Sādhanā (2022)47:247
https://doi.org/10.1007/s12046-022-02026-7

DeepHands: Dynamic hand gesture detection with depth estimation and 3D reconstruction from monocular RGB data

RAMEEZ SHAMALIK[1,2,*] and SANJAY KOLI[1,3]

[1]Department of Electronics and Telecommunications, G H Raisoni College of Engineering and Management, Pune, India
[2]Bharati Vidyapeeths college of engineering for women, Pune, India
[3]Department of Electronics and Telecommunications, Dr. D. Y. Patil School of Engineering, Pune, India
e-mail: shamalik1@gmail.com; sanjay.koli@dypic.in

MS received 12 May 2022; revised 19 July 2022; accepted 24 October 2022

Abstract. Human hand gestures are the most important tools for interacting with the real environment. Capturing hand motion is critical for a wide range of applications in Augmented Reality (AR)/Virtual Reality (VR), Human-computer Interface (HCI), and many other disciplines. This paper presents a 3 module pipeline for effective hand gesture detection in real-time at the speed of 100 frames per second (fps).Various hand gestures can be captured by simple RGB camera and then processed to first detect the palm and then find essential 3D landmarks, which helps in creating skeletal representation of hand. In order to form a 3D mesh around the skeletal hand 2D and 3D annotations of Hand gestures are merged and in the final module 3D animated hand gestures are presented using advanced neural network. 3D representation of hand gestures ensures greater understanding of depth ambiguity problem in monocular pose estimations and can be effectively used in computer vision and graphics applications. The proposed design is compared with several benchmarks to highlight improvements in the results achieved over conventional methods.

Keywords. Augmented reality; Human-computer interaction; 3D reconstruction; Virtual reality.

1. Introduction

Gesture detection is a complex issue as it has various aspects to look after. A lot of work has already been conducted in face detection while hand gesture detection is a relatively a difficult task. Hand gestures including fingers have more permutations and combinations of gestures to detect, which can be useful for computer graphics applications. Computer vision solutions face different challenges like self-occlusions, depth ambiguity-perceptions, noisy backgrounds etc. Also real-time detection of hand gestures requires auxiliary processing power that can process more frames per second, ideal speed being 30 fps. 3D motion capture of hand gestures becomes a challenging task due to motion parallax of depth in a monocular set up and fast movements of the hand in real-time. The recent research has extensively tried to solve such problems with the effective use of deep learning [1–3], yet there are two major issues to be resolved.

Firstly, even though the annotated hand data is severely constrained due to difficulties in gathering real human hand gestures with 3D annotations, the approaches make use of the classification of all publicly available training datasets separately. The inclusiveness of all kinds of data types in solving this problem is missing. Especially for acquiring 3D annotation for hand gestures, complex set ups such as stereo cameras or multiple cameras need to be set up at different locations [4]. Also, there is another way to capture hand motions, which includes use of 3D scanners [5, 6] or hand gloves with sensors [7] placed at required key points which are completely ignored due to hardware constraints. Secondly, the state–of-the-art gesture detection research comes with 3D joint detection using different deep learning techniques, but it misses outs an opportunity of a complete 3D hand representation which can be an ultimate solution for various computer graphics applications such as AR and VR. Some investigations mitigate this issue by undertaking an independent inquiry of fitting a dynamic hand model to sparse predictions [8], but lack local convergence due to excessive optimization. All of the research studies discussed lack strong supervision in training as it includes only similar kind of 2D or 3D annotated data.

In this research paper a technique to solve the above-mentioned issues is proposed. This technique uses monocular RGB images as input to efficiently detect hand gesture landmarks in 3D space with the help of all the possible data for training along with 3D representation of captured hand in real-time. Here, initially the palm in the frame is detected using real and synthetic hand dataset.

