ASTES

Advances in Science, Technology and Engineering Systems Journal
Vol. 5, No. 4, 144-159 (2020)
www.astesj.com

ASTES Journal
ISSN: 2415-6698

A Survey on 3D Hand Skeleton and Pose Estimation by Convolutional Neural Network

Van-Hung Le[*,1], Hung-Cuong Nguyen[2]

[1]*Tan Trao University, 22000, Vietnam*

[2]*Faculty of Technology and Engineering, Hung Vuong University, 35000, Vietnam*

ARTICLE INFO

Article history:
Received: 29 March, 2020
Accepted: 07 July, 2020
Online: 18 July, 2020

Keywords:
3D Hand Skeleton Estimation
3D Hand Pose Estimation
Convolutional Neural Network

ABSTRACT

Restoring, estimating the fully 3D hand skeleton and pose from the image data of the captured sensors/cameras applied in many applications of computer vision and robotics: human-computer interaction; gesture recognition, interactive games, Computer-Aided Design (CAD), sign languages, action recognition, etc. These are applications that flourish in Virtual Reality and Augmented Reality (VR/AR) technologies. Previous survey studies focused on analyzing methods to solve the relational problems of hand estimation in the 2D and 3D space: Hand pose estimation, hand parsing, fingertip detection; List methods, data collection technologies, datasets of 3D hand pose estimation. In this paper, we surveyed studies in which Convolutional Neural Networks (CNNs) were used to estimate the 3D hand pose from data obtained from the cameras (e.g., RGB camera, depth(D) camera, RGB-D camera, stereo camera). The surveyed studies were divided based on the type of input data and publication time. The study discussed several areas of 3D hand pose estimation: (i)the number of valuable studies about 3D hand pose estimation, (ii) estimates of 3D hand pose when using 3D CNNs and 2D CNNs, (iii) challenges of the datasets collected from egocentric vision sensors, and (iv) methods used to collect and annotate datasets from egocentric vision sensors. The estimation process followed two directions: (a) using the 2D CNNs to predict 2D hand pose, and (b) using the 3D synthetic dataset (3D annotations/ ground truth) to regress 3D hand pose or using the 3D CNNs to predict the immediacy of 3D hand pose. Our survey focused on the CNN model/architecture, the datasets, the evaluation measurements, the results of 3D hand pose estimation on the available. Lastly, we also analyze some of the challenges of estimating 3D hand pose on the egocentric vision datasets.

1 Introduction

A few recent years, Virtual Reality (VR) and Augmented Reality (AR) become promising technologies in human life. Based on computer vision techniques, they could be found in many applications, including human-computer interaction [1]-[2]; gesture recognition [3, 4]; interactive games [5]; Computer-Aided Design (CAD) [6], sign languages [7]; action recognition, etc. In those applications, real pictorial data (e.g. depth image, color image, stereo, RGB-D, and point cloud data as illustrated in Fig. 1) will be transformed into computer-based data and then could be used by the algorithms. The important work of VR/AR software is detecting the object in the environment from those data. To resolve this point, estimating the 3D skeleton of hand by the Convolutional Neural Networks (CNNs), as shown in Fig. 1 is a widely considered approach with more than 60 valuable studies over the last 4 years.

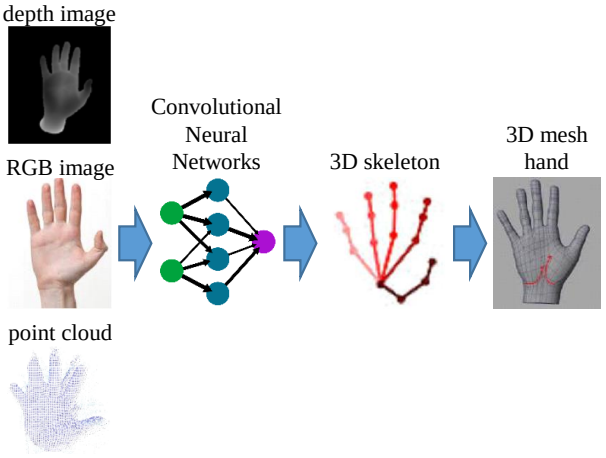
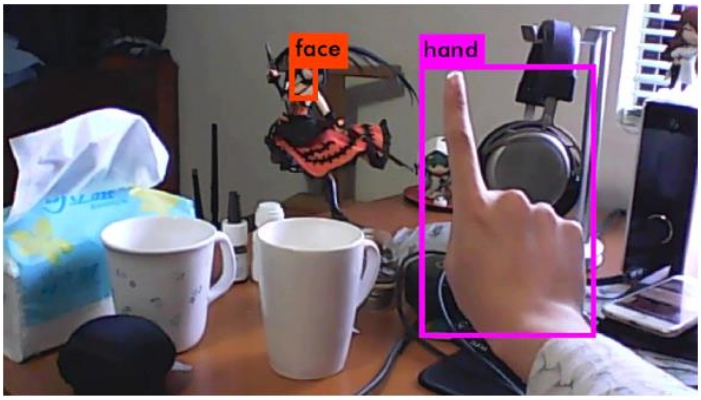
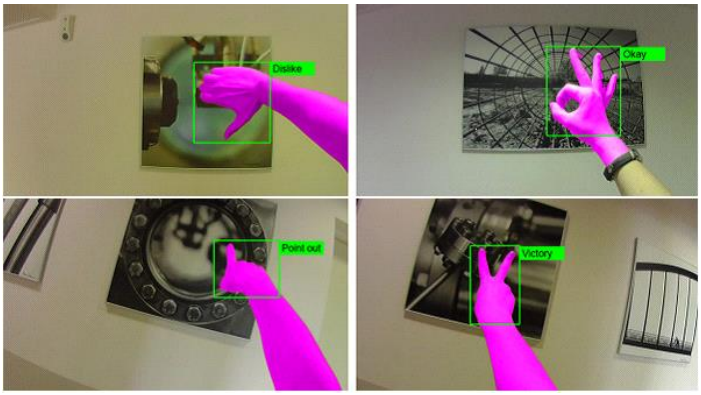


Figure 1: Illustrating the typical input data of 3D hand pose estimation and the results.

[*]Van-Hung Le, Tan Trao University, & Lehung231187@gmail.com



(a)



(b)

Figure 2: (a) the result of hand detection [8], (b) the result of the hand segmentation for Gesture Recognition in egocentric vision [9].

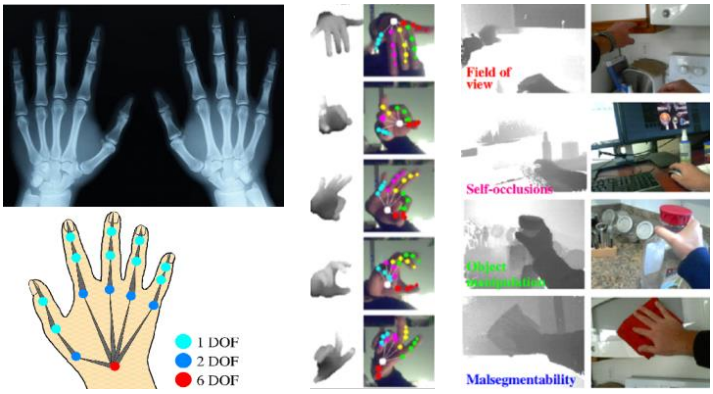


Figure 3: Top left: Hand anatomy; Bottom left: the kinematic model [10]; Right: 3D hand skeleton and pose estimation on the RGB-D image of the egocentric sensor [11].

With the strong development of the sensor/camera technology, with the appearance of depth sensors (e.g. MS Kinect v1 [12], [13], MS Kinect v2 [14], 3D Prime Sense Sensors [15], Intel Real Sense [16], Leap Motion, etc) made restoring 3D hand skeleton and pose easier and more accurate. However, the results of estimating, restoring 3D hand skeleton and pose are influenced by several factors from the captured image data such as hand motion, severe self-occlusion and self-similarity of fingers, especially the data of hand is obscured when collected from an egocentric sensor [10], [11] as Fig. 3, before conducting further research on 3D hand detection, recognition, and full estimation of 3D hand skeleton, pose. More specific when estimating and restoring the full hand skeleton and pose in the 3D space will help recognize the grasp types, grasp attributes, object attributes [17], [18] of the object more accurately, especially can evaluate the ability to activate of the fingers [19], [20]. In this paper, we survey of the methods and results of 3D hand pose, skeleton estimation following the type input data and publication time, we only focus on the approach that applies the CNNs to estimate 3D hand pose. The input data of 3D hand pose estimation methods can be the depth image, color image, stereo, RGB-D, and point cloud data. They are illustrated in Fig. 1. In particular, we also discuss the results of methods using 3D CNNs, 2D CNNs to estimate the location of joints in the 3D space following the four issues in the 3D hand pose estimation process: The number of valuable studies about 3D hand pose estimation; The estimated results of 3D hand pose when using 3D CNNs and 2D CNNs; The challenges of the datasets which is collected from egocentric vision sensors; The methods to collect and annotate datasets from egocentric vision sensors.

The rest of the paper is organized as follows: Section 1 introduces some overview of this paper. Section 2 discusses the related work. Section 3 discusses 3D hand pose estimation by CNNs to estimate 3D hand pose from some types of data, including the depth image (Sub-section 3.1), the RGB image (Sub-section 3.2), the RGB-D image, or other camera data (Sub-section 3.3). Section 4 presents, discusses some results of 3D hand pose estimation by the CNNs. Section 5 discusses the datasets (Sub-section 5.1) and challenges (Sub-section 5.2) for 3D hand pose estimation. Section (6) concludes the paper with future work.

2 Related Works

Many studies of estimating and restoring the full 3D hand model, i.e. skeleton and pose, have been published in recent years. Some of them are listed comprehensively in the survey of Li et al. [21]. This paper provides the answers to many questions, including *"What do we need to estimate of the hand?"*, *"What entangles do we need to overcome?"*, *"What is the depth sensor?"*, *"What are the useful methods?"*: **The objective of 3D hand estimation** is hand detection, hand tracking, hand parsing, fingertip detection, hand contour estimation, hand segmentation, gesture recognition, etc. **The challenges of estimating hand pose** are low resolution, self-similarity, occlusion, incomplete data, annotation difficulties, hand segmentation, real-time performance. **Existing depth sensors** are also summarized by Li et al. [21], including 19 popular depth sensors produced in the last decade. They are divided into groups and illustrated in Fig. 4. The considered parameters of those sensors are depth technology, measurement range, and a maximum speed of depth data. **The methods** to solve 3D hand pose estimation are the model-based method, appearance-based method, and hybrid method. It can be considered as an extension of a review written by Erol et al. [22] that introduced two main methods (i.e. model-based method and appearance-based method) to solve this problem in the 2D space:

- The model-based method compares the hypothetical hand

Figure 4: The depth sensor groups: (1) MS Kinect group; (2) ASUS Xtion group; (3) Leap Motion; (4) Intel RealSense group; (5) SoftKinetic group; (6) Creative Interactive Gesture; (7) Structure Sensor.

pose and the actual data obtained from the cameras. The comparison is evaluated based on an objective function that measures the discrepancy between the actual observations and the estimated data that are generated from the model of the hand.

- The appearance-based method based on learning the characteristics of from the observations to a discrete set of the annotated hand poses. This method uses a discriminative classifier or regression model to describe invariant characteristics of hand pose as a map of the joints of the fingers.

However, the survey of Li et al. [21] is listed only without the presentation of methods, datasets, and evaluation methods.

