IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 30, 2021, accepted September 16, 2021, date of publication September 22, 2021,
date of current version September 30, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3114871

3D Hand Gestures Segmentation and Optimized Classification Using Deep Learning

FAWAD SALAM KHAN[1,2], (Member, IEEE), MOHD NORZALI HAJI MOHD[1], (Senior Member, IEEE), DUR MUHAMMAD SOOMRO[1], SUSAMA BAGCHI[1], (Member, IEEE), AND M. DANIAL KHAN[2]

[1]Faculty of Electrical and Electronics Engineering (FKEE), Universiti Tun Hussain Onn Malaysia, Parit Raja 86400, Malaysia
[2]Department of Machine Learning Innovation, CONVSYS (Pvt) Ltd., Islamabad 45210, Pakistan

Corresponding authors: Fawad Salam Khan (he1980038@siswa.uthm.edu.my) and Mohd Norzali Haji Mohd (norzali@uthm.edu.my)

This work was supported in part by the Ministry of Education, Malaysia, under Fundamental Research Grant K187/FRGS/1/2019/ICT04/UTHM/02/2; and in part by Universiti Tun Hussain Onn Malaysia and CONVSYS (Pvt) Ltd., Islamabad, Pakistan.

ABSTRACT Hand gestures recognition system has massive applications which are mainly utilized in robotics and computer vision specially to control Unmanned Aerial Vehicles (UAV). These methods bypass the presence of electronic control to UAVs and provide an ease to the operators. In this paper, we present a method for 3D hand gestures segmentation and classification by combining MASK-RCNN with Grass Hopper Optimization. We created a private 3D and RGB hand gestures dataset using Intel Kinetic and Intel Real sense d435i camera, then proposed a model for RGB hand gestures to estimate the key points using human kinematics, the key points later then utilize to get the best degree of freedom (DoF). The grass hopper optimization besides minimum distance function was applied to achieve the finest deep features from the 3D hand gestures dataset. The ResNet50 network is used as the backbone to calculate the Overlap Coefficient (OC) for segmentation and the ResNet50, ResNet101 networks to calculate the classification for 3D hand gestures. The classification accuracy achieved on the private dataset is 99.05% and 99.29% on public Microsoft Kinect and Leap Motion dataset where the OC are 88.16%. and 88.19% respectively.

INDEX TERMS 3D hand gestures, grass-hopper optimization, segmentation, depth images.

I. INTRODUCTION

When considering any application for human-computer interaction like UAV control with hand gestures then it required accurate 3D hand pose estimation with key points and gestures recognition at the joint level which has various degrees of freedom (DoF) [1].

3D hand gestures recognitions have various principles which may utilized and are divided into two classes: (1) engineered features, extracted from 3D data, and (2) implicit features, extracted automatically using a deep neural network [2].

Techniques that are recently introduced using novel deep learning are used to bypass the restrictions of typical methods. The analysis of 3D hand gestures used the most current methods for estimating 3D locations from monocular RGB images by understanding hand key points but unable to describe the 3D shape of a hand. In recent years, the pose estimation tasks have massive advancement and this can be accredited to key developments in the field of deep learning and a decrease in the cost of depth sensors [1]. However, the specified problem may exist to face many challenging factors. These factors comprise huge distinctions in hand shapes, viewpoint changes, a large number of degrees of freedom (DoFs), constrained parameter space, self-similarity, and occlusions. Degree of freedom (DoF) is the minimum number of independent variables required to completely describe a body.

There are four types of transformation: translation, rotation, reflection, and dilation. This transformation falls into two categories: the rigid transformation that does not change the shape or size of the object and the non-rigid transformation that changes the size but not the shape of the object. For a rigid transformation of the hand, DoF means the number of independent coordinates needed to define the position of the body. Hand pose estimation means the estimation of the rigid global hand pose and the non-rigid finger articulation. The complexity because of the high degrees of freedom of

The associate editor coordinating the review of this manuscript and approving it for publication was Sunil Karamchandani.

the articulated hand is highly challenging for many visual tracking techniques. There is a greater number of degrees of freedom (DoF) and it will be quite difficult to capture Capturing hand and finger motions from the hand kinematic structure.

The Grass Hopper algorithm provides the best for solving optimization problems and mathematically demonstrated and copies the conduct of grasshopper hordes. The CEC2005 was the first test as the benchmarked by this GOA algorithm for the qualitative and quantitative performance [3].

Different steps are required to solve the optimization problem. Initially, the problem needs to be identified with the help of its parameters. Various parameters have different characteristics through which it can be classified as continuous or discrete. Secondly, recognition of parameters that has different constraints which are applied to the exact problem [4].