*For correspondence
Published online: 19 November 2022

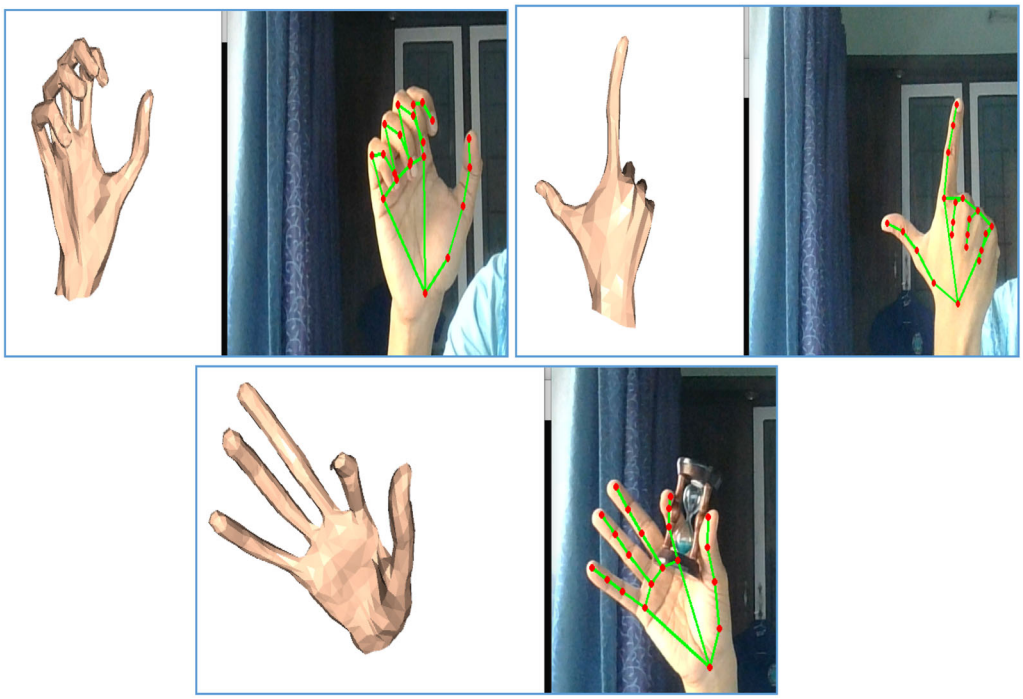


Figure 1. 3D skeletal as well as 3D virtual hand representation as final output.

Detecting palm is comparatively easy and hence, it is a fast process as the possibility of occlusion and blur is minimal.

Secondly, a hand landmark model trained on 2D and 3D annotated data is used to find out 21 landmarks in a given hand. The start of the wrist is highlighted as the ground truth. Three different datasets are used to increase accuracy, namely real-world, synthetic and combination of both. Both datasets have their pros and cons, but combined one gives scalable results. An accurate skeletal representation of hand gesture is revived to process it further.

When it comes to research in gesture detection, only skeletal representation of hand is not enough as the proposed system attempts to achieve it not only in real time but also continuously via live video feed that too at the staggering speed of 100 fps. A 3D Mesh representation of the hand gesture landmarks is selected. Therefore, because of the real-time video processing, it not only detects hand gestures but also detects motions in real-time.

For further research, it is also important to project 3D hand gestures to understand joint rotation in real time which is also known as the inverse kinematics problem [9]. For this purpose, IKNet6 is introduced, which solves the issue by using 3D hand gesture landmarks as well as quaternion representation of gestures in 3D space to animate a virtual hand. IKNet6 is trained on 2D as well 3D annotated data. It makes IKNet6 better than its previous version as training on Motion Capture data which gives strong supervision while training, thus it delivers superior performance in real-time.

A three-module pipeline is proposed for hand gesture detection in real-time:

1. Palm detector plus 2D hand landmarks
2. 3D mesh estimation of hand gesture
3. 3D mapping of hand gesture rotation

Figure 1 showcases the dynamic model, which captures and animates various hand gestures and poses in real-time. The proposed system works efficiently in various challenging scenarios such as self-occlusions, varying scales, and even object occlusions. To summarize, the proposed system delivers superior performance as compared to state-of-the-art techniques.

2. Related work

This section describes recently conducted research in this domain and how the proposed system is different and better.

2.1 *Standalone methods*

Santavas *et al* [10] proposed a lightweight Convolutional Neural Network (CNN) for 2D hand gesture detection for Human-computer Interaction (HCI). Although efficient and real–time, this system lacks the depth part for greater accuracy. A model named ArtiBoost [11] has recently been introduced for 3D hand pose detection. It has been trained only on the HO3D dataset and lacks inclusion of other possible hand datasets. BigHand 2.2 M is the benchmark dataset [12] produced exclusive outcomes for hand pose

estimation. It has been produced using six different magnetic sensors and highlights depth as well as 21 key points of hands, but it lacks joint rotation analysis. Body2Hands [13] is another technique, which has been introduced to infer 3D hands from a picture frame containing the upper body of the subject. It is a bit complicated as in every frame hand needs to be cropped from the body to process it further. ContactOpt [14] is a technique proposed to estimate the contact of human hand on the particular surface with the help of an optimized model, as it tries to find mesh in both i.e. hand and object surfaces. It turns out to be a bit complicated. Zimmermann *et al* [15] proposed a contrasting technique using self-supervised learning over a large dataset for hand shape estimation; this comes under visual representation learning and lacks a variety of possible datasets.