Another good survey of hand pose estimation is taken by Barsoum [23]. In this study, the author also discusses three methods to perform hand pose estimation from the depth image. The appearance-based method is shown in Fig. 3, the model-based method, the hybrid method is mentioned in Fig. 2. Barsoum focuses on the hand segmentation because its outcome affects the accuracy of algorithms. Facing this problem, the discussed methods are Color or IR skin based; Temperature-based; Marker-based; Depth based and Machine learning-based. In more detail, the author presents the limitation of applying deep learning in hand pose estimation with only two publications ([24] and [25]) in two years 2014 - 2015. From point of view of the limitations as mentioned before, we summaries a survey about the state of the art of hand pose estimation that uses Deep Learning (DL) / Convolutional Neural Network (CNN) in recent years.

3 3D Hand Pose Estimation by CNNs

In recent years, using CNNs in detection, recognization, and estimation objects is one of the most successful approaches in computer vision. Human hands are used in the applications of VR/AR and human-computer interaction because human hands can create many different states to execute control states. As Fig. 3(bottom left) is shown 26 degrees of freedom (DOF). To build control and interaction applications using human hands, firstly, human hands need to be fully and accurately estimated joints in the 3D space. Therefore, this issue is interested in research, especially with the success of CNN in computer vision.

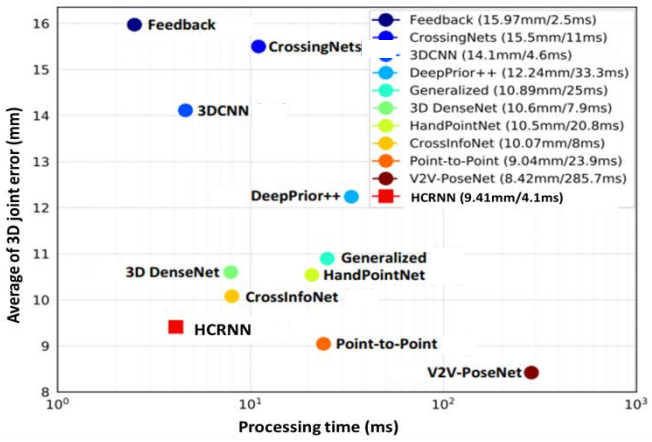


Figure 5: The results of some typical CNNs on the NYU dataset [26].

Firstly, we conducted a survey of methods, results and, discussions of estimating 3D hand pose by CNNs based on the type input data and publication time *(i)*. There are 60 studies in the period 2015 - 2019 as shown in Tab. 1. Some of the prominent results are illustrated in Fig. 5. As presented in Tab. 2, the studies in Tab. 1 is published in leading conferences and journals in the field of computer vision. The input data of CNNs to estimate 3D hand pose are the color image, the depth image, and the point cloud. These are data sources that can be collected from common image sensors. Therefore, building VR/AR and human-computer interaction applications in the 3D space can use low-cost sensors and have accurate results (average of 3D joint error 8-16mm as shown in Fig. 5).

The estimated methods [23], i.e. discriminative method, generative method, and hybrid method, are shown in Fig. 6. In the next sub-sections, we give more details on approaches using the CNNs to estimate 3D hand pose from various input data.

Table 1: Statistics of the number of studies used CNNs for 3D hand pose estimation.

Author	**CNN type**			**Data type**						**Approach**				**Publish**
	2D	3D	No	Depth	RGB	RGB-D	Stereo	Point cloud	Gray image	appearance based	hybrid based	model based	data set	
2015 (8 publications)														
Oberweger[25]	✓			✓						✓				CVWW
Choi[27]	✓					✓				✓				ICCV
Poier[28]	✓			✓							✓			BMVC
Li[29]	✓			✓						✓				ICCV
Oberweger[30]	✓			✓						✓				ICCV
Sun[31]	✓			✓						✓				CVPR
Tang[32]	✓			✓								✓		ICCV
Oberweger[33]	✓			✓						✓				TPAMI
2016 (7 publications)														
Wan[34]			✓	✓								✓		ECCV
Ye[35]	✓			✓							✓			ECCV
Sinha[36]	✓					✓				✓				CVPR
Oberweger[37]	✓			✓						✓				CVPR
Xu[38]			✓	✓						✓				IJCV
Zhang[39]	✓			✓			✓			✓				Arxiv
Ge[40]	✓			✓						✓				TIP
2017 (12 publications)														
Deng[41]		✓		✓						✓				Arxiv
Yuan[42]	✓			✓						✓				CVPR
Choi[43]	✓			✓						✓				ICCV
Choi[44]	✓			✓						✓				ICCV
Wan[45]	✓			✓						✓				CVPR
Ge[46]		✓		✓						✓				CVPR
Neverova[47]	✓			✓						✓				Arxiv
Mueller[10]	✓					✓				✓				ICCV
Zhang[48]	✓			✓						✓				Arxiv
Malik[49]	✓			✓						✓				Arxiv
Zimmermann[50]	✓				✓					✓				ICCV
Oberweger[51]	✓			✓						✓				ICCV
2018 (18 publications)														
Baek[52]	✓			✓						✓				CVPR
Wu[53]		✓		✓							✓			TOC
Supancic[54]			✓	✓						✓				ICCV
Madadi[55]	✓			✓						✓				Arxiv
Rad[56]	✓			✓	✓					✓				CVPR
Garcia[57]			✓			✓							✓	CVPR
Ge[58]		✓		✓						✓				TPAMI
Chen[59]	✓				✓					✓				Arxiv
Zhang[60]	✓			✓						✓				VIPC
Wohlke[61]	✓			✓						✓				Arxiv
Moon[62]		✓		✓						✓				CVPR
Ye[63]	✓			✓							✓			ECCV
Chen[64]		✓						✓		✓				Access
Spurr[65]	✓				✓					✓				CVPR
Huang[66]		✓		✓						✓				BMVC
Penteleris[67]	✓				✓					✓				WACV
Wan[68]	✓	✓								✓				CVPR
Ge[69]		✓		✓						✓				ECCV
2019 (15 publications)														
Zhang[70]	✓				✓					✓				Arxiv
Sharma[71]	✓				✓					✓				Arxiv
Yoo[72]	✓			✓						✓				Arxiv
Li[73]	✓								✓	✓				ICCV
Wan[74]	✓			✓						✓				Arxiv
Li [21]	✓						✓			✓				BMVC
Liu[75]	✓				✓					✓				TPAMI
Cejong[76]	✓			✓						✓				FG
Hampali[77]			✓			✓							✓	CVPR
Li[78]		✓						✓		✓				CVPR
Zhang[79]	✓			✓						✓				TIP
Baek[80]	✓				✓					✓				CVPR
Lee[81]	✓			✓						✓				Arxiv
Ge[82]	✓				✓					✓				CVPR
Du[83]	✓			✓						✓				CVPR
Total	**46**	**10**	**5**	**41**	**10**	**5**	**2**	**2**	**1**	**52**	**4**	**2**	**2**	

Table 2: The explanation the names of conferences and journals in Tab. 1.

Acronym	Explanation
CVWW	Computer Vision Winter Workshop
ICCV	IEEE International Conference on Computer Vision
BMVC	British Machine Vision Conference
CVPR	IEEE Conference on Computer Vision and Pattern Recognition
TPAMI	IEEE Transactions on Pattern Analysis and Machine Intelligence
ECCV	European Conference on Computer Vision
IJCV	International Journal of Computer Vision
TIP	IEEE Transactions on Image Processing
Arxiv	arxiv.org
TOC	IEEE Transactions On Cybernetics
VIPC	Electronic Imaging, Visual Information Processing and Communication
Access	IEEE Access
WACV	IEEE Winter Conference on Applications of Computer Vision
FG	IEEE International Conference on Automatic Face & Gesture Recognition

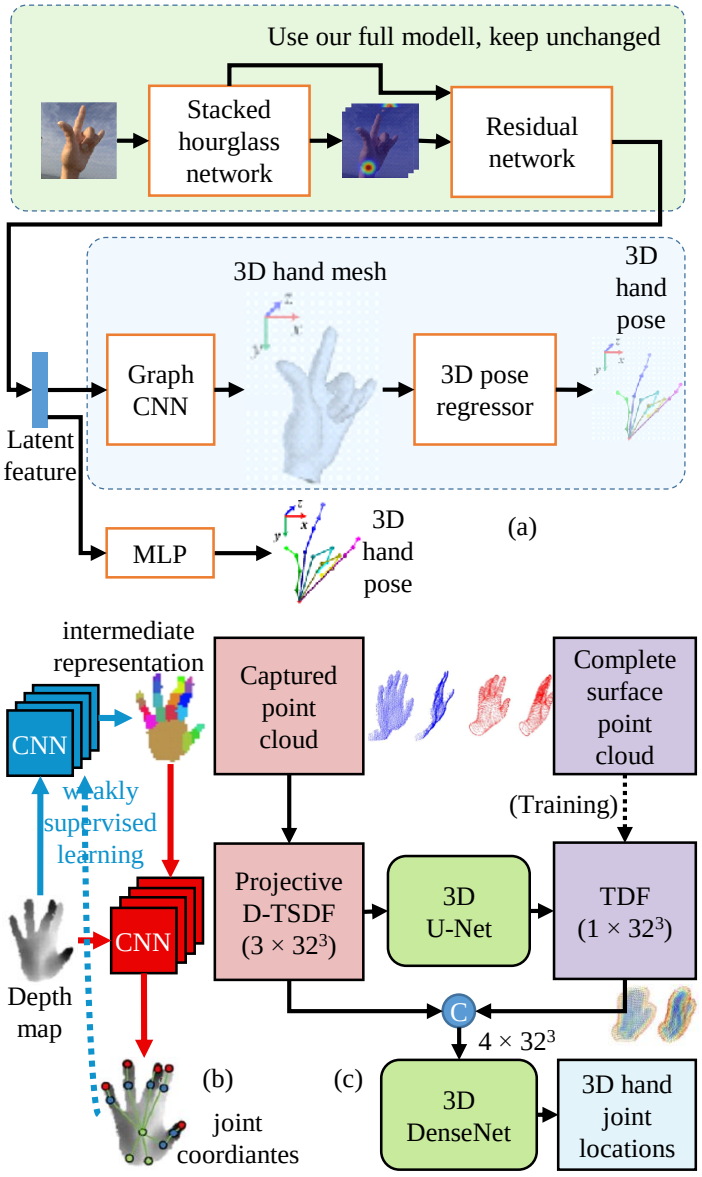


Figure 6: CNN architectures to estimate 3D hand pose as follows: (a) the RGB image [82]; (b) the depth image [47]; (c) the point cloud [58].

3.1 *Estimating by The Depth Image*

Being the origin format of 3D data, the depth image is the most widely used when estimating a 3D hand pose. There are 2 branches as illustrated in Fig. 7, based on the form of intermediate data, i.e. hand point cloud and heat-map.

