studies adapted CNN architectures developed for human pose estimation (e.g., OpenPose [105], [106]) for solving the hand pose estimation problem [38], [101] and localizing 21 hand joints. Tekin et al. [18] used a fully convolutional network (FCN) architecture to simultaneously estimate the 3D hand and object pose from RGB images. For each frame, the FCN produced a 3D discretized grid. The 3D location of the hand joints in camera coordinate system was then estimated combining the predicted location within the 3D grid and the camera intrinsic matrix.

3.4.3 Fingertip detection

Fingertip detection can be seen as an intermediate step between hand detection and hand pose estimation. Unlike pose estimation, only the fingertips of one or multiple fingers are detected. These key-points alone do not allow reconstructing the articulated hand pose, but can be used as input to HCI/HRI systems [107], [108], [109], as will be discussed in Section 5. If the objective is to estimate the joints of a single finger, the most common solution is to regress the coordinates of these points (e.g., the tip and knuckle of the index finger) from a previously detected hand ROI. This approach has been exploited in [91], [110]. The cropped images, after being resized, were passed to a CNN to regress the location of the key-points [110]. However, since the fingertip often lies at the border of the hand bounding box, the hand detection plays a significant role, and inaccurate detections greatly affect the fingertip localization result [91]. Wu et al. [111] extended the fingertip detection problem to the localization of the 5 fingertips of a hand. They proposed a heatmap-based FCN that, given the detected hand area, produced a 5-channel image containing the estimated likelihood of each fingertip at each pixel location. The maximum of each channel was used to predict the position of the fingertips.

3.4.4 Remarks on hand pose estimation

Among the hand localization tasks, hand pose estimation allows obtaining high-detail information with high semantic content at the same time (see Figure 1). This task, if performed correctly, can greatly simplify higher inference steps (e.g., hand gesture recognition and grasp analysis), but may be more prone to low robustness against partial hand occlusions.

Compared to other localization tasks, hand pose estimation presents a higher proportion of approaches that use depth sensors. This choice has several advantages: 1) the possibility to use motion

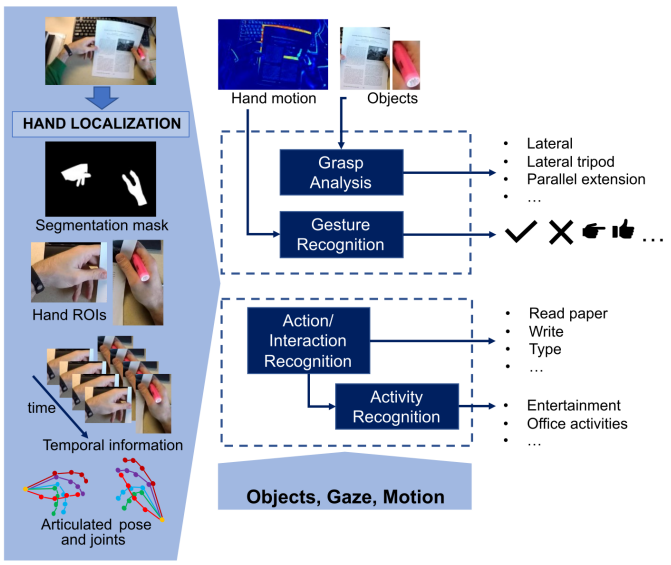


Fig. 3. Diagram of the hand interpretation areas in egocentric vision. Grasp analysis and gesture recognition focus directly on describing the hand. In action/interaction and activity recognition, the hand is instrumental in describing the user's behaviour.

capture methods for automatically obtaining the ground truth joint positions [71], [104]; 2) the availability of multiple streams (i.e., color and depth) that can be combined to refine the estimations [82], [108]; and 3) the possibility to synthesize large datasets of realistic depth maps [20], [100]. In the past few years, human pose estimation approaches [105], [106] have been successfully adapted to the egocentric POV, in order to estimate the hand and arm pose from monocular color images [38], [101]. This opens new possibilities to streamline and improve the performance of localization and hand-based higher inference tasks, such as grasp analysis. To further facilitate the adaptation of existing pose estimation approaches, large annotated datasets with hand joint information are needed. To this end, a combination of 2D and 3D information may be beneficial, in order to get accurate and extensive ground truth annotations in 3D that will allow solving the occlusion problems even when using color images alone.

4 Interpretation

After the hands have been localized within the images, higher-level inference can be conducted in the ROIs. This processing is usually devoted to the interpretation of gestures and actions of the hands that, in turn, can be used as cues for hand-based applications such as HCI and HRI, as will be discussed in Section 5. Based on the literature published so far, hand-based interpretation approaches in FPV can be divided into hand grasp analysis, hand gesture recognition, action/interaction recognition, and activity recognition (see Figures 1 and 3).

4.1 Hand grasp analysis and gesture recognition

According to Feix et al. [112], "A grasp is every static hand posture with which an object can be held securely with one hand, irrespective of the hand orientation". The recognition of the grasp types allows determining the different ways with which humans use their hands to interact with objects [113]. The common grasp modes can be used to describe hand-object manipulations, reducing the complexity of the problem, since the set of possible grasps is typically smaller than the set of possible hand shapes [112]. Moreover, the identification of the most recurrent grasp types has important applications in robotics, biomechanics, upper limb rehabilitation, and HCI. Thus, several taxonomies were proposed in the past decades [112], [114], [115], [116], [117], [118], [119]. For a comprehensive comparison among these taxonomies, the reader is referred to [112]. The analysis of hand grasps conducted via manual annotations is a lengthy and costly process. Thus, the intrinsic characteristics of egocentric vision allowed developing automated methods to study and recognize different grasp types, saving a huge amount of manual labor. Although in most cases the hand grasp analysis has been addressed in a supervised manner (i.e., grasp recognition – Section 4.1.1) [70], [90], [120], [121], [122], [123], some authors proposed to tackle this problem using clustering approaches, in order to discover dominant modes of hand-object interaction and identify high-level relationships among clusters (i.e., grasp clustering and abstraction – Section 4.1.2) [90], [113], [120], [124].

Similar to grasp analysis, hand gesture recognition aims at recognizing the semantic of the hand's posture and it is usually performed as input to HCI/HRI systems. However, two main differences exist between these two topics: 1) Grasp analysis looks at the hand posture during hand-object manipulations, whereas hand gesture recognition is usually performed on hands free of any manipulations; 2) grasp analysis aims at recognizing only static hand postures [112], whereas hand gesture recognition can also be generalized to dynamic gestures. According to the literature, hand gestures can be static or dynamic [72]: static hand gesture recognition (see Section 4.1.3) aims at recognizing gestures that do not depend on the motion of the hands, thus relying on appearance and hand posture information only [36], [72], [107], [125], [126], [127], [10]; dynamic hand gesture recognition (see Section 4.1.4) is performed using temporal information (e.g., hand tracking), in order to capture the motion cues that allow generating specific gestures [126], [10], [128], [129], [96], [97].

4.1.1 Hand grasp recognition

Supervised approaches for grasp recognition are based on the extraction of features from previously segmented hand regions [2] and their multiclass classification following one of the taxonomies proposed in the literature [112].

Cai et al. [120] used HOG features to represent the shape of the hand and a combination of HOG and SIFT to capture the object context during the manipulation. These features were classified with a multi-class SVM using a subset of grasp types from Feix's taxonomy [112], [114]. The authors extended their approach in [90], [121] by introducing CNN-based features extracted from the middle layers of [130] and features derived from the dense hand trajectory (DHT) [131] such as the displacement, gradient histograms, histogram of optical flow, and motion boundary histograms. The superior performance of CNN- and DHT-based features and their robustness across different tasks and users [90] suggested that high-level feature representation and motion and appearance information in the space-time volume may be important cues for discriminating different hand configurations. In [123], the authors used a graph-based approach to discriminate 8 grasp types. Specifically, the binary hand mask was used to produce a graph structure of the hand with an instantaneous topological map neural network. The eigenvalues of the graph's Laplacians were

used as features to represent the hand configurations, which were recognized using an SVM.

The use of depth sensors was explored by Rogez et al. [70]. The authors recognized 71 grasp types [119] using RGB-D data, by training a multi-class SVM with deep-learned features [67] extracted from both real and synthetic data. Moreover, the grasp recognition results were refined by returning the closest synthetic training example, namely the one that minimized the distance with the depth of the detected hand region.