2.2 *Semi-supervised methods*

A semi-supervised generative model [16] is used to overcome possible annotation error in hand pose estimation by compensating the faulty ground truth. Although useful in preparation of effective datasets, this technique lacks advanced application. A cascading multitask learning method is used to understand the correlation between a hand and the object in a particular scenario [17], heat maps are used for a better understanding of the same. As multiple datasets are used to implement this model, the outputs are predictable and vary in noisy backgrounds. A multi-view bootstrapping technique is used to triangulate a frame where hand key points are found from RGB images [18], this technique lacks real-time outputs. HandTailor [19] presented a technique to recover 3D hands from an input RGB image, but misses out on 3D mesh representation of the same. Ge *et al* [20] used Graph CNN for 3D Hand gesture estimation effectively, but they did not have MoCap data in training. Chen *et al* [21] made an attempt for effective 3D reconstruction of hand, but showed inadequate results when it came to uniform skin texture of hands. A recent upgrade in 3D reconstruction of hands was carried out especially while interacting [22] using collision aware factorized refinements, although impressive this method is prone to occlusions. Another semi-supervised model with pseudo labels was attempted to highlight the interaction between 3D hands and objects [23], this model had similar constraints as the previous one. On the similar grounds, research by NVIDIA proposed an adversarial motion modelling for hand gesture estimation using unlabelled images [24]. This method needs generalization in different types of videos in real-time.

2.3 *Disparity-based methods*

Due to the widespread availability of advanced depth cameras, many studies have explored estimating hand posture from depth images, which basically highlight disparity for better understanding. Initial depth-based studies approximated hand pose by integrating a probabilistic model onto a depth image [25–27]. In other cases, exclusionary projections [28–30] were also used for initialization and validation. Self-supervised parameter tuning was adopted using unlabelled depth information [31], whereas a realistic dataset was presented to improve robustness [32]. Additional representations, such as 3D point cloud [33, 34] and 3D spatial information [2, 35] can be extracted from depth maps and were used in some investigations. While these initiatives yield compelling outcomes, they remain restrained by the intrinsic limitations of depth sensors, which do not operate in direct sunlight, consume a lot of power and demand users to be in close proximity to the sensor.

3. Methodology

As shown in figure 2, the proposed system initiates by capturing the hand gestures in real-time, then it extracts features. Firstly, it detects palm then creates 2D and later 3D key points around the palm and fingers. In the second module, 3D mesh is formed around the skeletal representation and 3D shape estimation, followed by 3D representation of hand gestures in the third one. All of this is done in

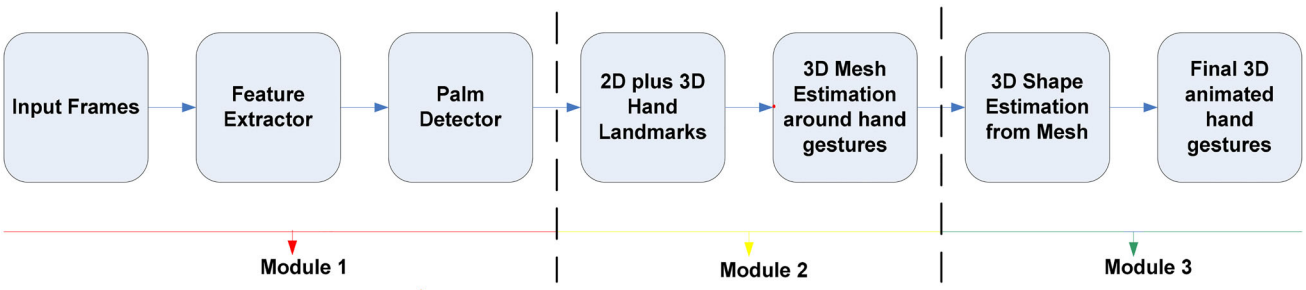


Figure 2. Step-wise implementation of the proposed system.

real-time. The total size of this three-module model is 535 MB with dataset included; the individual size of the modules does not exceed 50 MB. The detailed description of the process is presented in the following sub-sections.

3.1 *Palm detection and hand landmark module*

The first stage contains two sub-sections. First step is to detect the palm in the given frame. Once the palm is detected, then finding proper key points over the rest of the hand becomes easier. Supplying the hand landmark model with a correctly cropped palm image minimizes the necessity of data augmentation and allows the network to devote the majority of its capacity to landmark detection performance. The landmark prediction of the previous frame is used as input for the current frame to construct a bounding box, excluding to apply detector on each frame. Rather, the detector is used just on the first frame. In another scenario, the detector turns on only when there is no hand in the last frame. Thus, a lot of processing power is saved, which is crucial for a real-time system.