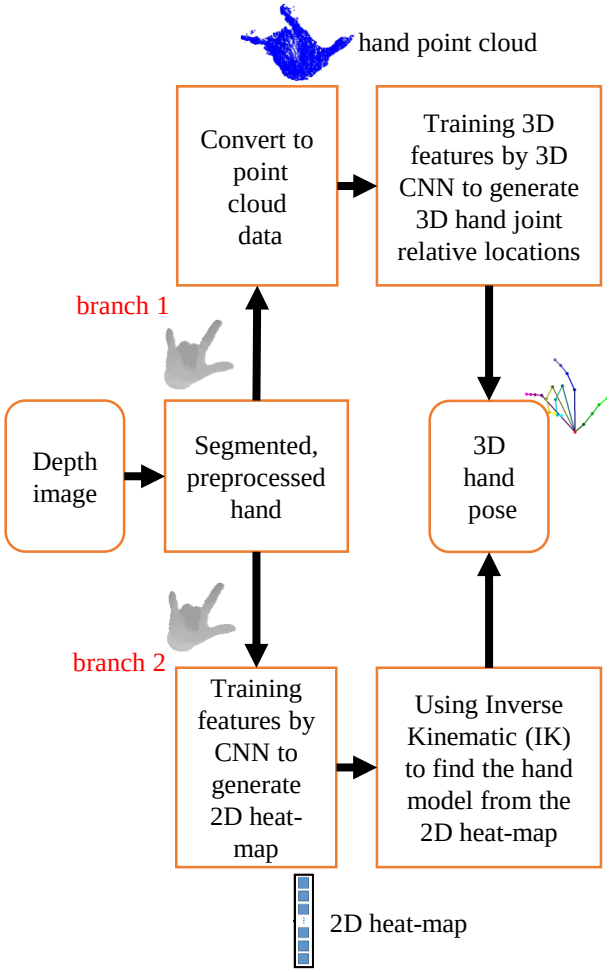


Figure 7: The CNN architecture to estimate 3D hand pose from the depth image.

3.1.1 Converting to Hand Point Cloud Data

The first branch of depth image approach converts collected depth data to the form of point cloud before putting it into CNN as a training data [40] [58]. Liuhao et al. [40] proposed a multi-view regression framework for 3D hand pose estimation, as is shown in Fig. 8. This framework generates heat-maps for three views by projecting the point cloud of the hand onto three orthogonal planes, i.e. $(x-y; y-z; z-x)$. Then each projected image is fed into a separate CNN to generate a set of heat-maps for hand joints. This method is similar to the method of [26] to generate a set of heat-maps. After that, the combination of those three views thus contains the location distribution of the joint in the 3D space. This proposed method was evaluated by the dataset of [84]. The average estimation errors are 22.8mm and the processing time is 14.1ms when to be trained and tested on the GPUs under the system whose two Intel Xeon processors, 64GB of RAM and two Nvidia Tesla K20 GPUs. The details of time are 2.6ms for multi-view projection, 6.8ms for CNN forward propagation, and 4.7ms for multi-view fusion.

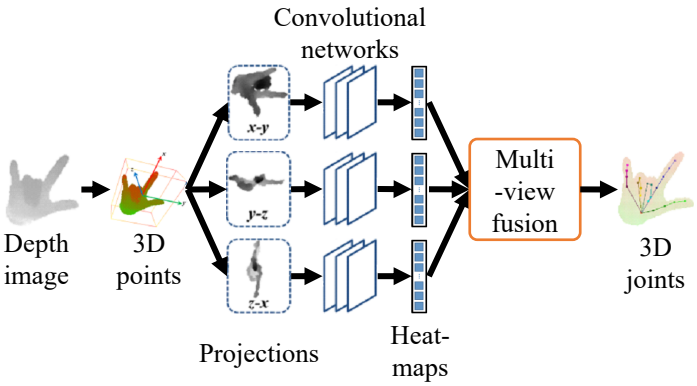


Figure 8: Multi-view regression framework for 3D hand pose estimation [40].

Wan et al. [34] proposed a Conditional Regression Forest (FCRF) which uses a set of new features. At each stage of the regression, the frame of reference is established from either the local surface normal or previously estimated hand joints of the point cloud. The normal difference feature of this method is highly robust to 3D rigid transformation because the 2.5D point cloud is projected and indexed to the image space. Therein, the hand pose estimation process is the process of estimating the joints of 5 fingers. The proposed method is evaluated on ICLV and MSRA datasets and the result of the average of joints error is about 8mm, 25mm-30mm, respectively. Ge et al. [46] proposed a simple approach for real-time 3D hand pose estimation from single depth images by using three-dimensional CNNs (3D CNNs). This 3D CNNs can effectively learn 3D features from the 3D volumetric representation. Liuhao et al. [58] proposed Hand PointNet-based method for 3D hand pose estimation. The 3D point cloud of the hand is down-sampled and normalized in an oriented bounding box (OBB) to make the proposed method robust to various hand orientations. This method uses the estimated surface normal and normalized points of the point cloud data of the hand as the input of the hierarchical PointNet [85] and then outputs a low dimensional representation of the 3D hand joint locations. Therein, the hierarchical PointNet consists of L point set abstraction levels. The higher the level is, the smaller the number of points. The authors evaluated the proposed method on three public hand pose datasets, including NYU [26], MSRA [84], and ICVL [86]. The experimental results when deploying in a workstation with two Intel Core i7 5930K, 64GB of RAM and an Nvidia GTX1080 GPU are:

- The per-joint mean error distances and the overall mean error distances are 10.5mm, 8.5mm, and 6.9mm, respectively.
- The average processing time of the proposed method is 20.5ms, including 8.2ms for point sampling and surface normal calculation, 9.2ms for the hand pose regression network forward propagation, 2.8ms for fingertip neighboring points search, and 0.3ms for fingertip refinement network forward propagation.

3.1.2 Training by CNNs to Generate 2D Heat-map

The second approach based on depth image often trains annotated joints of hand poses on a large dataset (synthesized data) by the CNNs [45]. Those datasets contain most of the actual hand poses. The estimation process evaluates the characteristics of the hand pose on the input data and finds the most fitting pose in the synthesized data as illustrated in Fig. 9. Many studies are using this method whose difference is the used of CNN to predict the position of joints. Fig. 5 illustrates the results of some prominent studies with the NYU dataset.

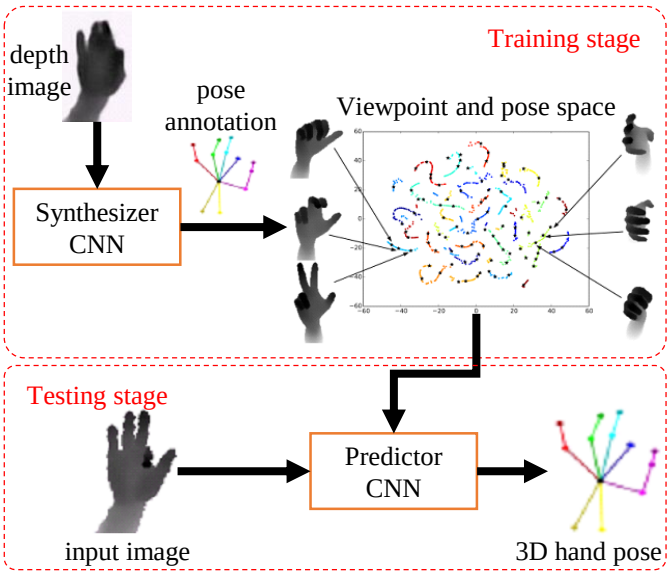


Figure 9: The estimation model of 3D hand pose from the depth image based on training and predicting the position of joints on depth images.

Oberweger et al. [30] proposed a model called "*feedback loop*". This model includes Deep Networks and is optimized to use training data. This model is capable of updating the estimated hand pose and provides the experimental results with the NYU dataset as in Fig. 5. Zhang et al. [39] introduced a method for estimating the 3D pose of the human hand, mouse, and fish from the depth images. This method used CNN to predict joint locations that are represented in the manifold space by Lie group, i.e. each joint of the skeleton is represented in the manifold space by $SE(3)$. Five fingers are

modeled by five-subchains. And then, those chains are connected to the palm center joint and integrated into a kinematic tree as the hand skeletal model. There are two variants of this model called "*deep-L2S-para*" and "*deep-L2S-seq*", whose experimental results with the NYU dataset are 15.84mm and 14.15mm, respectively. Wan et al. [45] proposed a dual generative model that captures the latent spaces of hand poses. This model uses the variation auto-encoder (VAE) and the generative adversarial network (GAN) for estimating 3D hand pose. In more detail, this model generates the synthesized realistic depth maps of highly articulated hand poses under dramatic viewpoint changes and reduces the number of annotated training data. The model of Neverova et al. [47] allows extracting information automatically from real data by deploying a semi-supervised and weakly-supervised training algorithm. These two learning methods are trained from two different datasets, i.e. the synthetic and the real dataset. This method aims to the objective by which to perform frame-by-frame without any dynamic information. The average of 3D joints error is 14.8mm. Authors verify that this method is better than some other methods like DeepPrior [25], Hand3D [41], Crossing nets [45], etc.

The CNN model of Choi et al. [36] [44] uses paired depth images. Firstly, the position of the hand and the object in the image are determined by using CNN to predict the heat-maps. And then, those heat-maps are projected into the space of 3D hand and CAD models of the synthetic dataset. At the same time, a synthetic dataset of human grasps is also built. The next, authors then classify the hand orientations and grasp type from the multi-channel network to reduce the search space for pose estimation. The model is trained by the synthetic dataset whose number of images is 16.5K. Each grasp is captured by 500 depth maps that are rendered randomly from different objects, orientations, and backgrounds. Being evaluated additionally by a publicly available GUN-714 dataset [87], the average of 3D joints error is smaller 20mm.

In the study of Baek et al. [52], the corresponding ground-truth hand poses annotations, and the skeleton entries are the input depth maps of the training stage. The skeleton entries of each dataset are generated from separate hand pose generator (HPG) and 3D hand pose estimator (HPE). This is because the training on input depth maps and the corresponding ground-truth hand pose annotations are not enough to cover variations in poses, shapes, views, etc. CNN is trained by the skeletal hand shape model of the Big Hand 2.2M dataset [42]. The number of added skeletal poses is greater than the number of existing datasets, i.e. Big Hand 2.2M, ICVL, NYU, and MSRA. In more detail, with the Big Hand 2.2M dataset, the average of joints error reduces from 17.1mm to 12.5mm.

Madadi et al. [55] used a novel hierarchical tree-like structured CNN, whose branches are trained to become specialized in predefined subsets of hand joints, called "*local poses*". Being extracted from hierarchical CNN branches, local pose features are fused to learn higher-order dependencies among joints in the final pose by end-to-end training. Especially, the used loss function is also defined to incorporate appearance and physical constraints about double hand motion and deformation. This function is used to optimize network parameters during training and regression stages. The averages of joints error are 11.0mm and 9.7mm when evaluated by NYU and MSRA datasets, respectively. Rad et al. [56] use a Deep Network to predict a 3D pose from an image. This Deep Network is trained by the features that are computed for a real image and in a synthetic image of the same pose. The average result of 3D joints error of this approach with the NYU dataset is 7.4mm. Zhang et al. [60] used the cascaded hierarchical regression in [31] to get rough locations of hand joints and proposed a refinement stage to re-estimate joint locations of stretching-out fingers. Therein, the authors used the method in [88] to predict the key joints localization. Evaluating by MSRA and ICVL datasets, the average errors for estimating all fingers are 18.02mm and 13.65mm, respectively. For estimating all fingertips, there are 20.12mm and 14.30mm, respectively. Considering the physical constraints of human hand kinematics, Wohlke et al. [61] proposed a hybrid approach that has embedded a kinematic layer into the CNN. The size of the input image is standardized over BoxNet, RotNet, and ScaleNet whose size is 176×176 pixels. The residual network [89] is used to estimate hand parameters and a kinematic hand model layer (FKINE) forwards kinematics from hand parameters to joint locations. The hand has 61 parameters. The average of 3D joints error is 11mm with the NYU dataset.