4.1.2 Hand grasp clustering and abstraction

The first attempt to discover hand grasps in FPV was [113]. HOG features were extracted from previously segmented hand regions and grouped by means of a two-stage clustering approach. First, a set of candidate cluster centers was generated through the fast determinantal point process (DPP) algorithm [132]. This step allowed generating a wide diversity of clusters to cover many possible hand configurations. Secondly, each segmented region was assigned to the nearest cluster center. The use of the DPP algorithm was proven to outperform other clustering approaches such as k-means and to be more appropriate in situations, like grasp analysis, where certain clusters are more recurrent than other ones. A hierarchical structure of the grasp types was learned using the same DPP-based clustering approach [113]. A hierarchical clustering approach was also used in [124] to find the relationships between different hand configurations based on a similarity measure between pairs of grasp types. Similarly, in [90], [120], the authors used a correlation index to measure the visual similarity between grasp types: grasp types with high correlation were clustered at the lower nodes, whereas low-correlated types were clustered higher in the hierarchy. The above approaches [90], [113], [120], [124] were used to build tree-like structures of the grasp types. These structures can be exploited to define new taxonomies depending on the trade-off between detail and robustness of grasp classification, as well as to discover new grasp types not included in previous categorizations [116].

4.1.3 Static hand gesture recognition

The recognition of static hand gestures is usually performed in a supervised manner, similarly to the approaches presented in Section 4.1.1 for hand grasp recognition. A common strategy is to exploit features extracted from previously segmented hand regions, classifying them into multiple gestures often using SVM classifiers [36], [72].

Serra et al. [36] classified the binary segmentation masks into multiple hand configurations by using an ensemble of exemplar-SVMs [133]. This approach was proven to be robust in case of unbalanced classes, like hand gesture recognition applications where most of the frames contain negative examples. Contour features were used in [72] to recognize 14 gestures. The authors described the silhouette of the hand shape using time curvature analysis and fed an SVM classifier with the extracted features. The use of CNNs has also been investigated for the recognition of static hand gestures [123], [127]. Ji et al. [127] used a hybrid CNN-SVM approach, where the CNN was implemented as feature extractor and the SVM as gesture recognizer. In [107], the authors proposed a CNN architecture to directly classify the binary hand masks into multiple gestures.

Depth information was used in [125], [126], [10]. In [125] the authors used depth context descriptors and random forest classification, whereas Jang et al. [126] implemented static-dynamic voxel features to capture the amount of point clouds within a voxel, in order to describe the static posture of the hands and fingers. Moreover, depth-based gesture recognition was demonstrated to be more discriminative than color-based recognition [10]. However, in addition to the drawbacks of wearable depth sensors already discussed in the previous sections, the performance were significantly lower in outdoor environments due to the deterioration of the depth map [10].

4.1.4 Dynamic hand gesture recognition

One of the most common choices for dynamic hand gesture recognition is to use optical flow descriptors from the segmented hand regions, in order to recognize the motion patterns of the gestures to be classified [128], [129], [96], [97].

Baraldi et al. [128], [129] developed an egocentric hand gesture classification system able to recognize the user's interactions with artworks in a museum. After removing camera motion, they computed and tracked the feature points at different spatial scales within the hand ROI and extracted multiple descriptors from the obtained spatio-temporal volume (e.g., HOG, HOF, and MBH). Linear SVM was used for recognizing multiple gestures from the above descriptors, using Bag of Words (BoW) and power normalization to avoid sparsity of the features. In [96], [97] the flow vectors were calculated over the entire duration of a gesture and, based on the resultant direction of the flow vectors, different swipe movements (e.g., left, right, up, and down) were classified using fixed thresholds on the movement orientation.

Other approaches recognized dynamic gestures as generalization of the static gesture recognition problem [126], [10]. In [126], the authors proposed a hierarchical approach for estimating hand gestures using a static-dynamic forest to produce hierarchical predictions on the hand gesture type. Static gesture recognition was performed at the top level of the hierarchy, in order to select a virtual object corresponding to the detected hand configuration (e.g., holding a stylus pen). Afterwards, the recognition of dynamic gestures, conditioned to the previously detected static gesture, was performed (e.g., pressing or releasing the button on the pen). Zhang et al. [10] compared engineered features and deep learning approaches (e.g., 2DCNN, 3DCNN, and recurrent models), demonstrating that 3DCNN are more suitable for dynamic hand gesture recognition and the combination of color and depth information can produce better results than the two image modalities alone.

4.1.5 Remarks on hand grasp analysis and gesture recognition

Many similarities can be found between grasp recognition and hand gesture recognition. As mentioned above, the main difference is the context in which the two problems are addressed. Grasp recognition is performed during hand-object manipulations, whereas hand gesture recognition is performed without the manipulation of physical objects. This difference links these two sub-areas to some of the higher levels and FPV applications. In fact, hand gesture recognition approaches have mainly been used for AR/VR applications [96], [125], [126], [97], whereas grasp analysis can be exploited for action/interaction recognition and activity recognition [121], [134]. In particular, the contextual relationship between grasp types and object attributes, such as rigidity and shape, has motivated authors [120], [121] to exploit object cues for improving the grasp recognition performance.

Hand grasp analysis and gesture recognition are the only interpretation sub-areas where the analysis of the hands is still the main target of the approaches. In fact, higher in the semantic content dimension, sub-areas like action recognition, interaction detection, and activity recognition may use the hand information in combination with other cues (e.g., object recognition) to perform higher level inference. It should be noted though, that not all the higher-level interpretation approaches utilized hand-based processing in FPV. Thus, in the following sections, we will discuss only those methods that explicitly used the hand information for predicting actions and activities, omitting other papers and referring the authors to other surveys or research articles.

4.2 Action/interaction and activity recognition

According to Tekin et al. [18], an action is a verb (e.g., "cut"), whereas an interaction is a verb-noun pair (e.g., "cut the bread"). Both definitions refer to short-term events that usually last a few seconds [37]. By contrast, activities are longer temporal events (i.e., minutes or hours) with higher semantic content, typically composed of temporally-consistent actions and interactions [8] (see Figure 3).

In this section, we summarize FPV approaches that relied on hand information to recognize actions, interactions, and activities from sequences of frames. Regarding the actions and interactions, two main types of approaches can be found in literature: those that used hands as the only cue for the prediction (see Section 4.2.1) and approaches that used a combination of object and hand cues (see Section 4.2.2). Although the second type of approaches might seem more suitable for interaction recognition (i.e., verb + noun prediction), some authors used them for predicting action verbs, exploiting the object information to prune the space of possible actions (i.e., removing unlikely verbs for a given object) [3]. Likewise, other authors tried to use only hand cues to recognize interactions [50], in order to produce robust predictions without relying on object features or object recognition algorithms. Either way, the boundary between action and interaction recognition is not well defined and often depends on the nature of the dataset on which a particular approach has been tested.

4.2.1 Action/interaction recognition using hand cues

These approaches inferred the camera wearer's actions exploiting the information provided by hand localization methods. The hypothesis is that actions and interactions can be recognized using only hand cues, for instance features related to the posture and motion of the hands. Existing studies can be divided into feature-based approaches [61], [135], [136] and deep learning-based approaches [37], [38], [50].

Feature-based approaches combined motion and shape features of the hands to represent two complementary aspects of the action: movements of hand's parts and grasp types. This representation allowed discriminating actions with similar hand motion, but different hand posture. Typical choices of motion features were dense trajectories [131], whereas the hand shape was usually represented with HOG [135] or shape descriptors on the segmented hand mask [136]. All these features were then combined and used to recognize actions/interactions via SVM classifiers. Ishihara et al. [135] used dense local motion features to track keypoints from which HOG, MBH, and HOF were extracted [131]. Global hand shape was represented using HOG features within the segmented hand region. The authors used Fisher vectors and principal component analysis to encode features extracted from time windows of fixed duration, followed by multiclass linear SVM for the recognition. Dense trajectory features were also used by Kumar et al. [61]. The authors proposed a feature sampling scheme that preserved dense trajectories closer to the hand centroid while removing trajectories from the background, which are likely caused by head motion. BoW representation was used and the recognition was performed using SVM with χ^2 kernel. Cai et al. [136] combined hand shape, hand position, and hand motion features for recognizing user's desktop actions (e.g., browse, note, read, type, and write). Histograms of the hand shape computed on the hand mask were used as shape features. Hand position was represented by the point within the hand region where a manipulation is most likely to happen (e.g., left tip of the right hand region). Motion descriptors relied on the computation of the large displacement optical flow (LDOF) [137] between two consecutive frames. Spatio-temporal distribution of hand motion (i.e., discrete Fourier transform coefficients on the average LDOF extracted from hand sub-regions over consecutive frames) was demonstrated to outperform temporal and spatial distributions alone, suggesting that spatial and temporal information should be considered together when recognizing hand's actions.