Hand detection is a difficult task due to two main reasons. Firstly, it has typical occlusion with its surrounding along with other fingers. Also, the pixel area in a frame covered by hand is pretty small. Secondly, as compared to face detection where there are diversified areas such as mouth, eyes, and nose, scarcity of such diversified areas makes detection of hand gesture a bit difficult task. This complexity is resolved by introducing a palm detector as palm is immune to aforementioned occlusions. A pre-trained Single Shot multibox Detector (SSD) is employed trained on COCO dataset [36], which uses Square bounding boxes for palm at the same time ignores the pixel ratio and reduces the anchors [37]. Further, Non Maximum Suppression (NMS) algorithm is applied for finding out an accurate bounding box. The NMS algorithm works well as it choses intersection over union even if there are interacting palms. A bounding box can be finalized comparatively in short period of time for a higher scene-context perception. Then a feature extractor based on Feature Pyramid Network (FPN) is made functional for object detection. An encoder-decoder feature extractor is manoeuvred akin to FPN, which minimizes the focal loss during training.

Following palm detection across the input frame, hand landmark model uses regression to conduct precise landmark placement of 21 key points within the detected hand regions. It consists of two-layered CNN trained on HGM-4 [38] dataset for real-world hand gesture data along with synthetic hand gestures from Creative Senz3D [39] dataset. This data is annotated with 21 key points over different hand gestures. This model is further trained on combined dataset i.e. real-world and synthetic hand gesture dataset to increase its robustness. The combined dataset contains total 3000 different hand gesture images out which 2000 are taken from real-world and 1000 synthetic images are taken from aforementioned datasets.

3.2 *3D mesh estimation of hand gestures*

2D key points data received from the last model is further processed with depth map estimated hand gesture data. A dataset named FabDepth I, on similar grounds to foreground-background separated hand gestures with depth map is developed [40]. Further 3D annotated dataset is also introduced while training this model for strong supervision. Mediapipe [41] model is introduced to get perfect key points at uniform distance which presents an accurate skeleton of the hand. Mediapipe has number of calculators, which make hand gesture tracking faster and more precise with a minimum number of anchors involved.

A quaternion representation is chosen to give an exact idea about the movement of hand in real time and in 3D space mesh. For developing the final hand gesture model and its 3D mesh estimation, MANO [5] model is incorporated. MANO's surface mesh can be entirely altered and depicted by the geometrical features.

$$M(\beta,\theta) = w\left(T_P(\beta,\theta), J(\beta), \theta, \hat{W}\right) \tag{1}$$

$$T_P(\beta,\theta) = T + B_S(\beta) + B_P(\theta) \tag{2}$$

As shown above, a skin feature w is applied to a rigged dynamic hand mesh with shape T_P, joint positions J establishing a kinematic branch, pose θ, shape β and blend weights $\hat{W}$ all of which are trained on the MANO dataset itself. With the help of this template 3D Hand skeleton shape estimated gestures ready to feed to the next stage of IKNet6 are made available.

3.3 *3D mapping of hand gesture rotation*

To thoroughly understand the hand gesture movements in real-time dynamic system, only 3D skeleton hand is not enough as the application area of this research lies in computer graphics applications such as AR/VR and also 360-degree video. IKNet6 was employed as mentioned before to come up with animated hand also known as hand gesture rotation. This model has many benefits as compared to contemporary networks. Firstly, it trains on motion captured data along with various 3D hand gesture data. This provides full supervision during training, which is not the case with similar networks. Also it has single feed forward pass, which gives it extra speed in operation in comparison with iterative methods tried in the related research.

IKNet6 is further trained on EgoGesture [42] dataset. This dataset focuses on hands and has depth frames and videos of various hand gestures. IKNet6 is a 6-layer fully

connected neural network with batch normalization, and its activation function is sigmoid. Due to better interpolation properties required in the data augmentation stage, the quaternion representation is selected over a horizontal angle representation.

The loss term has four different sections namely L_{cosine}, $L_{1\text{-}2}$, L_{3D} and $L_{norm,}$ therefore the equation becomes,

$$L_{cosine} + L_{1-2} + L_{3D} + L_{norm} \tag{3}$$

where L_{cosine} gives distance between the angles involved, It is connected via the ground truth quaternion Q^G and predicted Q, as seen in,

$$L_{cosine} = \left(1 - Q^G * Q^{-1}\right) \tag{4}$$

where Q^{-1} is the inverse quaternion and $*$ is the product of two terms. $L_{1\text{-}2}$ supports the quaternion presentation of the results and is given by,

$$L_{1-2} = \left\|Q^G - Q\right\|_2^2 \tag{5}$$

L_{3D} gives the measure of loss in 3D representation of hand gestures which can be represented by,

$$L_{3D} = \left\|T^G - D(Q)\right\|_2^2 \tag{6}$$

where T^G is nothing but 3D joints annotation ground truth and D refers to dynamic function. Finally, L_{norm} provides normalization loss involved, which can be represented with a non-normalized $\breve{Q}$ as,

$$L_{norm} = \left\|1 - \breve{Q}\right\|_2^2 \tag{7}$$

4. Results

In this section, we discuss about the framework in terms of instruments used for research experiments along with hyper parameters opted for training the model followed by qualitative and quantitative results, which are finally supported by an ablation study to highlight the significance of the parameters in the proposed design.