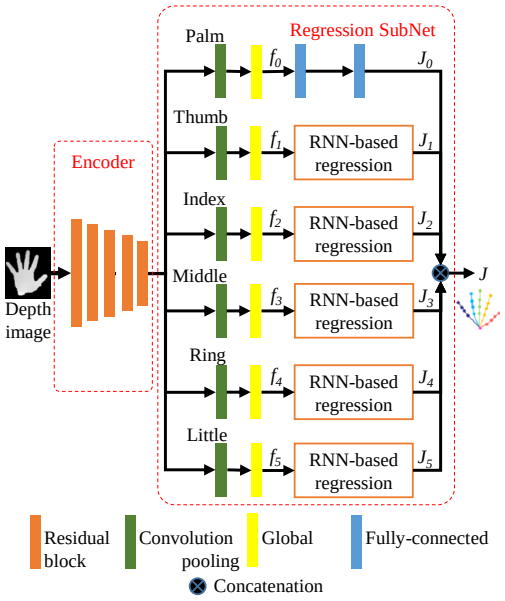


Figure 10: The HCRNN architecture for 3D hand pose estimation from a single depth image [72].

In the past year, the depth image also is used to estimate 3D hand pose in some studies. Yoo et al. [72] divide the hand into six parts, i.e. the palm and five fingers, as in Fig. 10. Then authors proposed a hierarchically-structured convolutional recurrent neural network (HCRNN) with six branches that correspond with those parts. This study exploits effectively the 2D characteristics of the depth image as input of the CNNs. Due to each branch of CNN trains and predicts a part of the hand, this approach has a very fast processing time, up to 240 fps on a single GPU. Being evaluated on the ICVL, NYU and MSRA datasets, the average of 3D joints error is 6.6mm, 9.4mm, and 7.8mm, respectively.

The presented studies are based on two methods [79] as illustrated in Fig. 7, including detection-based method (as the first branch) and regression-based method (as the second one). Facing lose spatial information of hand structure problem and lack direct supervision of joint coordinates problem, a new method of Zhang et

al. [79], called "*Pixel-wise Regression*", use spatial form representation (SFR) and differentiable decoder (DD). The authors explain their method as a combination of the two former above methods. Comparing with the state-of-the-art technique, this method reduces mean 3D joint error by more than 25%. Specifically, 3D joints error on the MSRA dataset is 5.186mm.

3.2 Estimating by RGB Image

As illustrated in Fig. 11, the 3D hand pose estimation process from RGB image usually contains five steps as follows:

- Predicting heat-maps on image space by a CNN.
- Predicting 2D hand pose.
- Training 2D hand pose and the ground truth of 3D hand pose of the synthetic dataset to generate a 3D model.
- Predicting 2D hand pose by real input data.
- Using 3D hand pose estimated model and 2D hand pose of the real data as input data to output 3D hand pose.

A few years ago, i.e. from 2015 to 2017, depth images are usually considered as input data of CNN to estimate 3D hand pose and skeleton. However, depth data is less common than color data in real-life because of the unpopularity and the expensive of depth sensors/cameras. Furthermore, CNN technologies have developed strongly. Therefore, in the last two years, researchers also use RGB images in their studies.

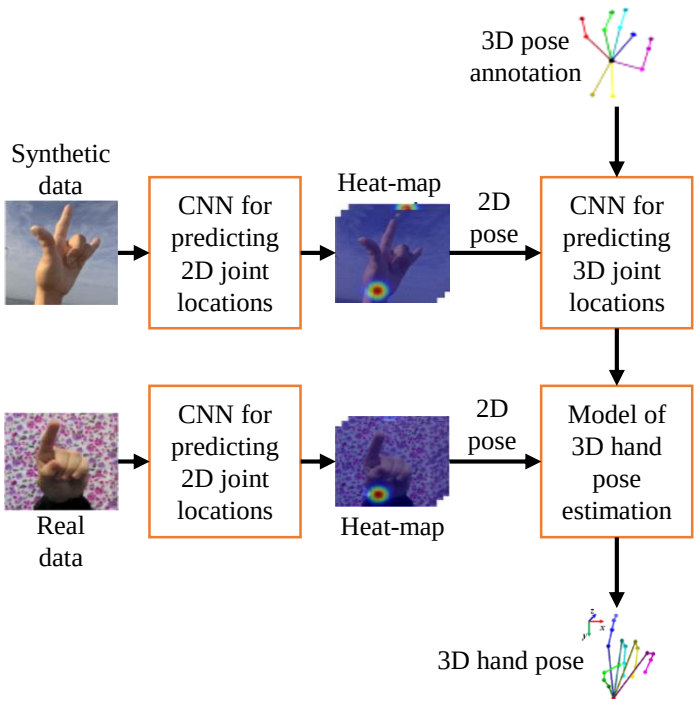


Figure 11: The model of 3D hand pose estimation from the RGB image.

In their article, Zimmermann et al. [50] estimate 3D hand poses from regular RGB images by three deep networks. The first CNN [90] provides a hand segmentation to locate the hand in the image. The second CNN [90] models 2D hand pose in the 2D images. The last CNN [91], [92] predicts the 3D hand pose from this model. Being evaluated on the Percentage of Correct Keypoints (PCK) measurement with the RWTH German Fingerspelling dataset, the result is 32%.

Panteleris et al. [67] recover the 3D hand pose by using least-squares minimization to fits a 3D model of the hand to the estimated 2D joint positions. Those 2D data are generated by the pre-trained network of OpenPose [93] from the detected hand in the image using YOLO v2 [94]. Authors evaluate their method by three datasets, including the Stereo hand pose dataset, Synthetic dataset, and Hands in action RGB-D dataset. The result of error thresholds is less than 30mm.

A method of Spurr et al. [65] generates a 3D hand pose from an RGB image by learning a single unified latent space via an extension of the VAE (Variational AutoEncoders) framework. The data of latent space, i.e. the RGB images and 3D joint configurations, are illustrated by the blue and green colors. The Stereo Hand Pose Tracking Benchmark (STB) and the Rendered Hand Pose Dataset (RHD) datasets are used to evaluate their model.

Chen et al. [59] develop tonality-alignment generative adversarial networks (TAGANs) that have high-quality ability to generate hand pose. The working mechanism of this network is aligning the tonality and color distributions between synthetic hand poses and real backgrounds. However, hand pose datasets are not large enough to learn a stable CNN hand pose estimator. Therefore this method adopted an opensource AR simulator to produce large-scale and high-quality hand pose images with accurate 2D/3D hand-keypoint labels. The authors used the convolutional pose machine (CPM) [90] for predicting and the Hand3D [50] for estimating 3D hand pose. The experimental results are 19.9mm and 7.3mm with RHP and STB datasets, respectively.

The idea of He et al. [95] is a hand-model regularized graph CNN trained under a generative adversarial learning framework (GraphPoseGAN) that contains two modules. The first "*hand model module*" generates a template 3D hand pose as a prior. Its inside encoder extracts the latent code z from the input image and a parametric hand model. The second "*GCN refinement module*" is used to refine 3D hand pose from 3D ground truth to choose a hand pose whose parameter is the best. Being evaluated by Stereo Hand Pose Tracking Benchmark (STB) and the Rendered Hand Pose (RHD) datasets, this model gets the average error in Euclidean space between the estimated 3D joints and the ground truth joints is 12.4mm (RHD) and 4.2mm (STB).

Being introduced at CVPR in 2019, the method of Baek et al. [80] predicts 2D heatmaps from 2D feature extractor and 2D hand mask. After that, the 3D skeleton of the input data is regressed by a 2D skeleton and 3D skeleton of the supervision stage. The used datasets are the Stereo Hand pose Dataset (SHD).

3.3 Estimating by RGB-D Image and Other Data

Choi et a. [27] developed a real-time algorithm to use RGB-D data. This method used the local shape descriptors to retrieve nearest neighbors from the labeled dataset. And then this information is used to evaluate the unknown pose parameters by a joint matrix factorization and completion (JMFC) approach on a hand pose library.

The method of Mueller et al. [10] provides a real-time, robust, and accurate hand pose estimation that uses RGB-D data of egocentric cameras. The collected data is clutter and occlusions. Firstly, this method uses a HALNet CNN to estimate the 2D position of the hand center in the input. And then a generated normalized cropped image is fed into a JORNet CNN to regress relative 3D hand joint locations. Both of those CNNs are trained with the new SynthHands dataset. Being evaluated by the EgoDexter benchmark dataset, the lowest average error is 32.6 mm.

In their research, Iqbal et al. [96] introduce a novel 2.5D pose representation to and a solution to reconstruct the 3D pose from this 2.5D model. In this scaled and invariant representation, the 2D coordinates are the coordinates of the points on the image and the remaining 0.5D is the coordinates of the palms that are predicted from the depth. The average End-Point-Error (EPE) is 25.56mm and 31.86mm with SHP and RHP datasets, respectively. Cejnog et al. [76] used the Pose-REN method [97] to train the Hands2017 dataset.

Extending the success of detecting, identifying, and estimating objects from the color image and depth image, CNNs have been used to work with 3D data, i.e. point cloud. Li et al. [98] proposed a novel CNN for working with an un-organization point cloud data. There are 1024 points in this 3D data. This CNN computes the point-wise features from each point by the PEL (Permutation Equivariant Layer) residual network. And then those features are used by the point-to pose voting to estimate the point of hand pose. By the NYU dataset, the mean joint error is 8.99mm and 8.35mm corresponds with the single view and the three views, respectively. Besides, several CNNs use 3D points as the input data including point-wise CNN [99], Deep KD-Networks [100], Self-Organizing Net [101], and Dynamic Graph CNN [102].

4 Findings/Results

Based on the surveys of 3D hand pose estimation using the CNNs presented in Tab. 1. The second issue discussed in this paper *(ii)* is the results of it when using 2D CNN and 3D CNN. As shown above, the objective of existing methods is the location joints estimation based on the 2D, 2.5D, and 3D data. So there are two types of CNNs, i.e. 2D and 3D, as illustrated in Fig. 12. We collect the results of estimating 3D hand pose by CNNs as in the Tab. 3. The average 3D distance error when using 3D CNN is lower than using 2D CNN. This problem happens because the input data of the 3D CNNs is the 3D data. Therefore, the accuracy of 3D CNN is better than 2D CNN, as shown in Tab. 3. When training on the 3D data to generate a 3D hand pose estimation model. Therefore, the 3D data has a higher number of dimensions than 2D data, the computational time is higher, as shown in the Tab. 4.

The third issue *(iii)* is discussed in this paper is the results and challenges of egocentric vision datasets. Most of the proposed methods for estimating 3D hand pose are quantitatively evaluated on MSRA, NYU, ICVL, etc. In these databases, the hands are often the full joints thus the results are high accuracy. Figure 13 shows the results of 3D hand pose estimation on the egocentric vision (EgoDexter) dataset [10] and based on reading paper, the average 3D error is 32.6mm.