The combination of temporal and spatial information was also exploited in deep-learning approaches. This strategy was usually implemented by means of multi-stream architectures. Singh et al. [37] proposed a CNN-based approach to recognize camera wearer's actions using the following inputs: pixel-level hand segmentation mask; head motion – as frame-to-frame homography using RANSAC on optical flow correspondences excluding the hand regions; and saliency map – as the flow map obtained after applying the homography. This information was passed to a 2-stream architecture composed of a 2DCNN and a 3DCNN. The deep-learned features from both streams were combined and actions were predicted using SVM. Urabe et al. [38] used the region around the hands to recognize cooking actions. Appearance and motion maps were obtained using the segmented hand mask passed to 2DCNN and 3DCNN, respectively. Afterwards, class-score fusion was performed by multiplying the output of both streams. The authors demonstrated that a multi-stream approach yielded better results than the two streams alone. Tang et al. [50] used the hand information as auxiliary stream within an end-to-end multi-stream deep neural network (MDNN) that used RGB, optical flow and depth maps as input. The hand stream was composed of a CNN with the hand mask as input. Its output was combined to the MDNN via weighted fusion, in order to predict the action label. The addition of the hand stream improved the recognition performance.

4.2.2 Action/interaction recognition combining hand and object cues

Many authors demonstrated that the combination of object and hand cues can improve the recognition performance [3], [138], [69], [121], [87]. This is quite intuitive, since during an interaction the grasp type and hand movements strictly depend on the characteristics of the object that is being manipulated (e.g., dimension, shapes, functionality) [121]. Thus, grasp type or hand pose/shape along with object cues can be used to recognize the actions and interactions [18], [3], [33], [12], [121], [134].

In [121], the authors predicted the attributes of the manipulated object (i.e., object shape and rigidity) and the type of grasp to recognize hand's actions. They proposed a hierarchical 2-

stage approach where the lower layer – visual recognition – classified the grasp type and the object attributes and pass this information to the upper layer – action modeling – responsible for the action classification via linear SVM. Coskun et al. [134] implemented a recurrent neural network (RNN) to exploit the temporal dependencies of consecutive frames using a set of deep-learned features related to grasp, optical flow, object-object, and hand-object interactions, as well as the trajectories of the hands over the past few frames. Other authors [18], [3], [33], [12], used hand cues with lower semantic content than hand grasp, such as shape and pose. Fathi et al. [3] extracted a set of object and hand descriptors (e.g., object and hand labels, optical flow, location, shape, and size) at super-pixel level and performed a 2-stage interaction recognition. First, they recognized actions using Adaboost; second, they refined the object recognition in a probabilistic manner by exploiting the predicted verb label and object classification scores. Likitlersuang et al. [33] detected the presence of interactions between the camera wearer's hands and manipulated objects. This was accomplished by combining the hand shape, represented with HOG descriptors, with color and motion descriptors (e.g., color histogram and optical flow) for the hand, the background, and the object (i.e., regions around the hands). Random forest was used for classification. The articulated hand pose was used in [18], [12]. Garcia-Hernando et al. [12] passed the hand and object key-points to an LSTM that predicted the interactions over the video frames. This approach was extended in [18], where hand-object interactions were first modeled using a multi-layer perceptron and then used as input to the LSTM.

Other approaches, instead of explicitly using the hand information for predicting actions and interactions, exploited the results of hand localization algorithms to guide the feature extraction within a neighborhood of the manipulation region [138], [139]. This strategy was motivated by the fact that the most important cues (i.e., motion, object, etc.) during an action are likely to be found in proximity of the hands and manipulated object. Li et al. [138] used a combination of local descriptors for motion and object cues in conjunction with a set of egocentric cues. The former, were extracted from the dense trajectories to represent the motion of the action (i.e., shape of the trajectories, MBH, HoF) and the object appearance (e.g., HOG, LAB color histogram, and LBP along the trajectories). The latter were used to approximate the gaze information, by combining camera motion removal and hand segmentation, in order to focus the attention on the area where the manipulation is happening. Ma et al. [139] used a multi-stream deep learning approach composed of an appearance stream to recognize the object and a motion stream to predict the action verb. The object recognition network predicted the object label by using as input the hand mask and object ROI, whereas the action recognition network used the optical flow map to infer the verb. A fusion layer combined verb and object labels and predicted the interactions. Zhou et al. [42] used the hand segmentation mask, object features extracted from middle layers of AlexNet [130], and optical flow to localize and recognize the active object using VGG-16 [67]. Afterwards, object features were represented in a temporal pyramid manner and combined with motion characteristics extracted from improved dense trajectories, in order to recognize interactions using non-linear SVM. Although the above approaches might differ for the type of features and algorithm used to predict actions and interactions, most of them demonstrated that the combination of object and hand cues can provide better recognition performance than single modality recognition [138], [69].

4.2.3 Activity recognition

As we climb the semantic content dimension in the proposed framework, the strong dependency on hand cues fades away. Other information comes into play and can be used in conjunction with the hands to predict the activities. This diversification becomes clear when we look at the review published by Nguyen et al. [8], which categorized egocentric activity recognition as: 1) combination of actions; 2) combination of active objects; 3) combination of active objects and locations; 4) combination of active objects and hand movements; and 5) combination of other information (e.g., gaze, motion, etc.). The description of all these approaches goes beyond the scope of this work, since we are interested in characterizing how hands can be used in activity recognition methods. For a more comprehensive description of activity recognition in FPV, the reader is referred to [8]. The boundary between the recognition of short and long temporal events (i.e., actions/interactions and activities, respectively) is not always well defined and, similar to action/interaction recognition, it may depend on the dataset used for training and testing a particular approach. In fact, some of the methods described in the previous subsections were also tested within an activity recognition framework [69], [139]. Generally, we can identify two types of approaches: activity recognition based on actions and interactions [3], [81] and approaches that used hand localization results to directly prectict the activities [19], [49], [73].

Approaches that relied on actions and interactions learned a classifier for recognizing the activities using the detected actions or hand-object interactions as features for the classification. This can be performed by using the histogram of action frequency in the video sequence and its classification using adaboost [3]. Nguyen et al. [81] used Bag of Visual Words representation to model the interactions between hands and objects, since these cues play a key role in the recognition of activities. Dynamic time warping was then used to compare a new sequence of features with the key training features.

Other authors [19], [49], [73] investigated how good the hand segmentation map is in predicting a small set of social activities, such as 4 interactions between two individuals. The authors used a CNN-based approach using the binary hand segmentation maps as input. The prediction was performed on a frame-by-frame basis and using temporal integration implemented through a voting strategy among consecutive frames, with the latter approach providing better results (up to 73% of recognition accuracy) [19]. This result confirms what was already shown for actions and interactions, namely the temporal information becomes essential when performing higher-level inference, especially when modeling relatively long term events like activities. However, this approach was tested only in case of a small sample of social activities. To the best of our knowledge, no experiments using hand cues only were conducted for predicting other types of activities, such as ADLs.

4.2.4 Remarks on action/interaction and activity recognition

Many authors demonstrated that action/interaction recognition performance can be improved by combining different cues, such as hand, object, and motion information. This was proven regardless of the actual method. In fact, both feature-based and deep-learning based methods implemented this strategy by combining

multiple features or using multi-stream deep-learning approaches. Recently, multi-task learning approaches have also been proposed for solving the action recognition problem [140], demonstrating that predicting hand coordinates as an auxiliary task leads to an improvement in verb recognition performance with respect to the single-task approach. In the near future, it will be interesting to compare multi-task and multi-stream architectures to understand whether the joint prediction of action labels and hand positions can actually provide state-of-the-art performance in one or both tasks.

Another important aspect on which one should focus when developing novel approaches for action/interaction recognition is the temporal information. This was exploited by using 3DCNN and RNNs or, in case of feature-based approaches, by encoding it in the motion information. The same conclusion can be drawn for activity recognition where, considering the longer duration of the events, the temporal information becomes even more important [19].

Sometimes the literature is not consistent on the choice of the taxonomy to describe these sub-areas. Some of the approaches summarized above, even though not explicitly referred as action/interaction recognition, actually recognized short actions or interactions. We preferred to be consistent with the definition proposed by Tekin et al. [18], as we believe that a consistent taxonomy may help authors comparing different approaches and unify their efforts towards solving a specific problem. Moreover, the term "action" has often been used interchangeably with "activity", which indicates a longer event with higher semantic content. The actions and interactions can rather be seen as the building blocks of the activities. This allowed some authors to exploit this mutual dependency, in order to infer activities in a hierarchical manner, using the methods described above [3], [81].