4.1 *Instrumentation*

As the system works in real-time, Octacore I5 machine backed with NVIDIA 1080Ti Max Q Graphics Processing Unit (GPU), all three modules running together give avant-grade 100 fps runtime performance speed which is better than contemporary research carried out in recent times. First two modules of the model can run on CPU but with a limited speed of 30 fps. For the last module, GPU is a must for processing 3D reconstruction and animation of hand gestures.

4.2 *Training details*

The hyper parameters are selected to achieve a trade-off between the expected results and the complexity of the model. All three modules are trained with Adam optimizer with a learning rate of 10^{-4}. The batch size for the first module is 32 while for the second one it is 64, both having 50 iterations each. For the third module, the batch size is 64 but the number of iterations are increased to 100. The entire framework is run on PyTorch.

4.3 *Qualitative results*

Demonstration of the applicability of this unique method in a variety of scenarios is given in this subsection, proving that it generalizes effectively to previously unseen data. The first two outputs of figure 3 indicate that the proposed method is effective for swift motions and unclear images due to complex background, as well as tricky stances like holding a pen between fingers in an uneven manner. The third output shows a hand holding a ball in a side pose as well as the fourth one a complicated hand finger gesture being reconstructed with fine precision. The middle part of figure 3 shows key points of hand gestures highlighted in 3D space. In figure 4, it is shown that one can capture biologically different hand shapes such as that of a Kid or a Man with the help of the proposed method. It is worth noting that the finger and palm shapes have been adapted and appear genuine. The results show that estimated hands give a realistic representation of varying inputs.

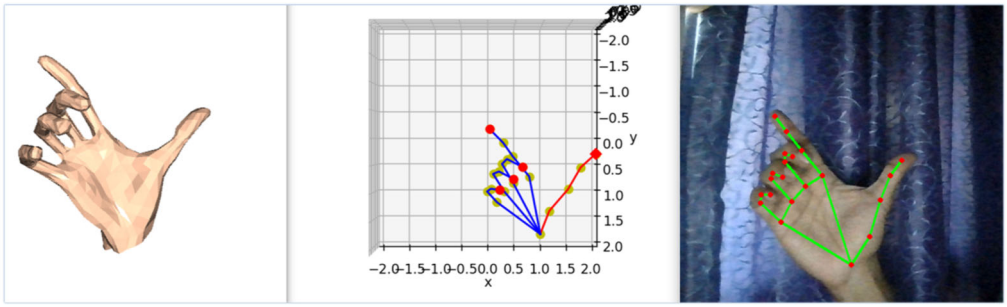
4.4 *Comparative study*

The proposed combinational model is compared with its peers on various datasets. These datasets and benchmarks are selected such that the proposed model is not trained on them previously. Two such datasets as test sets namely DO [21] and ED [23] are selected. Needless to say, these datasets have different numbers of hand sequences. The percentage of correct 3D key points (PCK) and the area under the PCK curve (AUC) are employed as evaluation metrics, with the thresholds ranging from 25 mm to 50 mm. Global alignment is undertaken as pre-synthesis, to precisely measure the local hand pose. The centroid of the finger was aligned for ED and DO.

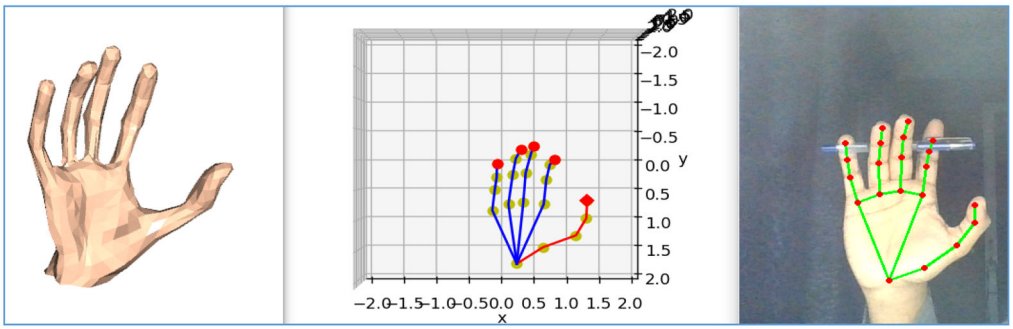
4.5 *Quantitative analysis*

Table 1 gives one to one comparison between the latest techniques incorporated on DO and ED datasets, as none of the models included in the comparative study are trained on them.