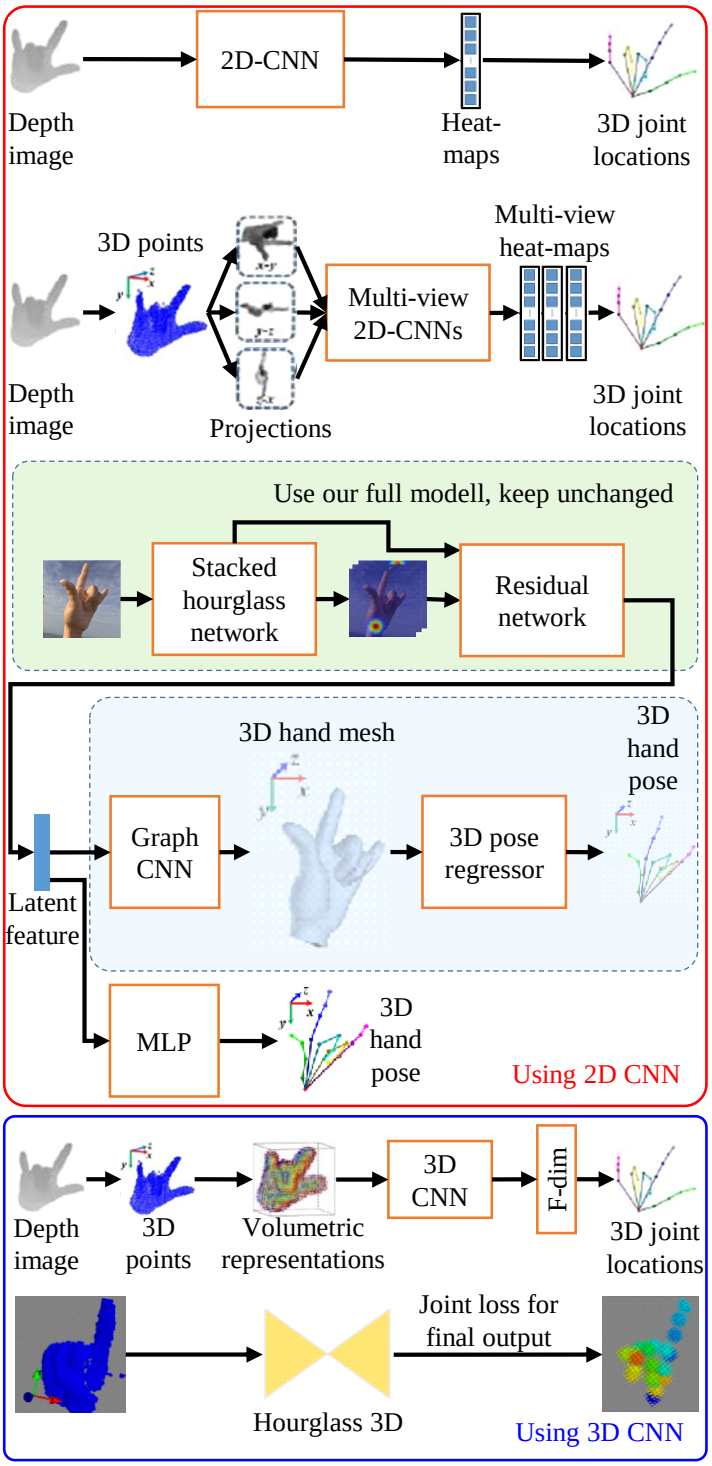


Figure 12: Illustration of two CNNs types: 2D CNN, 3D CNN.

This error is very high compared to other datasets (ICVL is 6.28 - 10.4mm, NYU is 8.42 - 20.7mm, MSRA is 7.49 - 13.1mm). Figure 14 is also shown a comparison of 3D hand pose estimation results on BigHand dataset and egocentric dataset [103]. The results on the BigHand dataset have more than 90% of 3D distance errors being less than 10mm (Fig. 14(top)). On the egocentric dataset is about 30% of 3D distance errors being less than 10mm (Fig. 14(bottom)).

Table 3: The average 3D distance error of the CNNs on the ICVL, NYU, MSRA datasets for 3D hand pose estimation [72].

Method	Mean error (mm)			Input	
	ICVL	NYU	MSRA	2D	3D
Multi-view CNNs[40]	-	-	13.1	✓	
DISCO [104]	-	20.7	-	✓	
DeepPrior [25]	10.4	19.73	-	✓	
Feedback [30]	-	15.97	-	✓	
Global2Local [55]	-	15.6	12.8	✓	
CrossingNets [45]	10.2	15.5	12.2	✓	
HBE [105]	8.62	-	-	✓	
REN (4x6x6) [106]	7.63	13.39	-	✓	
REN (9x6x6) [107]	7.31	12.69	9.79	✓	
DeepPrior++ [51]	8.1	12.24	9.5	✓	
Pose-REN [97]	6.79	11.81	8.65	✓	
Generalized [33]	-	10.89	-	✓	
CrossInfoNet [83]	6.73	10.07	7.86	✓	
HCRNN [72]	6.58	9.41	7.77	✓	
3D CNN [46]	-	14.1	9.58		✓
SHPR-Net [64]	7.22	10.78	7.96		✓
3D DenseNet [58]	6.7	10.6	7.9		✓
Hand PointNet [108]	6.94	10.5	8.5		✓
Point-to-Point [69]	6.33	9.04	7.71		✓
V2V-PoseNet [62]	6.28	8.42	7.49		✓

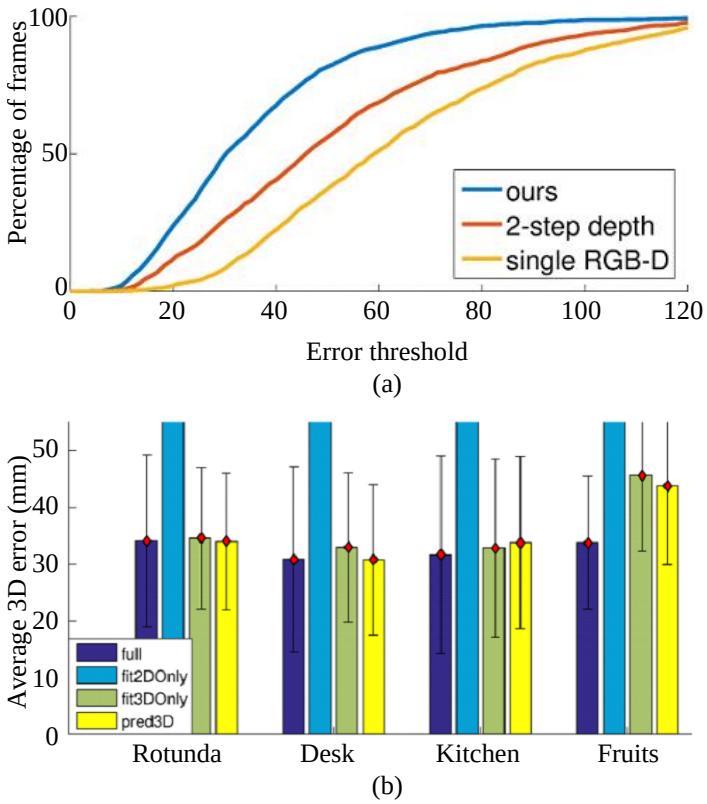


Figure 13: The distribution of 3D distance errors on the EgoDexter dataset [10]; (a) 3D distance errors of 3D hand pose estimation by the CNN that combine of HALNet and JORNet CNNs on the EgoDexter dataset; (b) the average 3D error of 3D hand pose estimation on the EgoDexter dataset.

Table 4: The test speed of the CNNs with a single GPU [72].

Method	Test speed (fps)	Input	
		2D	3D
V2V-PoseNet [62]	3.5		✓
Point-to-Point [69]	41.8		✓
HandPointNet [108]	48		✓
3D DenseNet [58]	126		✓
3D CNN [46]	215		✓
DeepPrior++ [51]	30	✓	
Generalized [33]	40	✓	
CrossingNets [45]	90.9	✓	
CrossInfoNet [83]	124.5	✓	
Feedback [30]	400	✓	
HCRNN [72]	240	✓	

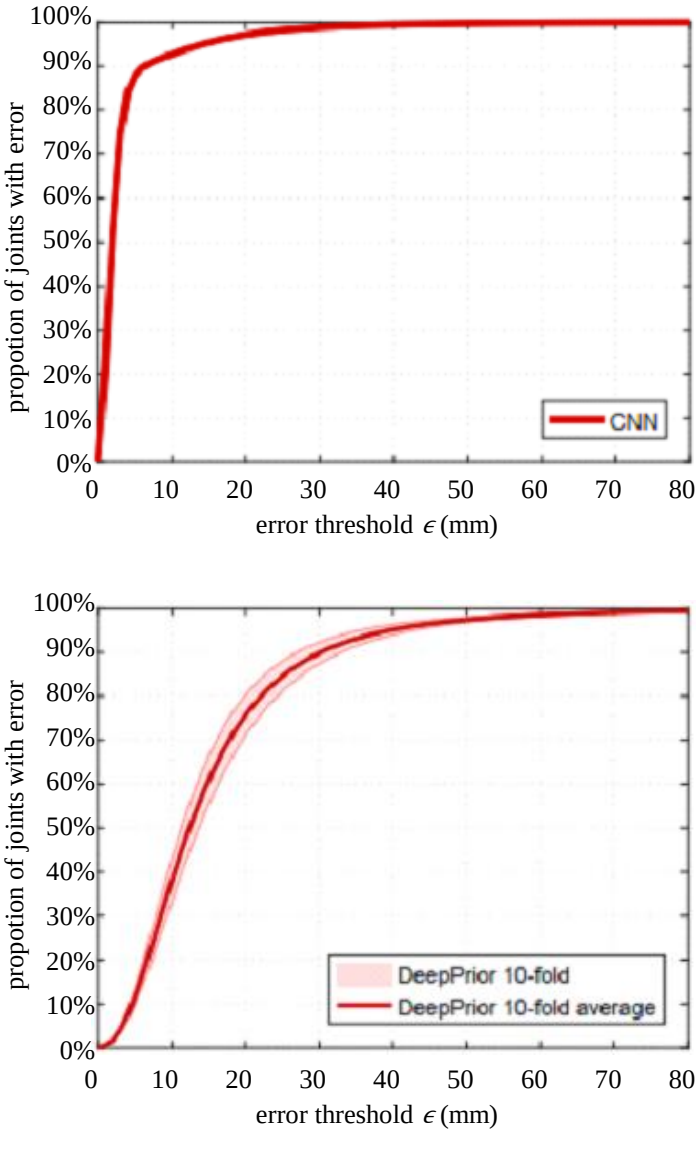


Figure 14: The distribution of 3D distance errors by baseline CNN on BigHand dataset (a) and Egocentric dataset (b) [103].

Based on the survey of 3D hand pose estimation on the egocen-

tric datasets, the estimated distance error is high because the hand is obscured by many objects or the view of the camera just looks at the palm hand, as illustrated in Fig. 15.

In reality, the real-time hand pose estimation from moving, the camera viewpoints in cluttered real-world scenes of the hand is often occluded as it naturally interacts with objects, remains an unsolved problem. In real activities as game playing, human interaction, the image data of scenes are collected from cameras mounted on the head (for VR/AR applications), shoulder, or chest. Occlusions, cluttered backgrounds, manipulated objects, and field-of-view limitations make this scenario particularly challenging. Therefore, the problem of estimating 3D hand pose on the egocentric datasets must be studied in the future.

Figure 15: The illustration data of hands are lost or obscured [109].