The number of egocentric activity recognition approaches based on hand information is lower than the number of action and interaction recognition approaches. This difference is due to the fact that higher in the semantic content, authors have a wider choice of cues and features for recognizing a temporal event. In particular, over the past few years, more and more end-to-end approaches for activity recognition have been proposed, similarly to video recognition [141].

5 Application

The hand-based approaches summarized so far can be implemented to design real-world FPV applications. Most of these applications relied on HCI and HRI and included AR and VR systems [142], [108], robot control and learning [143], [107], as well as healthcare applications [33], [80].

5.1 AR and VR applications

Given the recent success of VR headsets and the surge of AR applications, many hand-based methods in FPV were used for AR and VR systems to design natural user interfaces. Most of these applications relied on hand localization – in particular, hand detection, segmentation, and pose-estimation – and gesture recognition algorithms, for example to manipulate virtual objects in an AR or VR scenario [142], [144], [126], [125]. Depth sensors were usually implemented to capture the scene, whereas head-worn displays allowed projecting the virtual object in the AR/VR scenario. The recognition of specific hand gestures allowed providing inputs and commands to the system, in order to produce a specific action (e.g., the selection of a virtual object by recognizing the clicking gesture [144]). The use of depth sensors has usually been preferred to RGB cameras since the localization of hands and objects can be more robust to illumination changes. Some authors even implemented multiple depth sensors [142]: one specific for short distances (i.e., up to 1 m) – to capture more accurate hand information – and a long-range depth camera to reproduce the correct proportions between the physical and virtual environment [142]. To improve the hand localization robustness, other systems combined multiple hand localization approaches, such as hand pose estimation in conjunction with fingertip detection [144]. This approach can be helpful when the objective is to localize the fingertips in situations with frequent self-occlusions. Other AR/VR applications relied on dynamic hand gestures (e.g. swipe movements) recorded with a smartphone camera and frugal VR devices (e.g., Google Cardboard), in order to enable interactions in the virtual environment [96], [97]. AR/VR applications were also implemented for tele-support and coexistence reality [145], [146], in order to allow multiple users to collaborate together remotely. Specific fields of applications were remote co-surgery [145] and expert's tele-assistance and support [146].

Within the AR context, the use of hand-based information was also exploited for recognizing hand-written characters [109], [147]. This application was performed in four steps: 1) hand localization through hand detection and tracking; 2) hand gesture recognition – to recognize a specific hand posture that triggers the writing module; 3) fingertip detection – to identify the point to track, whose trajectory defines the characters; and 4) character recognition, based on the trajectories of the detected fingertip [109], [147].

Hand localization and gesture recognition approaches were also used for cultural heritage applications to develop systems for immersive museum and augmented touristic experiences [36], [128], [128], [108]. Users can experience an entertaining way of accessing the museum knowledge, for example by taking pictures and providing feedback to the artworks with simple hand gestures [129]. Other authors [108] proposed a smart glasses-based system that allowed users to access touristic information while visiting a city by using pointing gestures of the hand.

5.2 Robot control and learning

In the HRI field, FPV hand-based approaches have mainly been used for two purposes: robot learning and robot control. Approaches for robot learning recognized movements and/or actions performed by the user's hands, in order to train a robot performing the same set of movements autonomously [143], [148]. Aksoy et al. [143], decomposed each manipulation into shorter chunks and encoded each manipulation into a semantic event chain (SEC), which encodes the spatial relationship between objects and hands in the scene. Each temporal transition in the SEC (e.g., change of state in the scene configuration) was considered as movement primitive for the robot imitation. In [148], the robot used the tracked hand locations of a human to learn the hand's future position and predict trajectories of the hands when a particular action has to be executed. By contrast, robot control approaches mainly relied on hand gesture recognition to give specific real-time commands to the robots [107], [127]. The hand gestures are seen as means of communication between the human and the robot and can encode specific commands such as the action to be performed by a robot arm [107] or the direction to be taken by a reconnaissance robot [127].

5.3 Remote healthcare monitoring

Egocentric vision has demonstrated the potential to have an important impact in healthcare. The possibility to automatically analyze the articulated hand pose and recognize actions and ADLs have made these methods appealing for the remote assessment of upper limb functions [32], [25], [149], [33], [85] and AAL systems [150], [8], [80]. The assessment of upper limb function is an important phase in the rehabilitation after stroke or cSCI that allows clinicians to plan the optimal treatment strategy for each patient. However, geographical distances between patients and hospitals create barriers towards obtaining optimal assessments and rehabilitation outcomes. Egocentric vision has inspired researchers to develop video-based approaches for automatically studying hand functions at home [32], [25], [33], [85]. Studies have been conducted in individuals with cSCI, tackling the problem of hand function assessment from two perspectives: localization [32], [98], [85] and interpretation [25], [33]. Fine-tuning object detection algorithms to localize and recognize hands in people with SCI allowed developing hand localization approaches robust to impaired hand poses and uncontrolled situations [85]. Moreover, strategies for improving the computational performance of hand detection algorithms have been adopted (e.g., combining hand detection and tracking), making this application suitable for the use at home. The automatic detection of hand-object manipulations allowed extracting novel measures reflective of the hand usage at home, such as number of interactions per hour, the duration of interactions, and the percentage of interaction over time [33]. These measures, once validated against clinical scores, will help clinicians to better understand how individuals with cSCI and stroke use their hands at home while performing ADLs.

Another healthcare application is the development of AAL systems. The increasing ageing population is posing serious social and financial challenges in many countries. These challenges have stimulated the interest in developing technological solutions to help and support older adults with and without cognitive impairments during their daily life [151]. Some of these applications used egocentric vision to provide help and support to older adults during ADLs at home [150], [8], [80]. Egocentric vision AAL builds upon the action and activity recognition approaches illustrated in Section 4.2. In particular, approaches have been proposed to automatically recognize how older adults perform ADLs at home, for example to detect early signs of dementia [150] or to support people in conducting the activities [80].

In these specific applications, the use of egocentric vision presents important advantages with respect to other solutions (e.g., sensor-based and third person vision):

- FPV can provide high quality videos on how people manipulate objects. This is important when the aim is the recognition of hand-object manipulations and ADLs, since hand occlusions tend to be minimized.
- Egocentric vision provides more details of hand-object interactions than sensor-based technology, by capturing information about both the hand and the object being manipulated. Other sensor-based solutions such as sensor gloves, although providing highly accurate hand information, may limit movements and sensation, which are already reduced in individuals with upper limb impairment [33], [149].

6 FPV Datasets with hand annotation

The importance that this field of research gained in recent years is clear when we look at the number of available datasets published since 2015 (Table 1). Although the type of information and ground truth annotations made available by the authors is heterogeneous, it is possible to identify some sub-areas that are more recurrent than others. The vast majority of datasets provided hand segmentation masks [2], [128], [129], [19], [25], [136], [49], [156], [50], [11], reflecting the high number of approaches proposed in this area and summarized in Section 3. However, the high number of datasets is counterbalanced by a relative low number of annotated frames, usually in the order of a few hundreds or thousands of images. To expedite the lenghty pixel-level annotation process and build larger datasets for hand segmentation, some authors proposed semi-automated techniques, for example based on Grabcut [136], [46]. Actions/activities [152], [69], [136], [12], [156], [50] and hand gestures [128], [129], [109], [97], [111], [154], [10] are other common information that were captured and annotated in many datasets. This large amount of data has been used by researchers for developing robust HCI applications that relied on hand gestures. Compared to the amount of hand segmentation masks, action/activities and hand gestures datasets are usually larger, since the annotation process is easier and faster than pixel-level segmentation.

The vast majority of datasets included color information recorded from head-mounted cameras. The head position is usually preferred over the chest or shoulders, since it is easier to focus on hand actions and manipulations whenever the camera wearer's is performing a specific activity. GoPro cameras were the most widely used devices for recording the videos, since they are specifically designed for egocentric POV and are readily available on the market. Few datasets, usually designed for hand pose estimation [12], [104], [154], hand gesture recognition [109], [10], and action/activity recognition [69], [12], [50], include depth or color and depth information. In most cases, videos were collected using Creative® Senz3D™ or Intel® RealSense™ SR300 depth sensors, as these devices were small and lightweight. Moreover, these cameras were preferred over other depth sensors (e.g., Microsoft® Kinect™) because they were originally developed for natural user interface that made them more suitable for studying hand movements in the short range (i.e., up to 1 m of distance from the camera).