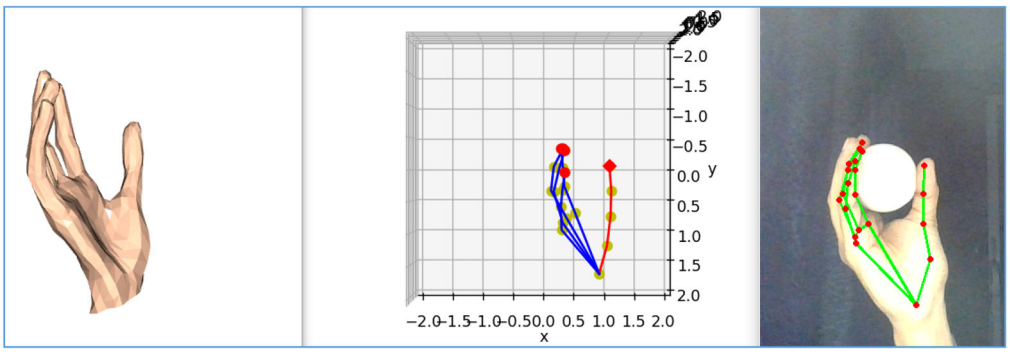
This gives an impartial platform for the analysis of these techniques. As seen in Table 1, the proposed model gives a



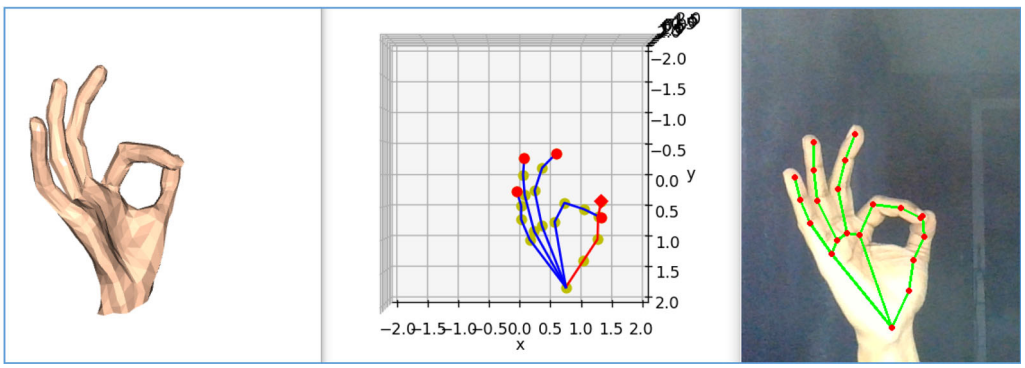
(a)



(b)



(c)



(d)

Figure 3. Examples of results in four scenarios are shown, (a) Noisy background, (b) Self and Object occlusion, (c) Grabbing a ball and (d) A Challenging gesture.

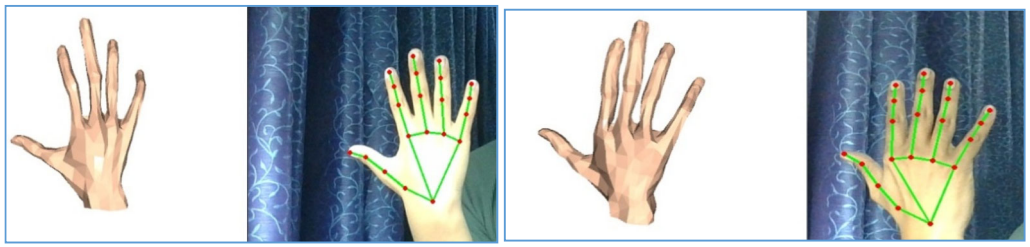


Figure 4. Degree of shape estimation between two different hand shapes. (a) Kid's hand and (b) Man's Hand.

Table 1. Comparative analysis of AUC of PCK

	AUC of PCK	
	Datasets	
Methods	DO	ED
Boukhayma *et al* [1]	.763	.674
Mueller *et al* [9]	.482	–
Zhang *et al* [43]	.825	–
Iqbal *et al* [3]	.672	.543
Proposed System	**.952**	**.825**

superior performance as compared to others and outperforms the rest of the models in both test benchmarks. The basic reason behind it is, being trained on the extra number of datasets especially MoCap and EgoGesture datasets that help in full and strong supervision of the model.

4.6 *Ablation study*

Two separate ablation studies are presented in this subsection. The first one is about the palm detector and 2D landmark module. Key terms are swapped with each other to better understand the proposed design. As observed in Table 2, the decoder with focal loss gives at-par accuracy.

In the second ablation study, final architecture is verified by first presenting the AUC of IkNet6 and then removing the support of the same from module 2 i.e. 3D mesh estimation and hand gesture detection part. Later, the effect of final datasets used without MoCap and then EgoGesture data is studied. The final analysis of the design is performed by removing two key loss terms from module 3.