5 Discussion

To verify the proposed CNN, researchers often use some standard datasets. The greater number of those datasets lets many challenges when building CNN. The properties of datasets, i.e. the training set, testing set, validation set, and evaluation matrix, and those challenges are shown as followings:

5.1 Benchmark Datasets

5.1.1 Obtained Datasets from A Fixed Number of Perspectives

NYU dataset [26] includes 72757 and 8252 images of training and testing set, respectively. Each frame consists of a pair of RGB images and depth images from three MS Kinect v1, i.e. a frontal view and two side views. Those images are annotated by the ground-truth hand-pose. The authors used the Randomized Decision Forest (RDF) to train a binary classification model by this dataset. And then this classification segments each pixel that belongs to a hand or background in the depth image. 3D ground truth includes 42 DOF of 25 joints.

76k depth images of 9 subjects' right hands are captured using Intel's Creative Interactive Gesture Camera in **MSRA dataset** [110]. Each subject has 17 gestures captured. There are about 500 frames and 21 3D ground truth hand joints per frame, including wrist, index mcp, index pip, index dip, index tip, middle mcp, middle pip, middle dip, middle tip, ring mcp, ring pip, ring dip, ring tip, little mcp, little pip, a little dip, little tip, thumb mcp, thumb pip, thumb dip, and thumb tip. The resolution of the image is 320×240 pixels. The camera's intrinsic parameters are also provided, i.e. principal point of the image is (160, 120) and the focal length is 241.42.

ICVL dataset [111] includes 22K training frames and 1.6K testing frames that captured by the Intel's Creative Interactive Gesture Camera. It also provides 3D ground truth with 16 hand joints, including palm, thumb root, thumb mid, thumb tip, index root, index mid, index tip, middle root, middle mid, middle tip, ring root, ring mid, ring tip, pinky root, pinky mid, and pinky tip.

Stereo Hand Pose Tracking Benchmark (STB) dataset [39] includes 18,000 stereo and depth images with the 3D ground-truth of 21 hand joints. Those truths are palm center(not wrist or hand center), little mcp, little pip, a little dip, little tip, ring mcp, ring pip, ring dip, ring tip, middle mcp, middle pip, middle dip, middle tip, index mcp, index pip, index dip, index tip, thumb mcp, thumb pip, thumb dip, and thumb tip. The stereo is captured by a Point Grey Bumblebee2 stereo camera and the depth image is captured from an Intel Real Sense F200 active depth camera. This dataset also provides the camera parameters.

Rendered Hand Pose Dataset (RHD) [50] provides 41258 training images and 2728 testing images whose resolution is 320×320 pixels. The images include the RGB and depth images. This dataset also provides the 3D ground truth with 21 joint points. 214971 annotated depth images of the hands of the **HandNet dataset** [34] are divided into three groups. The training set includes 202198 images. The testing set contains 10000 images. The validation set has 2773 images. The used sensor is RealSense RGBD. The hand pose annotation is per pixel classes, 6D fingertip pose, and heatmap. There are 102,000 depth images of a subject in the **MSRC dataset** [112]. 100k of them belong to the training set. The resolution is 512×424 pixels and the number of viewpoints is 3. This dataset also provides the annotation data with 22 joint points.

5.1.2 Obtained Datasets from Egocentric Vision

Being captured from the Intel Creative camera mounted on the chest of humans from the right hand and left hand, the **UCI-EGO dataset** [11] provides 400 frames. 3D annotations of keypoints with 26 joint points are also provided. To annotate this dataset for evaluating 3D hand pose estimation and hand tracking the authors developed a semi-automatic labeling tool which allows to accurately annotate partially occluded hands and fingers in the 3D space by using the techniques: A few 2D joints are first manually labeled in the image and used to select the closest synthetic exemplars in the training set; A full hand pose is then created combining the manual labeling and the selected 3D exemplar; This pose is manually refined, leading to the selection of a new exemplar, and the creation of a new pose; This iterative process is followed until acceptable labeling is achieved.

Graz16 dataset [113] has more than 2000 depth frames of several egocentric sequences of six subjects. 3D annotations are made with 21 joint points. The size of the image is 320×240 pixels. The authors proposed a semi-automated the application that makes it easy to annotate sequences of articulated poses in the 3D space. This application asks a human annotator to provide an estimate of the 2D re-projections of the visible joints in frames they are called

reference frames. It proposes a method to automatically select these reference frames to minimize the annotation effort, based on the appearances of the frames over the whole sequence. It then uses this information to automatically infer the 3D locations of the joints for all the frames, by exploiting appearance, temporal, and distances constraints.

Ego Dexter dataset is an RGB-D dataset for evaluating hand tracking and 3D hand pose estimation in the cases of occlusions and clutter. It is captured from Intel RealSense SR300. It consists of 4 sequences with 4 actors (2 female), and varying interactions with various objects and cluttered background. Fingertip positions were manually annotated 1485 frames in 3190 frames.

Dexter+Object dataset [114] provide 3014 frames with ground truth annotations. The frames are collected in pairs: RGB frame is captured from the Creative Senz3D color camera; The depth frame is captured from Creative Senz3D close range TOF depth camera. It consists of 6 sequences of a hand manipulating a cuboid (2 different sizes) in different hand-object configurations and grasps. The annotation of hand joints manually annotated pixels on the depth image to mark 5 fingertip positions, and 3 cuboid corners. It is illustrated in Fig. 16.

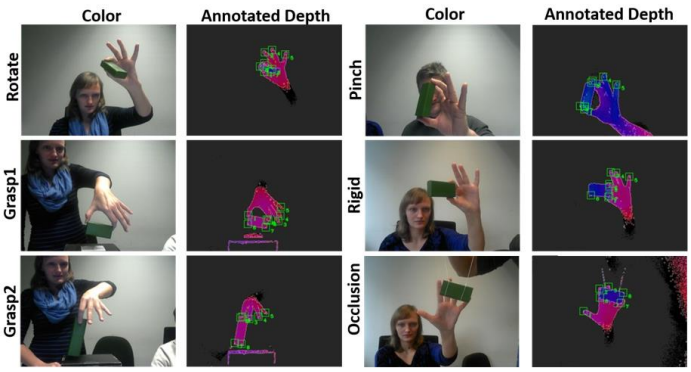


Figure 16: The illustration annotation of the Ego Dexter dataset [114].

Especially, the **BigHand2.2M dataset** [103] provides 2.2 million depth maps of ten subjects (7 males, 3 females) with accurately annotated joint locations. To determine the 3D annotations, the authors use two hardware synchronized electromagnetic tracking units, including six 6D magnetic sensors and one mid-range transmitter. The captured device is the Intel RealSense SR300 camera whose maximum speed is 60fps. The resolution is 640×480 pixels and the number of degrees of freedom (DOF) is 31. This dataset is divided into three parts, including 1.534 million images of the prior predefined pose, 375K images of random poses, and 290K images of egocentric poses.

Hampali et al. [77] introduce a benchmark dataset with 80,000 frames of 10 different users. They manipulate one among 10 different objects from the YCB dataset. The size of both depth and RGB image is 640×480 pixels. This dataset is synchronized from five cameras. The authors also proposed a method to automatically annotate each frame with accurate estimates of the poses, despite large mutual occlusions.

From the reality of the egocentric datasets, the data of hands are suffering from occlusions, cluttered backgrounds, manipulated objects, and field-of-view limitations. Unlike human pose estimation, the size of a person is large so it is easier to get a standard benchmark with a hand, thus there exist no standard benchmarks for hand pose estimation, especially in egocentric datasets. As illustrated in Fig. 17, the data of the fingers is obscured, the annotated joints of these fingers are difficult. Although there are already some semi-automatic annotation methods like in [11], [113]. However, all methods have errors as shown in Table 3 [113]. Therefore, to annotate the joints for evaluating 3D hand pose estimates on the egocentric vision datasets requires further research. This is also the fourth discussion *(iv)* in this paper.

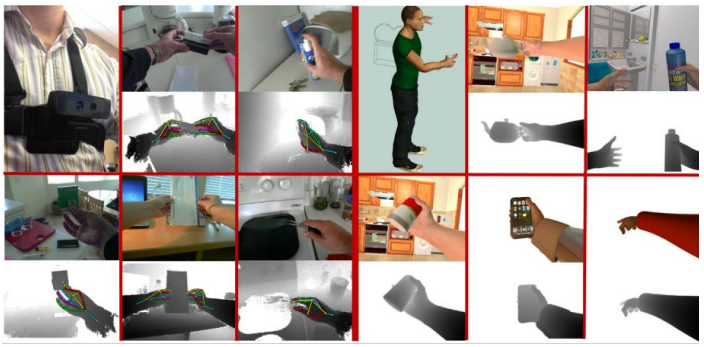


Figure 17: The illustration of fingers is occluded of UCI-EGO dataset [11].

5.1.3 *Evaluation Measurements*

There are three measurements to evaluate 3D hand pose estimation as follows:

- The first is **3D pose error**, which is the average error in Euclidean space between the estimated 3D joints and the ground truth joints.
- The second is **3D PCK**, as the percentage of correct key points of which the Euclidean error distance is below a threshold.
- The last is **AUC**, which is the area under the curve on PCK for different error thresholds.

5.2 *Challenges*

The 2017 Hands in the Million Challenge [115] is built on the BigHand2.2M [103] and First-Person Hand Action [109] datasets. This challenge had two tasks, i.e. 3D hand pose tracking and 3D hand pose estimation when hand interacts with different objects (e.g, juice bottle, salt bottle, knife, milk bottle, soda can, etc.). Based on this challenge, the proposed method of [116] is accepted in the IEEE Conference on Computer Vision and Pattern Recognition 2018. The tasks of the HANDS 2019 challenge [117] are Depth-Based 3D Hand Pose Estimation in the BigHand2.2M [103], Depth-Based 3D Hand Pose Estimation while Interacting with Objects in the F-PHAB [109] and RGB-Based 3D Hand Pose Estimation while Interacting with Objects in the HO-3D [77].

6 Conclusions

3D hand pose estimation problem is applied in many applications of computer vision and robotics: human-computer interaction; gesture recognition, interactive games, Computer-Aided Design (CAD), sign languages, action recognition, etc. The studies of 3D hand pose estimation for recognizing the grasping attributes of the objects, thereby promoting the development of robotic arms grasping objects. Before building these applications, a 3D hand pose should be fully estimated. When grasping objects, the data of the hand will be lost, missing, obscured, and are collected from cameras mounted on the head (for VR/AR applications), shoulder, or chest, thus, the process of estimating the 3D hand pose is a challenge. In this paper, we survey by the CNN methods, datasets, results of 3D hand pose estimation according to the type input data. Studies have shown that to estimate the 3D hand pose, it is necessary to use 3D hand pose libraries or 3D ground truth data to regress 3D hand pose. We also analyzed the challenges and current results of CNNs for 3D hand pose estimation on the normal benchmark datasets and egocentric datasets. In particular, we discussed internally on four issues in estimating 3D hand pose: The number of valuable studies about 3D hand pose estimation; The estimated results of 3D hand pose when using 3D CNNs and 2D CNNs; The challenges of the datasets which are collected from egocentric vision sensors; The methods to collect and annotate datasets from egocentric vision sensors. In the future, we plan to build a benchmark dataset to evaluate 3D hand pose estimation. This dataset will use the egocentric camera to collect hand data while grasping the objects. We also plan to propose a method for 3D location joints in manifold space that uses the Lie group, then extract the characteristics for training to generate an estimation model by CNNs to predict 3D location joints.

Acknowledgment: This research was funded by the elementary level topic of Hung Vuong University, Vietnam. The title is "Using the Lie algebra, Lie group to improve the skeleton hand presentation".