Although FPV is gaining a lot of interest for developing healthcare applications for remote monitoring of patients living in the community, only one dataset (i.e., the ANS-SCI dataset [33]) included videos from people with clinical conditions such as cSCI. This lack of available data is mainly due to ethical constraints that make it harder to share videos and images collected from people with diseases or clinical conditions. In the next few years researchers should try – within the ethical and privacy constraints – to build and share datasets for healthcare applications including videos collected from patients. This will benefit the robustness of the hand-based approaches in FPV against the inter-group and within group variability that can be encountered in many clinical conditions.

7 Conclusion

In this paper we showed how hand-related information can be retrieved and used in egocentric vision. We summarized the existing literature into three macro-areas, identifying the most

Dataset	Year	Mode	Device	Location	Frames	Videos	Duration	Subjects	Resolution (pixels)	Annotation
GTEA [65]	2011	C	GoPro	H	∼31k	28	34 min	4	1280×720	act msk
ADL [152]	2012	C	GoPro	H	>1M	20	∼10 h	20	1280×960	act obj
EDSH [2]	2013	C	-	H	∼20k	3	∼10 min	-	1280×720	msk
Interactive Museum [128]	2014	C	-	H	-	700	-	5	800×450	gst msk
EgoHands [19]	2015	C	Google Glass	H	∼130k	48	72 min	8	1280×720	msk
Maramotti [129]	2015	C	-	H	-	700	-	5	800×450	gst msk
UNIGE Hands [153]	2015	C	GoPro Hero3+	H	∼150k	-	98 min	-	1280×720	det
GUN-71 [70]	2015	CD	Creative Senz3D	C	∼12k (annotated)	-	-	8	-	grs
RGBD Egocentric Action [69]	2015	CD	Creative Senz3D	H	-	-	-	20	C:640×480 D:320×240	act
Fingerwriting in mid-air [109]	2016	CD	Creative Senz3D	H	∼8k	-	-	-	-	ftp gst
Ego-Finger [147]	2016	C	-	H	∼93k	24	-	24	640×480	det ftp
ANS able-bodied [25]	2016	C	Looxcie 2	-	-	-	44 min	4	640×480	int
UT Grasp [90]	2017	C	GoPro Hero2	H	-	50	∼4 h	5	960×540	grs
GestureAR [97]	2017	C	Nexus 6 and Moto G3	H	-	100	-	8	1280×720	gst
EgoGesture [111]	2017	C	-	-	∼59k	-	-	-	-	det ftp gst
Egocentric hand-action [154]	2017	D	Softkinetic DS325	H	∼154k	300	-	26	320×240	gst
BigHand2.2M [104]	2017	D	Intel RealSense SR300	-	∼290k	-	-	-	640×480	pos
Desktop Action [136]	2018	C	GoPro Hero2	H	∼324k	60	3 h	6	1920×1080	act msk
Epic Kitchens [155]	2018	C	GoPro	H	11.5M	-	55 h	32	1920×1080	act
FPHA [12]	2018	CD	Intel RealSense SR300	S	∼105k (annotated)	1,175	-	6	C:1920×1080 D:640×480	act pos
EYTH [49]	2018	C	-	-	1,290 (annotated)	3	-	-	-	msk
EGTEA+ [156]	2018	C	SMI wearable eye-tracker	H	>3M	86	∼28 h	32	1280×960	act gaz msk
THU-READ [50]	2018	CD	Primesense Carmine	H	∼343k	1,920	-	8	640×480	act msk
EgoGesture [10]	2018	CD	Intel RealSense SR300	H	∼3M	24,161	-	50	640×480	gst
EgoDaily [86]	2019	C	GoPro Hero5	-	∼50k	50	-	10	1920×1080	det hid
ANS SCI [33]	2019	C	GoPro Hero4	H	-	-	-	17	1920×1080	det int
KBH [11]	2019	C	HTC Vive	H	∼12.5k (annotated)	161	-	50	230×306	msk

TABLE 1
List of available datasets with hand-based annotations in FPV. Image modality (Mode): Color (C); Depth (D); Color+Depth (CD). Camera location: Head (H); Chest (C); Shoulder (S). Annotation: actions/activities (**act**); hand presence and location (**det**); fingertip positions (**ftp**); gaze (**gaz**); grasp types (**grs**); hand gestures (**gst**); hand disambiguation (**hid**); hand-object interactions (**int**); hand segmentation masks (**msk**); object classes (**obj**); hand pose (**pos**).

prominent approaches for hand localization (e.g., hand detection, segmentation, pose estimation, etc.), interpretation (grasp analysis, gesture recognition, action and activity recognition), as well as the FPV applications for building real-world solutions. We believe that a comprehensive taxonomy and an updated framework of hand-based methods in FPV may serve as guidelines for the novel approaches proposed in this field by helping to identify gaps and standardize terminology.

One of the main factors that promoted the development of FPV approaches for studying the hands is the availability of wearable action cameras and AR/VR systems. However, we also showed how the use of depth sensors, although not specifically developed for wearable applications, has been exploited by many authors, in order to improve the robustness of hand localization. We believe that the possibility to develop miniaturized wearable depth sensors may further boost the research in this area and the development of novel solutions, since a combination of color and depth information can improve the performance of several hand-based methods in FPV. In particular, these advantages can be exploited in settings that involve short–range observations (i.e., less than 1 m) and indoor environments, which are often encountered when analyzing the hands in FPV.

From this survey it is clear how the hand localization step plays a vital role in any processing pipeline, as a good localization is necessary condition for hand-based higher inference, such as gesture or action recognition. This importance has motivated the extensive research conducted in the past 10 years, especially in sub-areas like hand detection and segmentation. The importance of hand localization methods may also be seen in those approaches where the hands play an auxiliary role, such as activity recognition. In fact, the position of the hands can be used to build attention-based classifiers, where more weight is given to the manipulation area.

Like other computer vision fields, the advent of deep learning has had a great impact on this area, by boosting the performance of several localization and interpretation approaches, as well as optimizing the number of steps required to pursue a certain objective (see the hand identification example – Section 3.3). Hand detection is the localization sub-area that has seen the largest improvements, especially thanks to the availability of object detection networks retrained on large datasets. Other sub-areas, such as hand segmentation and pose estimation, perhaps will see larger improvements in the next few years, especially if the amount of available data annotations grows. Recurrent models, 3DCNN, and the availability of large datasets (e.g., Epic Kitchens, EGTEA+, etc.) have helped pushing the state of the art of action and activity recognition, considering that the combination of temporal and appearance information was demonstrated to be crucial for these tasks. In the near future, efforts should be made in improving methods for the recognition of larger classes of unscripted ADLs, which would benefit the development of applications such as AAL.

As this field of research is still growing, we will see novel applications and improvement of the existing ones. The impact of hand-based methods in egocentric vision is clear from the development of applications that relied on HCI and HRI. The importance of the hands as our primary means of interaction with the world around us is currently exploited by VR and AR systems, and the position of the wearable camera offers tremendous advantages for assessing upper limb function remotely and supporting older adults in ADLs. This will translate in the availability of rich information captured in natural environments, with the possibility to improve assessment and diagnosis, provide new interaction modalities, and enable personalized feedback on tasks and behaviours.

ACKNOWLEDGMENTS

This work was supported in part by the Craig H. Neilsen Foundation (542675). The authors would also like to thank Gustavo Balbinot, Guijin Li, and Mehdy Dousty for the helpful discussions and feedback.

REFERENCES

[1] Govert J Snoek, Maarten J IJzerman, Hermie J Hermens, Douglas Maxwell, and Fin Biering-Sorensen. Survey of the needs of patients with spinal cord injury: impact and priority for improvement in hand function in tetraplegics. *Spinal Cord*, 42(9):526, 2004.

[2] Cheng Li and Kris M Kitani. Pixel-level hand detection in ego-centric videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3570–3577, 2013.

[3] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 407–414. IEEE, 2011.

[4] Siddharth S Rautaray and Anupam Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54, 2015.

[5] Xenophon Zabulis, Haris Baltzakis, and Antonis A Argyros. Vision-based hand gesture recognition for human-computer interaction. *The Universal Access Handbook*, 34:30, 2009.

[6] Steve Mann. 'wearcam'(the wearable camera): personal imaging systems for long-term use in wearable tetherless computer-mediated reality and personal photo/videographic memory prosthesis. In *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No. 98EX215)*, pages 124–131. IEEE, 1998.

[7] Alejandro Betancourt, Pietro Morerio, Carlo S Regazzoni, and Matthias Rauterberg. The evolution of first person vision methods: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(5):744–760, 2015.