Table 2. Ablation study of module 1 design.

Variations	Accuracy (%)
Decoder with focal loss	94
Decoder with cross entropy loss	91
Cross entropy loss without decoder	85

Table 3. Ablation study of module 3.

	AUC of PCK	
Different variations in design	DO	ED
Final Design	.952	.825
Without IKNet 6	.918	.803
Without Mocap data	.940	.820
Without EgoGesture data	.936	.824
Without L_{cosine} , $L_{1\text{-}2}$	.945	.819

5. Conclusion

In this study, a combinational approach is introduced to estimate monocular hand posture, shape as well as gestures using data from two fundamentally distinct modalities i.e. image and motion data. The novel neural network design IKNet6 provides 3D representation of an animated hand. As shown in table 1, the characteristics such as accuracy percentage (95.2 on DO dataset and 82.5 on ED dataset), robustness, and runtime (100 fps) show significant advancement over the state-of-the-art networks. For future research, this network can be upgraded to capture and process more than one hand in the given frame through RGB input.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Boukhayma A, Bem R D and Torr P H 2019. 3D Hand Shape and Pose from Images in the Wild *2019 IEEE/CVF*

Conference on Computer Vision and Pattern Recognition (CVPR), 10835–10844

[2] Ge L, Ren Z and Yuan J 2018 Point-to-point regression PointNet for 3D hand pose estimation. *ECCV* 2: 55

[3] Iqbal U, Molchanov P, Breuel T M, Gall J and Kautz J (2018). Hand Pose Estimation via Latent 2.5D Heatmap Regression. *ArXiv, abs/1804.09534*

[4] Shamalik R M and Koli S M 2020 Emergence and functionality of 3D videos. *International Journal of Engineering and Advanced Technology.* 9(3): 4319–4322

[5] Romero J, Tzionas D and Black M J 2017 Embodied hands. *ACM Transactions on Graphics (TOG)* 36: 1–17

[6] Shamalik R M and Koli S M 2021 Real time human gesture recognition: methods, datasets and strategies. *Recent Trends in Intensive Computing* 3: 1445

[7] Glauser O, Wu S, Panozzo D, Hilliges O and Sorkine-Hornung O 2019 Interactive hand pose estimation using a stretch-sensing soft glove. *ACM Transactions on Graphics (TOG)* 38: 1–15

[8] Tompson J, Stein M, LeCun Y and Perlin K 2014 Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (TOG)* 33: 1–10

[9] Mueller F, Bernard F, Sotnychenko O, Mehta D, Sridhar S, Casas D and Theobalt C 2018 GANerated hands for real-time 3D hand tracking from monocular RGB. *IEEE/CVF Conference on Computer Vision and Pattern Recognition* 2018: 49–59

[10] Santavas N, Kansizoglou I, Bampis L, Karakasis E G and Gasteratos A 2021 Attention! a lightweight 2D hand pose estimation approach. *IEEE Sensors Journal* 21: 11488–11496

[11] Li K, Yang L, Zhan X, Lv J, Xu W, Li J and Lu C (2021). ArtiBoost: boosting articulated 3D hand-object pose estimation via online exploration and synthesis. *ArXiv, abs/2109.05488*

[12] Yuan S, Ye Q, Stenger B, Jain S and Kim T (2017). BigHand2.2M benchmark: hand pose dataset and state of the art analysis. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2605-2613

[13] Ng E, Joo H, Ginosar S and Darrell T 2021 Body2Hands: learning to infer 3D hands from conversational gesture body dynamics. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2021: 11860–11869

[14] Grady P, Tang C, Twigg C D, Vo M, Brahmbhatt S and Kemp C C 2021 ContactOpt: optimizing contact to improve grasps. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2021: 1471–1481

[15] Zimmermann C, Argus M and Brox T 2021 Contrastive representation learning for hand shape estimation. *GCPR* 2: 744

[16] Wang J, Mueller F, Bernard F and Theobalt C (2020). Generative Model-Based Loss to the Rescue: A Method to Overcome Annotation Errors for Depth-Based Hand Pose Estimation. *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition* (FG 2020), 101–108

[17] Zhang X, Huang H, Tan J, Xu H, Yang C, Peng G, Wang L and Liu J 2021 Hand image understanding via deep multi-task learning. *IEEE/CVF International Conference on Computer Vision (ICCV)* 2021: 11261–11272

[18] Simon T, Joo H, Matthews I and Sheikh Y 2017 Hand keypoint detection in single images using multiview bootstrapping. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 2017: 4645–4653