References

[1] P. Krejov, A. Gilbert, R. Bowden, "Guided optimisation through classification and regression for hand pose estimation," Computer Vision and Image Understanding, **155**, 124–138, 2016.

[2] P. Krejov, R. Bowden, "Multi-touchless: Real-time fingertip detection and tracking using geodesic maxima," in 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, FG 2013, 2013.

[3] Y. Zhou, G. Jiang, Y. Lin, "A novel finger and hand pose estimation technique for real-time hand gesture recognition," Pattern Recognition, **49**, 102–114, 2016, doi:10.1016/j.patcog.2015.07.014.

[4] A. Spurr, "Gesture Recognition : Hand Pose Estimation," ETH Zurich, 2014.

[5] A. T. Chan, H. V. Leong, S. H. Kong, "Real-time tracking of hand gestures for interactive game design," in IEEE International Symposium on Industrial Electronics, 98–103, 2009.

[6] R. Y. Wang, S. Paris, J. Popovic, "6D hands: Markerless hand tracking for computer aided design," in UIST'11 - Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 549–557, 2011.

[7] J. Isaacs, S. Foo, "Optimized wavelet hand pose estimation for American sign language recognition," in Proceedings of the 2004 Congress on Evolutionary Computation, CEC2004, volume 1, 797–802, 2004.

[8] YOLOv2, "Hand detection, Gesture recognition by YOLOv2," https://www.youtube.com/watch?v=JJPsHpVt_lA, 2020, [Accessed 31 May 2020].

[9] T. Malisiewicz, "Hand Segmentation for Gesture Recognition in EGO-Vision," https://twitter.com/quantombone/status/425560005434015744, 2020, [Accessed 31 May 2020].

[10] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, C. Theobalt, "Real-Time Hand Tracking under Occlusion from an Egocentric RGB-D Sensor," in Proceedings of the IEEE International Conference on Computer Vision, volume 2017-Octob, 1163–1172, 2017.

[11] G. Rogez, M. Khademi, J. S. Supanvcivc, J. M. Montiel, D. Ramanan, "3D hand pose detection in egocentric RGB-D images," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 8925, 356–371, 2015.

[12] Microsoft, "Microsoft Xbox 360 Kinect Launches November 4," https://gizmodo.com/microsoft-xbox-360-kinect-launches-november-4-5563148, 2010, [Online; accessed 7-February-2020].

[13] J. Kramer, N. Burrus, F. Echtler, H. C. Daniel, M. Parker, "Hacking the Kinect," Apress, 2012.

[14] Microsoft, "Kinect v2 comes to the PC on July 15," https://www.pcgamer.com/kinect-for-windows-v2-to-release-july-15/, 2014, [Online; accessed 7-February-2020].

[15] W. G. Wong, "How Microsoft's PrimeSense-based Kinect Really Works," https://www.electronicdesign.com/technologies/embedded-revolution/article/21795925/how-microsofts-primesensebased-kinect-really-works, 2011, [Online; accessed 7-February-2020].

[16] Intel, "Intel RealSens Technology," https://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html, 2020, [Online; accessed 7-February-2020].

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

[18] M. Cai, K. M. Kitani, Y. Sato, "Understanding hand-object manipulation with grasp types and object attributes," Robotics: Science and Systems, **12**, 2016.

[19] M. Teremetz, F. Colle, S. Hamdoun, M. A. Maier, P. G. Lindberg, "A novel method for the quantification of key components of manual dexterity after stroke," Journal of NeuroEngineering and Rehabilitation, **12**(1), 2015.

[20] W. Yan, H. Nie, J. Chen, D. Han, "Optimal design and grasp ability evaluation of four-finger tendon-driven hand," International Journal of Advanced Robotic Systems, **14**(6), 1–14, 2017, doi:10.1177/1729881417748444.

[21] Y. Li, Z. Xue, Y. Wang, L. Ge, Z. Ren, J. Rodriguez, "End-to-End 3D Hand Pose Estimation from Stereo Cameras," in BMVC, 1–13, 2019.

[22] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, X. Twombly, "Vision-based hand pose estimation: A review," Computer Vision and Image Understanding, **108**(1-2), 52–73, 2007.

[23] E. Barsoum, "Articulated Hand Pose Estimation Review," https://arxiv.org/pdf/1604.06195, 1–50, 2016.

[24] J. Tompson, M. Stein, Y. Lecun, K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," ACM Transactions on Graphics, **33**(5), 2014.

[25] M. Oberweger, P. Wohlhart, V. Lepetit, "Hands Deep in Deep Learning for Hand Pose Estimation," in Computer Vision Winter Workshop, 2015.

[26] J. Tompson, M. Stein, Y. Lecun, K. Perlin.

[27] C. Choi, A. Sinha, J. H. Choi, S. Jang, K. Ramani, "A collaborative filtering approach to real-time hand pose estimation," in Proceedings of the IEEE International Conference on Computer Vision, volume 2015 Inter, 2336–2344, 2015.

[28] G. Poier, K. Roditakis, S. Schulter, D. Michel, H. Bischof, A. A. Argyros, "Hybrid One-Shot 3D Hand Pose Estimation by Exploiting Uncertainties," in BMVC, 182.1–182.14, 2015.

[29] P. Li, H. Ling, X. Li, C. Liao, "3D hand pose estimation using randomized decision forest with segmentation index points," in Proceedings of the IEEE International Conference on Computer Vision, volume 2015 Inter, 819–827, 2015.

[30] M. Oberweger, P. Wohlhart, V. Lepetit, "Training a feedback loop for hand pose estimation," in Proceedings of the IEEE International Conference on Computer Vision, volume 2015 Inter, 3316–3324, 2015.

[31] X. Sun, Y. Wei, S. Liang, X. Tang, J. Sun, "Cascaded hand pose regression," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 07-12-June, 824–832, 2015.

[32] D. Tang, Q. Ye, S. Yuan, J. Taylor, P. Kohli, C. Keskin, T.-k. Kim, J. Shotton, "Opening the Black Box : Hierarchical Sampling Optimization for Hand Pose Estimation," in iccv, volume 8828, 1–14, 2015.

[33] M. Oberweger, P. Wohlhart, V. Lepetit, "Generalized Feedback Loop for Joint Hand-Object Pose Estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, **14**(8), 1–1, 2015.

[34] C. Wan, A. Yao, L. Van Gool, "Hand pose estimation from local surface normals," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 9907 LNCS, 554–569, 2016.

[35] Q. Ye, S. Yuan, T.-k. Kim, "Spatial Attention Deep Net with Partial PSO for Hierarchical Hybrid Hand Pose Estimation," https://arxiv.org/abs/1604.03334, 1–16, 2016.

[36] A. Sinha, W. Lafayette, "DeepHand : Robust Hand Pose Estimation by Completing a Matrix Imputed with Deep Features," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4150–4158, 2016.

[37] M. Oberweger, G. Riegler, P. Wohlhart, V. Lepetit, "Efficiently creating 3D training data for fine hand pose estimation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2016-Decem, 4957–4965, 2016.

[38] C. Xu, L. N. Govindarajan, Y. Zhang, L. Cheng, "Lie-X: Depth Image Based Articulated Object Pose Estimation, Tracking, and Action Recognition on Lie Groups," International Journal of Computer Vision, **123**(3), 454–478, 2016.

[39] Y. Zhang, C. Xu, L. Cheng, "Learning to Search on Manifolds for 3D Pose Estimation of Articulated Objects," https://arxiv.org/abs/1612.00596, 2016.

[40] L. Ge, H. Liang, J. Yuan, D. Thalmann, "Robust 3D Hand Pose Estimation from Single Depth Images Using Multi-View CNNs," IEEE Transactions on Image Processing, **27**(9), 4422–4436, 2016.

[41] X. Deng, S. Yang, Y. Zhang, P. Tan, L. Chang, H. Wang, "Hand3D: Hand Pose Estimation using 3D Neural Network," https://arxiv.org/abs/1502.06807, 2017.

[42] S. Yuan, Q. Ye, B. Stenger, S. Jain, T. K. Kim, "BigHand2.2M benchmark: Hand pose dataset and state of the art analysis," in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, volume 2017-Janua, 2605–2613, 2017.

[43] C. Choi, S. Kim, K. Ramani, "Learning Hand Articulations by Hallucinating Heat Distribution," in Proceedings of the IEEE International Conference on Computer Vision, volume 2017-Octob, 3123–3132, 2017.

[44] C. Choi, S. H. Yoon, C.-N. Chen, K. Ramani, "Robust Hand Pose Estimation during the Interaction with an Unknown Object," in IEEE International Conference on Computer Vision (ICCV), i, 3142–3151, 2017.

[45] C. Wan, T. Probst, L. V. Gool, A. Yao, "Crossing Nets : Combining GANs and VAEs with a Shared Latent Space for Hand Pose Estimation," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[46] L. Ge, H. Liang, J. Yuan, D. Thalmann, "3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[47] N. Neverova, C. Wolf, F. Nebout, G. W. Taylor, "Hand pose estimation through semi-supervised and weakly-supervised learning," Computer Vision and Image Understanding, **164**, 56–67, 2017.

[48] X. Zhang, C. Xu, Y. Zhang, T. Zhu, L. Cheng, "Multivariate Regression with Grossly Corrupted Observations: A Robust Approach and its Applications," https://arxiv.org/abs/1701.02892, 2017.

[49] J. Malik, A. Elhayek, D. Stricker, "Simultaneous Hand Pose and Skeleton Bone-Lengths Estimation from a Single Depth Image," https://arxiv.org/abs/1712.03121, 2017.

[50] C. Zimmermann, T. Brox, "Learning to Estimate 3D Hand Pose from Single RGB Images," in IEEE International Conference on Computer Vision (ICCV), 2017, https://arxiv.org/abs/1705.01389.

[51] M. Oberweger, V. Lepetit, "DeepPrior++: Improving Fast and Accurate 3D Hand Pose Estimation," in Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017, volume 2018-Janua, 585–594, 2017.

[52] S. Baek, K. I. Kim, T. K. Kim, "Augmented Skeleton Space Transfer for Depth-Based Hand Pose Estimation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 8330–8339, 2018.

[53] Y. Wu, W. Ji, X. Li, G. Wang, J. Yin, F. Wu, "Context-Aware Deep Spatiotemporal Network for Hand Pose Estimation From Depth Images," IEEE Transactions on Cybernetics, 1–11, 2018.

[54] J. S. Supancic, G. Rogez, Y. Yang, J. Shotton, D. Ramanan, "Depth-Based Hand Pose Estimation: Methods, Data, and Challenges," International Journal of Computer Vision, **126**(11), 1180–1198, 2018.

[55] M. Madadi, S. Escalera, X. Baro, J. Gonzalez, "End-to-end Global to Local CNN Learning for Hand Pose Recovery in Depth Data," https://arxiv.org/pdf/1705.09606.pdf, 2018.

[56] M. Rad, M. Oberweger, V. Lepetit, "Feature Mapping for Learning Fast and Accurate 3D Pose Inference from Synthetic Images," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 4663–4672, 2018.

[57] G. Garcia-Hernando, S. Yuan, S. Baek, T. K. Kim, "First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 409–419, 2018.

[58] L. Ge, H. Liang, J. Yuan, S. Member, D. Thalmann, "Real-time 3D Hand Pose Estimation with 3D Convolutional Neural Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, **8828**(c), 2018.