[8] Thi-Hoa-Cuc Nguyen, Jean-Christophe Nebel, Francisco Florez-Revuelta, et al. Recognition of activities of daily living with egocentric vision: A review. *Sensors*, 16(1):72, 2016.

[9] Alejandro Betancourt, Pietro Morerio, Lucio Marcenaro, Emilia Barakova, Matthias Rauterberg, and Carlo Regazzoni. Towards a unified framework for hand-based methods in first person vision. In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2015.

[10] Yifan Zhang, Congqi Cao, Jian Cheng, and Hanqing Lu. Egogesture: a new dataset and benchmark for egocentric hand gesture recognition. *IEEE Transactions on Multimedia*, 20(5):1038–1050, 2018.

[11] Wei Wang, Kaicheng Yu, Joachim Hugonot, Pascal Fua, and Mathieu Salzmann. Recurrent u-net for resource-constrained segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2142–2151, 2019.

[12] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 409–419, 2018.

[13] Hong Cheng, Lu Yang, and Zicheng Liu. Survey on 3d hand gesture recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(9):1659–1673, 2015.

[14] Rui Li, Zhenyu Liu, and Jianrong Tan. A survey on 3d hand pose estimation: Cameras, methods, and datasets. *Pattern Recognition*, 93:251–272, 2019.

[15] Ana Garcia del Molino, Cheston Tan, Joo-Hwee Lim, and Ah-Hwee Tan. Summarization of egocentric videos: A comprehensive survey. *IEEE Transactions on Human-Machine Systems*, 47(1):65–76, 2016.

[16] Marc Bolanos, Mariella Dimiccoli, and Petia Radeva. Toward storytelling from visual lifelogging: An overview. *IEEE Transactions on Human-Machine Systems*, 47(1):77–90, 2016.

[17] Alexandros André Chaaraoui, Pau Climent-Pérez, and Francisco Flórez-Revuelta. A review on vision techniques applied to human behaviour analysis for ambient-assisted living. *Expert Systems with Applications*, 39(12):10873–10888, 2012.

[18] Bugra Tekin, Federica Bogo, and Marc Pollefeys. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4511–4520, 2019.

[19] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1949–1957, 2015.

[20] Grégory Rogez, Maryam Khademi, JS Supančič III, Jose Maria Martinez Montiel, and Deva Ramanan. 3d hand pose detection in egocentric rgb-d images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 356–371. Springer, 2014.

[21] Xiaolong Zhu, Wei Liu, Xuhui Jia, and Kwan-Yee K Wong. A two-stage detector for hand detection in ego-centric videos. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8. IEEE, 2016.

[22] Alejandro Cartas, Mariella Dimiccoli, and Petia Radeva. Detecting hands in egocentric videos: Towards action recognition. In *International Conference on Computer Aided Systems Theory*, pages 330–338. Springer, 2017.

[23] Michael J Jones and James M Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.

[24] Hui Liang, Junsong Yuan, and Daniel Thalman. Egocentric hand pose estimation and distance recovery in a single rgb image. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2015.

[25] Jirapat Likitlersuang and Jose Zariffa. Interaction detection in egocentric video: Toward a novel outcome measure for upper extremity function. *IEEE Journal of Biomedical and Health Informatics*, 22(2):561–569, 2016.

[26] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[27] Thomas P Weldon, William E Higgins, and Dennis F Dunn. Efficient gabor filter design for texture segmentation. *Pattern Recognition*, 29(12):2005–2015, 1996.

[28] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. 2005.

[29] David G Lowe et al. Object recognition from local scale-invariant features. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 99, pages 1150–1157, 1999.

[30] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 778–792. Springer, 2010.

[31] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 11, page 2. Citeseer, 2011.

[32] José Zariffa and Milos R Popovic. Hand contour detection in wearable camera video using an adaptive histogram region of interest. *Journal of Neuroengineering and Rehabilitation*, 10(1):114, 2013.

[33] Jirapat Likitlersuang, Elizabeth R Sumitro, Tianshi Cao, Ryan J Visée, Sukhvinder Kalsi-Ryan, and José Zariffa. Egocentric video: a new tool for capturing hand use of individuals with spinal cord injury at home. *Journal of Neuroengineering and Rehabilitation*, 16(1):83, 2019.

[34] Piotr Dollár and C Lawrence Zitnick. Structured forests for fast edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1841–1848, 2013.

[35] Alejandro Betancourt, Miriam M López, Carlo S Regazzoni, and Matthias Rauterberg. A sequential classifier for hand detection in the framework of egocentric vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 586–591, 2014.

[36] Giuseppe Serra, Marco Camurri, Lorenzo Baraldi, Michela Benedetti, and Rita Cucchiara. Hand segmentation for gesture recognition in ego-vision. In *Proceedings of the 3rd ACM international workshop on Interactive multimedia on mobile & portable devices*, pages 31–36. ACM, 2013.

[37] Suriya Singh, Chetan Arora, and CV Jawahar. First person action recognition using deep learned descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2620–2628, 2016.

[38] Shuichi Urabe, Katsufumi Inoue, and Michifumi Yoshioka. Cooking activities recognition in egocentric videos using combining 2dcnn and 3dcnn. In *Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management*, pages 1–8. ACM, 2018.

[39] Xiaolong Zhu, Xuhui Jia, and Kwan-Yee K Wong. Pixel-level hand detection with shape-aware structured forests. In *Asian Conference on Computer Vision*, pages 64–78. Springer, 2014.

[40] Xiaolong Zhu, Xuhui Jia, and Kwan-Yee K Wong. Structured forests for pixel-level hand detection and hand part labelling. *Computer Vision and Image Understanding*, 141:95–107, 2015.

[41] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.

[42] Yang Zhou, Bingbing Ni, Richang Hong, Xiaokang Yang, and Qi Tian. Cascaded interactional targeting network for egocentric video analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1904–1913, 2016.

[43] Yinlin Li, Lihao Jia, Zidong Wang, Yang Qian, and Hong Qiao. Unsupervised and semi-supervised hand segmentation in egocentric images with noisy label learning. *Neurocomputing*, 334:11–24, 2019.

[44] Zhengqin Li and Jiansheng Chen. Superpixel segmentation using linear spectral clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1356–1363, 2015.

[45] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[46] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[47] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[48] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1925–1934, 2017.

[49] Aisha Urooj and Ali Borji. Analysis of hand segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4710–4719, 2018.

[50] Yansong Tang, Zian Wang, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Multi-stream deep neural networks for rgb-d egocentric action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[51] Wei Wang, Kaicheng Yu, Joachim Hugonot, Pascal Fua, and Mathieu Salzmann. Beyond one glance: Gated recurrent architecture for hand segmentation. *arXiv preprint arXiv:1811.10914*, 2018.

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[53] Minglei Li, Lei Sun, and Qiang Huo. Flow-guided feature propagation with occlusion aware detail enhancement for hand segmentation in egocentric videos. *Computer Vision and Image Understanding*, 187:102785, 2019.

[54] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2349–2358, 2017.

[55] Alejandro Betancourt, Lucio Marcenaro, Emilia Barakova, Matthias Rauterberg, and Carlo Regazzoni. Gpu accelerated left/right hand-segmentation in first person vision. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 504–517. Springer, 2016.

[56] Cheng Li and Kris M Kitani. Model recommendation with virtual probes for egocentric hand detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2624–2631, 2013.

[57] Shao Huang, Weiqiang Wang, and Ke Lu. Egocentric hand detection via region growth. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 639–644. IEEE, 2016.

[58] Shao Huang, Weiqiang Wang, Shengfeng He, and Rynson WH Lau. Egocentric hand detection via dynamic region growing. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(1):10, 2018.

[59] Ying Zhao, Zhiwei Luo, and Changqin Quan. Unsupervised online learning for fine-grained hand segmentation in egocentric video. In *2017 14th Conference on Computer and Robot Vision (CRV)*, pages 248–255. IEEE, 2017.

[60] Ying Zhao, Zhiwei Luo, and Changqin Quan. Coarse-to-fine online learning for hand segmentation in egocentric video. *EURASIP Journal on Image and Video Processing*, 2018(1):20, 2018.

[61] Jayant Kumar, Qun Li, Survi Kyal, Edgar A Bernal, and Raja Bala. On-the-fly hand detection training with application in egocentric action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 18–27, 2015.

[62] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.

[63] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[64] Javier Sánchez Pérez, Enric Meinhardt-Llopis, and Gabriele Facciolo. Tv-l1 optical flow estimation. *Image Processing On Line*, 2013:137–150, 2013.