[19] Lv J, Xu W, Yang L, Qian S, Mao C and Lu C 2021 HandTailor: towards high-precision monocular 3D hand recovery. *BMVC* 3: 5550

[20] Ge L, Ren Z, Li Y, Xue Z, Wang Y, Cai J and Yuan J 2019 3D hand shape and pose estimation from a single RGB image. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2019: 10825–10834

[21] Chen Y, Tu Z, Kang D, Bao L, Zhang Y, Zhe X, Chen R and Yuan J 2021 Model-based 3D hand reconstruction via self-supervised learning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2021: 10446–10455

[22] Rong Y, Wang J, Liu Z and Loy C C (2021). Monocular 3D reconstruction of interacting hands via collision-aware factorized refinements. *2021 International Conference on 3D Vision (3DV)*, 432-441

[23] Liu S, Jiang H, Xu J, Liu S and Wang X 2021 Semi-supervised 3D hand-object poses estimation with interactions in time. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2021: 14682–14692

[24] Spurr A, Molchanov P, Iqbal U, Kautz J and Hilliges O (2021). Adversarial motion modelling helps semi-supervised hand pose estimation. *ArXiv, abs/2106.05954*

[25] Tkach A, Pauly M and Tagliasacchi A 2016 Sphere-meshes for real-time hand modeling and tracking. *ACM Transactions on Graphics (TOG)* 35: 1–11

[26] Fleishman S, Kliger M, Lerner A and Kutliroff G 2015 ICPIK: inverse kinematics based articulated-ICP. *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* 2015: 28–35

[27] Tagliasacchi A, Schröder M, Tkach A, Bouaziz S, Botsch M and Pauly M 2015 Robust articulated-ICP for real-time hand tracking. *Computer Graphics Forum* 34: 14445

[28] Taylor J, Tankovich V, Tang D, Keskin C, Kim D, Davidson P L, Kowdle A and Izadi S 2017 Articulated distance fields for ultra-fast tracking of hands interacting. *ACM Transactions on Graphics (TOG)* 36: 1–12

[29] Tzionas D, Ballan L, Srikantha A, Aponte P, Pollefeys M and Gall J 2016 Capturing hands in action using discriminative salient points and physics simulation. *International Journal of Computer Vision* 118: 172–193

[30] Sridhar S, Mueller F, Zollhöfer M, Casas D, Oulasvirta A and Theobalt C 2016 Real-time joint tracking of a hand manipulating an object from RGB-D input. *ECCV* 5: 740

[31] Wan C, Probst T, Gool L V and Yao A 2019 Self-supervised 3D hand pose estimation through training by fitting. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2019: 10845–10854

[32] Mueller F, Mehta D, Sotnychenko O, Sridhar S, Casas D and Theobalt C 2017 Real-time hand tracking under occlusion from an egocentric RGB-D sensor. *IEEE International Conference on Computer Vision Workshops (ICCVW)* 2017: 1284–1293

[33] Li S and Lee D 2019 Point-to-pose voting based hand pose estimation using residual permutation equivariant layer. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2019: 11919–11928

[34] Ge L, Liang H, Yuan J and Thalmann D 2019 Real-time 3D hand pose estimation with 3D convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41: 956–970

[35] Huang F, Zeng A, Liu M, Qin J and Xu Q 2018 Structure-aware 3D hourglass network for hand pose estimation from single depth image. *BMVC.* 2: 447

[36] Lin T, Maire M, Belongie S J, Hays J, Perona P, Ramanan D, Dollár P and Zitnick C L 2014 Microsoft COCO: common objects in context. *ECCV.* 35: 1444

[37] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S E, Fu C and Berg A C 2016 SSD: single shot multibox detector. *ECCV* 6: 4008

[38] Hoang V T 2020 HGM-4: a new multi-cameras dataset for hand gesture recognition. *Data in Brief* 30: 211

[39] Memo A, Minto L and Zanuttigh P 2015 Exploiting silhouette descriptors and synthetic data for hand gesture recognition. *STAG.* 2: 7888

[40] Shamalik R M 2022 FabDepth I. *Mendeley Data*. https://doi.org/10.17632/vvdy2x5vpr.1

[41] Lugaresi C, Tang J, Nash H, McClanahan C, Uboweja E, Hays M, Zhang F, Chang C, Yong M G, Lee J, Chang W, Hua W, George M and Grundmann M (2019). MediaPipe: a framework for building perception pipelines. *ArXiv, abs/1906.08172*

[42] Zhang Y, Cao C, Cheng J and Lu H 2018 EgoGesture: a new dataset and benchmark for egocentric hand gesture recognition. *IEEE Transactions on Multimedia* 20: 1038–1050

[43] Zhang X, Li Q, Zhang W and Zheng W (2019). End-to-End Hand Mesh Recovery From a Monocular RGB Image. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2354–2364