[59] L. Chen, S.-Y. Lin, Y. Xie, H. Tang, Y. Xue, X. Xie, Y.-Y. Lin, W. Fan, "Generating Realistic Training Images Based on Tonality-Alignment Generative Adversarial Networks for Hand Pose Estimation," https://arxiv.org/abs/1811.09916, 2018.

[60] C. Zhang, G. Wang, H. Guo, X. Chen, F. Qiao, H. Yang, "Interactive hand pose estimation: Boosting accuracy in localizing extended finger joints," in International Symposium on Electronic Imaging Science and Technology, 1, 1–6, 2018.

[61] J. Wohlke, S. Li, D. Lee, "Model-based Hand Pose Estimation for Generalized Hand Shape with Appearance Normalization," https://arxiv.org/abs/1807.00898, 2018.

[62] G. Moon, J. Y. Chang, K. M. Lee, "V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR, 5079–5088, 20118.

[63] Q. Ye, T. K. Kim, "Occlusion-aware hand pose estimation using hierarchical mixture density network," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 11214 LNCS, 817–834, 2018.

[64] X. Chen, G. Wang, S. Member, C. Zhang, K. I. M. Member, X. Ji, "SHPR-Net : Deep Semantic Hand Pose Regression From Point Clouds," IEEE Access, **PP**(c), 1, 2018.

[65] A. Spurr, J. Song, S. Park, O. Hilliges, "Cross-modal Deep Variational Hand Pose Estimation," in CVPR, 89–98, 2018.

[66] F. Huang, A. Zeng, C. Science, H. Kong, H. Kong, M. Liu, J. Qin, Q. Xu, "Structure-Aware 3D Hourglass Network for Hand Pose Estimation from Single Depth Image," https://arxiv.org/abs/1812.10320, 1–12, 2018.

[67] P. Panteleris, I. Oikonomidis, A. Argyros, "Using a single RGB frame for real time 3D hand pose estimation in the wild," in 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018.

[68] C. Wan, T. Probst, L. V. Gool, A. Yao, "Dense 3D Regression for Hand Pose Estimation," in CVPR, 2018.

[69] L. Ge, Z. Ren, J. Yuan, "Point-to-point regression pointnet for 3D hand pose estimation," in European Conference on Computer Vision, volume 11217 LNCS, 489–505, 2018.

[70] Y. Zhang, L. Chen, Y. Liu, J. Yong, W. Zheng, "Adaptive Wasserstein Hourglass for Weakly Supervised Hand Pose Estimation from Monocular RGB," https://arxiv.org/abs/1909.05666, 2019.

[71] S. Sharma, S. Huang, D. Tao, "An End-to-end Framework for Unconstrained Monocular 3D Hand Pose Estimation," https://arxiv.org/abs/1911.12501, 1–12, 2019.

[72] C.-h. Yoo, S.-w. Kim, S.-w. Ji, Y.-g. Shin, S.-j. Ko, "Capturing Hand Articulations using Recurrent Neural Network for 3D Hand Pose Estimation," https://arxiv.org/abs/1911.07424, 2019.

[73] Y. Li, C. Twigg, Y. Ye, L. Tao, X. Wang, "Disentangling Pose from Appearance in Monochrome Hand Images," https://arxiv.org/abs/1904.07528, 2019.

[74] C. Wan, T. Probst, L. Van Gool, A. Yao, "Dual Grid Net: hand mesh vertex regression from single depth maps," https://arxiv.org/abs/1907.10695, 2019.

[75] J. Liu, H. Ding, A. Shahroudy, L. Y. Duan, X. Jiang, G. Wang, A. C. Kot, "Feature Boosting Network for 3D Pose Estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, **42**(2), 494–501, 2019.

[76] L. W. X. Cejnog, R. M. Cesar, T. E. De Campos, V. M. C. Elui, "Hand range of motion evaluation for Rheumatoid Arthritis patients," in Proceedings - 14th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2019, 2, 2019.

[77] S. Hampali, M. Rad, M. Oberweger, V. Lepetit, "HOnnotate: A method for 3D Annotation of Hand and Objects Poses," https://arxiv.org/abs/1907.01481, 2019.

[78] S. Li, D. Lee, "Point-to-Pose Voting based Hand Pose Estimation using Residual Permutation Equivariant Layer," in CVPR, 2019.

[79] X. Zhang, F. Zhang, "Pixel-wise Regression : 3D Hand Pose Estimation via Spatial-form Representation and Differentiable Decoder," **XX**(Xx), 1–10, 2019.

[80] S. Baek, T.-k. Kim, "Pushing the Envelope for RGB-based Dense 3D Hand Pose Estimation via Neural Rendering," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[81] K.-w. Lee, S.-h. Liu, H.-t. Chen, K. Ito, "Silhouette-Net : 3D Hand Pose Estimation from Silhouettes," https://arxiv.org/abs/1912.12436, 2019.

[82] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, J. Yuan, "3D Hand Shape and Pose Estimation from a Single RGB Image," in CVPR, 2019.

[83] K. Du, X. Lin, Y. Sun, X. Ma, "Crossinfonet: Multi-task information sharing based hand pose estimation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2019-June, 9888–9897, 2019.

[84] X. Sun, Y. Wei, S. Liang, X. Tang, J. Sun, "Cascaded hand pose regression," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 07-12-June, 824–832, 2015.

[85] C. Qi, L. Yi, H. Su, L. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," in NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, 5105–5114, 2017.

[86] D. Tang, H. J. Chang, A. Tejani, T. K. Kim, "Latent regression forest: Structured estimation of 3D hand poses," IEEE Transactions on Pattern Analysis and Machine Intelligence, **39**(7), 1374–1387, 2017.

[87] J. Supancic, D. Ramanan, J. S. Supan, "Understanding Everyday Hands in Action from RGB-D Images," in 2015 IEEE International Conference on Computer Vision (ICCV), 3889–3897, 2015.

[88] X. Chen, C. Shi, B. Liu, "Static hand gesture recognition based on finger root-center-angle and length weighted Mahalanobis distance," Real-Time Image and Video Processing 2016, **9897**(61271390), 98970U, 2016.

[89] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[90] S.-E. Wei, V. Ramakrishna, T. Kanade, Y. Sheikh, "Convolutional pose machines," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016) url = http://dx.doi.org/10.1109/CVPR.2016.511, year = 2016,.

[91] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," https://arxiv.org/abs/1603.04467, 2016.

[92] D. P. Kingma, J. L. Ba, "Adam: A method for stochastic optimization," in 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, 1–15, 2015.

[93] openpose, "openpose," `https://github.com/CMU-Perceptual-Computing-Lab/openpose`, 2019, [Accessed 23 April 2019].

[94] J. Redmon, A. Farhadi, "YOLO9000: Better, faster, stronger," in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, volume 2017-Janua, 6517–6525, 2017.

[95] Y. He, W. Hu, S. Yang, X. Qu, P. Wan, Z. Guo, "GraphPoseGAN: 3D Hand Pose Estimation from a Monocular RGB Image via Adversarial Learning on Graphs," https://arxiv.org/abs/1912.01875, 2019.

[96] U. Iqbal, P. Molchanov, T. Breuel, J. Gall, J. Kautz, "Hand Pose Estimation via Latent 2.5D Heatmap Regression," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 11215 LNCS, 125–143, 2018.

[97] X. Chen, G. Wang, H. Guo, C. Zhang, "Pose guided structured region ensemble network for cascaded hand pose estimation," Neurocomputing, 2019.

[98] S. Li, D. Lee, "Point-to-pose voting based hand pose estimation using residual permutation equivariant layer," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2019-June, 11919–11928, 2019.

[99] "Pointwise Convolutional Neural Networks," in IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[100] R. Klokov, V. Lempitsky, "Escape from Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models," in Proceedings of the IEEE International Conference on Computer Vision, volume 2017-Octob, 863–872, 2017.

[101] J. Li, B. M. Chen, "So-net: Self-organizing network for point cloud analysis," in Proceedings of the IEEE International Conference on Computer Vision, 9397–9406, 2018.

[102] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, J. M. Solomon, "Dynamic graph Cnn for learning on point clouds," ACM Transactions on Graphics, **38**(5), 2019.

[103] S. Yuan, Q. Ye, u. . h. y. Björn Stenger and Siddhant Jain and Tae-Kyun Kim, booktitle=CVPR, "BigHand2.2M Benchmark: Hand Pose Dataset and State of the Art Analysis," .

[104] D. Bouchacourt, P. K. Mudigonda, S. Nowozin, "DISCO Nets : DISsimilarity COefficients Networks," in D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, R. Garnett, editors, Advances in Neural Information Processing Systems 29, 352–360, Curran Associates, Inc., 2016.

[105] Y. Zhou, J. Lu, K. Du, X. Lin, Y. Sun, X. Ma, "HBE: Hand branch ensemble network for real-time 3d hand pose estimation," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), **11218 LNCS**, 521–536, 2018.

[106] H. Guo, G. Wang, X. Chen, C. Zhang, F. Qiao, H. Yang, "Region ensemble network: Improving convolutional network for hand pose estimation," in Proceedings - International Conference on Image Processing, ICIP, volume 2017-Septe, 4512–4516, 2018.

[107] H. Guo, G. Wang, X. Chen, C. Zhang, "Towards Good Practices for Deep 3D Hand Pose Estimation," Journal of Visual Communication and Image Representation, **55**, 404–414, 2017.

[108] L. Ge, Y. Cai, J. Weng, J. Yuan, "Hand PointNet : 3D Hand Pose Estimation using Point Sets," Cvpr, 3–5, 2018.

[109] G. Garcia-Hernando, S. Yuan, S. Baek, T.-K. Kim, "First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations," in Proceedings of Computer Vision and Pattern Recognition (CVPR), 2018.

[110] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, "Multi-view Convolutional Neural Networks for 3D Shape Recognition," in Proc. ICCV, 264–272, 2015.

[111] D. Tang, H. J. Chang, A. Tejani, T. K. Kim, "Latent regression forest: Structured estimation of 3D hand poses," IEEE Transactions on Pattern Analysis and Machine Intelligence, **39**(7), 1374–1387, 2017.

[112] "Accurate, robust, and flexible realtime hand tracking," in Conference on Human Factors in Computing Systems - Proceedings, volume 2015-April, 3633–3642, 2015.

[113] M. Oberweger, G. Riegler, P. Wohlhart, V. Lepetit, "Efficiently creating 3D training data for fine hand pose estimation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 2016-Decem, 4957–4965, 2016.

[114] S. Sridhar, F. Mueller, M. Zollhoefer, D. Casas, A. Oulasvirta, C. Theobalt, "Real-time Joint Tracking of a Hand Manipulating an Object from RGB-D Input," in Proceedings of European Conference on Computer Vision (ECCV), 2016.

[115] ICCV, "The 2017 Hands in the Million Challenge on 3D Hand Pose Estimation," `http://icvl.ee.ic.ac.uk/hands17/challenge/`, 2017, [Online; accessed 22-march-2020].

[116] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Yong Chang, K. Mu Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge, J. Yuan, X. Chen, G. Wang, F. Yang, K. Akiyama, Y. Wu, Q. Wan, M. Madadi, S. Escalera, S. Li, D. Lee, I. Oikonomidis, A. Argyros, T.-K. Kim, "Depth-Based 3D Hand Pose Estimation: From Current Achievements to Future Goals," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[117] ICCV, " the HANDS19 Challenge," `https://sites.google.com/view/hands2019/challenge`, 2019, [Online; accessed 22-march-2020].