[65] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3281–3288. IEEE, 2011.

[66] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015.

[67] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[68] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

[69] Shaohua Wan and JK Aggarwal. Mining discriminative states of hands and objects to recognize egocentric actions with a wearable rgbd camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 36–43, 2015.

[70] Grégory Rogez, James S Supancic, and Deva Ramanan. Understanding everyday hands in action from rgb-d images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3889–3897, 2015.

[71] Wataru Yamazaki, Ming Ding, Jun Takamatsu, and Tsukasa Ogasawara. Hand pose estimation and motion recognition using egocentric rgb-d video. In *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 147–152. IEEE, 2017.

[72] Yiyi Ren, Xiang Xie, Guolin Li, and Zhihua Wang. Hand gesture recognition with multiscale weighted histogram of contour direction normalization for wearable applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(2):364–377, 2016.

[73] Sven Bambach, David J Crandall, and Chen Yu. Viewpoint integration for hand-based recognition of social interactions from a first-person view. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 351–354. ACM, 2015.

[74] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[75] Alejandro Betancourt, Pietro Morerio, Lucio Marcenaro, Matthias Rauterberg, and Carlo Regazzoni. Filtering svm frame-by-frame binary classification in a detection framework. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 2552–2556. IEEE, 2015.

[76] Ran Margolin, Ayellet Tal, and Lihi Zelnik-Manor. What makes a patch distinct? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1139–1146, 2013.

[77] Jingtao Wang and Chunxuan Yu. Finger-fist detection in first-person view based on monocular vision using haar-like features. In *Proceedings of the 33rd Chinese Control Conference*, pages 4920–4923. IEEE, 2014.

[78] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[79] Sergio R Cruz and Antoni B Chan. Hand detection using deformable part models on an egocentric perspective. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7. IEEE, 2018.

[80] Thi Hoa Cuc Nguyen, Jean-Christophe Nebel, Gordon Hunter, and Francisco Florez-Revuelta. Automated detection of hands and objects in egocentric videos, for ambient assisted living applications. In *2018 14th International Conference on Intelligent Environments (IE)*, pages 91–94. IEEE, 2018.

[81] Jean-Christophe Nebel, Francisco Florez-Revuelta, et al. Recognition of activities of daily living from egocentric videos using hands detected by a deep convolutional network. In *International Conference Image Analysis and Recognition*, pages 390–398. Springer, 2018.

[82] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1284–1293, 2017.

[83] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[84] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[85] Ryan J Visée, Jirapat Likitlersuang, and José Zariffa. An effective and efficient method for detecting hands in egocentric videos for rehabilitation applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2020.

[86] Sergio Cruz and Antoni Chan. Is that my hand? an egocentric dataset for hand disambiguation. *Image and Vision Computing*, 89:131–143, 2019.

[87] Georgios Kapidis, Ronald Poppe, Elsbeth van Dam, Lucas PJJ Noldus, and Remco C Veltkamp. Egocentric hand track and object-based human action recognition. *arXiv preprint arXiv:1905.00742*, 2019.

[88] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 21–37. Springer, 2016.

[89] Stefan Lee, Sven Bambach, David J Crandall, John M Franchak, and Chen Yu. This hand is my hand: A probabilistic approach to hand disambiguation in egocentric video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 543–550, 2014.

[90] Minjie Cai, Kris M Kitani, and Yoichi Sato. An ego-vision system for hand grasp analysis. *IEEE Transactions on Human-Machine Systems*, 47(4):524–535, 2017.

[91] Xiaorui Liu, Yichao Huang, Xin Zhang, and Lianwen Jin. Fingertip in the eye: An attention-based method for real-time hand tracking and fingertip detection in egocentric videos. In *Chinese Conference on Pattern Recognition*, pages 145–154. Springer, 2016.

[92] Antonis A Argyros and Manolis IA Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 368–379. Springer, 2004.

[93] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[94] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468. IEEE, 2016.

[95] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 702–715. Springer, 2012.

[96] Srinidhi Hegde, Ramakrishna Perla, Ramya Hebbalaguppe, and Ehtesham Hassan. Gestar: Real time gesture interaction for ar with egocentric view. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 262–267. IEEE, 2016.

[97] Shreyash Mohatta, Ramakrishna Perla, Gaurav Gupta, Ehtesham Hassan, and Ramya Hebbalaguppe. Robust hand gestural interaction for smartphone based ar/vr applications. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 330–335. IEEE, 2017.

[98] J Likitlersuang and J Zariffa. Arm angle detection in egocentric video of upper extremity tasks. In *World Congress on Medical Physics and Biomedical Engineering, June 7-12, 2015, Toronto, Canada*, pages 1124–1127. Springer, 2015.

[99] Alejandro Betancourt, Pietro Morerio, Emilia Barakova, Lucio Marcenaro, Matthias Rauterberg, and Carlo Regazzoni. Left/right hand segmentation in egocentric videos. *Computer Vision and Image Understanding*, 154:73–81, 2017.

[100] Grégory Rogez, James S Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4325–4333, 2015.

[101] Gerald Baulig, Thomas Gulde, and Cristóbal Curio. Adapting egocentric visual hand pose estimation towards a robot-controlled exoskeleton. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[102] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217. IEEE, 2009.

[103] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *3dim*, volume 1, pages 145–152, 2001.

[104] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4866–4874, 2017.

[105] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[106] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7291–7299, 2017.

[107] Hongyong Song, Weijiang Feng, Naiyang Guan, Xuhui Huang, and Zhigang Luo. Towards robust ego-centric hand gesture analysis for robot control. In *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*, pages 661–666. IEEE, 2016.

[108] Nadia Brancati, Giuseppe Caggianese, Maria Frucci, Luigi Gallo, and Pietro Neroni. Robust fingertip detection in egocentric vision under varying illumination conditions. In *2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE, 2015.

[109] Hyung Jin Chang, Guillermo Garcia-Hernando, Danhang Tang, and Tae-Kyun Kim. Spatio-temporal hough forest for efficient detection–localisation–recognition of fingerwriting in egocentric camera. *Computer Vision and Image Understanding*, 148:87–96, 2016.

[110] Yichao Huang, Xiaorui Liu, Lianwen Jin, and Xin Zhang. Deepfinger: A cascade convolutional neuron network approach to finger key point detection in egocentric vision with mobile camera. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pages 2944–2949. IEEE, 2015.

[111] Wenbin Wu, Chenyang Li, Zhuo Cheng, Xin Zhang, and Lianwen Jin. Yolse: Egocentric fingertip detection from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 623–630, 2017.

[112] Thomas Feix, Javier Romero, Heinz-Bodo Schmiedmayer, Aaron M Dollar, and Danica Kragic. The grasp taxonomy of human grasp types. *IEEE Transactions on Human-Machine Systems*, 46(1):66–77, 2015.

[113] De-An Huang, Minghuang Ma, Wei-Chiu Ma, and Kris M Kitani. How do we use our hands? discovering a diverse set of common grasps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 666–675, 2015.

[114] Thomas Feix, Roland Pawlik, Heinz-Bodo Schmiedmayer, Javier Romero, and Danica Kragic. A comprehensive grasp taxonomy. In *Robotics, science and systems: workshop on understanding the human hand for advancing robotic manipulation*, volume 2, pages 2–3, 2009.

[115] Noriko Kamakura, Michiko Matsuo, Harumi Ishii, Fumiko Mitsuboshi, and Yoriko Miura. Patterns of static prehension in normal hands. *American Journal of Occupational Therapy*, 34(7):437–445, 1980.

[116] Mark R Cutkosky et al. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3):269–279, 1989.

[117] CM Light, PH Chappell, PJ Kyberd, and BS Ellis. A critical review of functionality assessment in natural and prosthetic hands. *British Journal of Occupational Therapy*, 62(1):7–12, 1999.

[118] Ian M Bullock, Joshua Z Zheng, Sara De La Rosa, Charlotte Guertler, and Aaron M Dollar. Grasp frequency and usage in daily household and machine shop tasks. *IEEE Transactions on Haptics*, 6(3):296–308, 2013.

[119] Jia Liu, Fangxiaoyu Feng, Yuzuko C Nakamura, and Nancy S Pollard. A taxonomy of everyday grasps in action. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 573–580. IEEE, 2014.

[120] Minjie Cai, Kris M Kitani, and Yoichi Sato. A scalable approach for understanding the visual structures of hand grasps. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1360–1366. IEEE, 2015.

[121] Minjie Cai, Kris M Kitani, and Yoichi Sato. Understanding hand-object manipulation with grasp types and object attributes. In *Robotics: Science and Systems*, volume 3. Ann Arbor, Michigan;, 2016.

[122] Yizhou Lin, Gang Hua, and Philippos Mordohai. Egocentric object recognition leveraging the 3d shape of the grasping hand. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 746–762. Springer, 2014.

[123] Mohamad Baydoun, Alejandro Betancourt, Pietro Morerio, Lucio Marcenaro, Matthias Rauterberg, and Carlo Regazzoni. Hand pose recognition in first person vision through graph spectral analysis. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1872–1876. IEEE, 2017.

[124] Yinlin Li, Yuren Zhang, Hong Qiao, Ken Chen, and Xuanyang Xi. Grasp type understanding—classification, localization and clustering. In *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, pages 1240–1245. IEEE, 2016.

[125] Daniel Thalmann, Hui Liang, and Junsong Yuan. First-person palm pose tracking and gesture recognition in augmented reality. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics*, pages 3–15. Springer, 2015.

[126] Youngkyoon Jang, Ikbeom Jeon, Tae-Kyun Kim, and Woontack Woo. Metaphoric hand gestures for orientation-aware vr object manipulation with an egocentric viewpoint. *IEEE Transactions on Human-Machine Systems*, 47(1):113–127, 2016.

[127] Peng Ji, Aiguo Song, Pengwen Xiong, Ping Yi, Xiaonong Xu, and Huijun Li. Egocentric-vision based hand posture control system for reconnaissance robots. *Journal of Intelligent & Robotic Systems*, 87(3-4):583–599, 2017.

[128] Lorenzo Baraldi, Francesco Paci, Giuseppe Serra, Luca Benini, and Rita Cucchiara. Gesture recognition in ego-centric videos using dense trajectories and hand segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 688–693, 2014.

[129] Lorenzo Baraldi, Francesco Paci, Giuseppe Serra, Luca Benini, and Rita Cucchiara. Gesture recognition using wearable vision sensors to enhance visitors' museum experiences. *IEEE Sensors Journal*, 15(5):2705–2714, 2015.

[130] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[131] Heng Wang, Alexander Kläser, Cordelia Schmid, and Liu Cheng-Lin. Action recognition by dense trajectories. 2011.

[132] Alex Kulesza, Ben Taskar, et al. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286, 2012.

[133] Tomasz Malisiewicz, Abhinav Gupta, and Alexei Efros. Ensemble of exemplar-svms for object detection and beyond. 2011.

[134] Huseyin Coskun, Zeeshan Zia, Bugra Tekin, Federica Bogo, Nassir Navab, Federico Tombari, and Harpreet Sawhney. Domain-specific priors and meta learning for low-shot first-person action recognition. *arXiv preprint arXiv:1907.09382*, 2019.

[135] Tatsuya Ishihara, Kris M Kitani, Wei-Chiu Ma, Hironobu Takagi, and Chieko Asakawa. Recognizing hand-object interactions in wearable camera videos. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1349–1353. IEEE, 2015.

[136] Minjie Cai, Feng Lu, and Yue Gao. Desktop action recognition from first-person point-of-view. *IEEE Transactions on Cybernetics*, 49(5):1616–1628, 2018.

[137] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):500–513, 2010.

[138] Yin Li, Zhefan Ye, and James M Rehg. Delving into egocentric actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 287–295, 2015.

[139] Minghuang Ma, Haoqi Fan, and Kris M Kitani. Going deeper into first-person activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1894–1903, 2016.

[140] Georgios Kapidis, Ronald Poppe, Elsbeth van Dam, Lucas Noldus, and Remco Veltkamp. Multitask learning to improve egocentric action recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[141] Itsaso Rodríguez-Moreno, José María Martínez-Otzeta, Basilio Sierra, Igor Rodriguez, and Ekaitz Jauregi. Video activity recognition: State-of-the-art. *Sensors*, 19(14):3160, 2019.

[142] Taejin Ha, Steven Feiner, and Woontack Woo. Wearhand: Head-worn, rgb-d camera-based, bare-hand user interface with visually enhanced

depth perception. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 219–228. IEEE, 2014.

[143] Eren Erdal Aksoy, Mohamad Javad Aein, Minija Tamosiunaite, and Florentin Wörgötter. Semantic parsing of human manipulation activities using on-line learned models for robot imitation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2875–2882. IEEE, 2015.

[144] Youngkyoon Jang, Seung-Tak Noh, Hyung Jin Chang, Tae-Kyun Kim, and Woontack Woo. 3d finger cape: Clicking action and position estimation under self-occlusions in egocentric viewpoint. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):501–510, 2015.

[145] Jeongmin Yu, Seungtak Noh, Youngkyoon Jang, Gabyong Park, and Woontack Woo. A hand-based collaboration framework in egocentric coexistence reality. In *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 545–548. IEEE, 2015.

[146] Archie Gupta, Shreyash Mohatta, Jitender Maurya, Ramakrishna Perla, Ramya Hebbalaguppe, and Ehtesham Hassan. Hand gesture based region marking for tele-support using wearables. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 69–75, 2017.

[147] Yichao Huang, Xiaorui Liu, Xin Zhang, and Lianwen Jin. A pointing gesture based egocentric interaction system: Dataset, approach and application. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition workshops*, pages 16–23, 2016.

[148] Jangwon Lee and Michael S Ryoo. Learning robot activities from first-person human videos using convolutional future regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–2, 2017.

[149] Jirapat Likitlersuang, Elizabeth R Sumitro, Pirashanth Theventhiran, Sukhvinder Kalsi-Ryan, and José Zariffa. Views of individuals with spinal cord injury on the use of wearable cameras to monitor upper limb function in the home and community. *The Journal of Spinal Cord Medicine*, 40(6):706–714, 2017.

[150] Svebor Karaman, Jenny Benois-Pineau, Vladislavs Dovgalecs, Rémi Mégret, Julien Pinquier, Régine André-Obrecht, Yann Gaëstel, and Jean-François Dartigues. Hierarchical hidden markov model in detecting activities of daily living in wearable videos for studies of dementia. *Multimedia Tools and Applications*, 69(3):743–771, 2014.

[151] Parisa Rashidi and Alex Mihailidis. A survey on ambient-assisted living tools for older adults. *IEEE Journal of Biomedical and Health Informatics*, 17(3):579–590, 2012.

[152] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2847–2854. IEEE, 2012.

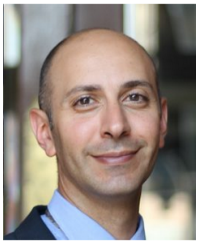
[153] Alejandro Betancourt, Pietro Morerio, Emilia I Barakova, Lucio Marcenaro, Matthias Rauterberg, and Carlo S Regazzoni. A dynamic approach and a new dataset for hand-detection in first person vision. In *International conference on Computer Analysis of Images and Patterns*, pages 274–287. Springer, 2015.

[154] Chi Xu, Lakshmi Narasimhan Govindarajan, and Li Cheng. Hand action detection from ego-centric depth sequences with error-correcting hough transform. *Pattern Recognition*, 72:494–503, 2017.

[155] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018.

[156] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 619–635, 2018.

Andrea Bandini Andrea Bandini (M'16) received his Master's degree in Biomedical Engineering from the University of Firenze (Italy) in 2012, and the PhD the Bioengineering from the University of Bologna (Italy) in 2016. He has been a postdoctoral research fellow at KITE - University Health Network (Toronto, Canada) since September 2016. His research aims at developing intelligent tools for remote assessment and rehabilitation of motor signs associated with neurological disorders (spinal cord injury, stroke, amyotrophic lateral sclerosis, and Parkinson's disease), by using computer vision and machine learning techniques.



José Zariffa José Zariffa (M'01, SM'18) received the Ph.D. degree in 2009 from the University of Toronto's Department of Electrical and Computer Engineering and the Institute of Biomaterials and Biomedical Engineering. He later completed post-doctoral fellowships at the International Collaboration On Repair Discoveries (ICORD) at the University of British Columbia in Vancouver, Canada, and at the Toronto Rehabilitation Institute – University Health Network in Toronto, Canada. He is a currently a Scientist at KITE - Toronto Rehabilitation Institute - University Health Network and an Associate Professor at the Institute of Biomaterials and Biomedical Engineering at the University of Toronto in Toronto, Canada. His research interests include technology for upper limb rehabilitation after spinal cord injury, neural prostheses, and interfaces with the peripheral nervous system. Dr. Zariffa is the recipient of an Ontario Early Researcher Award.