The purpose of the research is to reduce the number of variables utilize by optimizing selected features to develop a recognition model for 3D hand gestures where a novel technique establishes by taking feature maps using Mask-RCNN and to minimize the number of input variables by integrating state of art grass hopper optimization algorithm. The motivation behind this research is to utilize the 3D hand recognition model to control the reinforcement leaning based Unmanned Aerial Vehicles (UAVs).

In this paper, a 3D hand gestures recognition method is developed by combining Mask-RCNN for segmentation and Grass-Hopper optimization at the feature map for the classification of hand gestures.

The main contribution of the research can be summarized as:

1. 3D hand gestures recognition method for deep features optimization using grass hopper to calculate the best accuracy for classification.
2. Key points estimation using the human kinematics-based model for RGB hand gestures.

II. RELATED WORK

Pose estimation for 3D hand gestures has commonly been used for detecting sparse joints [5]. Different methods utilized for prediction are applied on only skeletons, our emphasis is to get as output a condensed hand mesh that can gather relations with objects. Very recently, Panteleris *et al.* [6] and Malik *et al.* [7] have produced full hand meshes. However, [8] achieves this as a post-processing step by fitting it to 2D predictions. The hand estimation component used in this research is more similar to [9]. But in contrast to [15], our method takes RGB images as input and not depth. For hand pose estimation in the presence of objects, Mueller *et al.* [10] use 7 objects in a merged reality environment to render artificial hand pose datasets. However, objects only serve the role of occludes. This approach is difficult to scale for a large number of object instances. A novel framework designed using graphs as a part of end-to-end GCN learning. The design principle based on the minimization of a constrained loss to corresponds with convolutional parameters of GCNs also the fundamental contiguousness matrices to capture the topology of input graphs [11]. The 3D joint positions used kinematics where physical forces rendered unique challenges for the measurement for unsuitable supervised learning paradigms [12].

The review from 2007 by Erol *et al.* [13] proposed a taxonomy of approaches for hand tracking. Learning-based approaches are more effective for solving single-frame pose estimation. Optionally it can also be combined with hand model fitting for higher precision, e.g., [14]. The review by Supancic *et al.* [15] demonstrated a comparison of 13 methods on a new dataset. His comparison led to the conclusion that deep models are suitable for pose estimation [18]. It also highlighted that if training sets are in scale, the models then generalize well. The author extended the depth images for the analyses and compare deep learning methods on human fall detection, and assume analysis for error sources and different design scenarios [16].

Vision-based hand pose estimation and tracking have been extensively studied over many years. The successes in performance in this field have been mainly due to two dominating trends: depth image and deep learning [7]. The problem of 3D hand pose tracking has been mainstreaming among researchers of computer vision, as it plays an important role in human-computer interaction such as virtual/augmented reality applications. Over the past few years, hand pose estimation techniques have almost entirely shifted to the use of depth images only, owing to the popularity and availability of Depth sensors such as Intel Real sense and MS Kinect. Depth information resolves many ambiguities present in the monocular RGB input. Secondly, deep learning has transformed the way the vision problem is being solved. With the use of deep neural networks in this field, hand pose estimation has become much easier.

Tan *et al.* [17] formulated 3D tracking of hand joints as an optimization problem. It works by minimizing the inconsistencies between a 3D structure of the hand and the appearance of a hypothesized 3D hand model instance. Qian *et al.* [18] demonstrated a model of a hand by using a number of spheres. After that, a hybrid 5 method was proposed. The hybrid 5 method combines gradient-based and stochastic optimization methods to estimate the 3D hand model. This method not only provided fast convergence but also good accuracy. Also, Chin Yun *et al.* [19] proposed an estimation algorithm for a 3D hand skeleton model. This algorithm makes use of depth images from an Active Shape Model (ASM). Principle Component Analysis (PCA) appearance models can generate a new appearance using a small training set, but its application is limited by linear correlations. Complex scenes, occlusion, and clutters pose serious challenges to these representations. The use led to a need for dimensionality reduction for all hand posture estimation approaches and also to reduce associated computational complexity. Santello *et al.* [20] revealed 90 percent of the variance of the data of grasps directed towards household objects could be described by as little as 3 principal components (PCs). Many other studies have

since supported this view [21] for dimensionality reduction for hand pose estimation and tracking.

The researcher in [22] proposed an algorithm using 3D CNN for hand gesture recognition from challenging depth and intensity data. Their solution associated multiple spatial scales information for the final prediction which employed data augmentation of spatial-temporal to decrease potential overfitting.

In [8], the author provides a comprehensive study for 3D hand pose estimation has mainly focused on its significance in numerous applications of human-computer interaction (HCI) and represents deep learning-based methods and proposed a new taxonomy for input data modality which comprises RGB and depth multimodal information.

III. PROPOSED METHODOLOGY

The novel framework for 3D hand gestures segmentation and classification system is proposed which comprises two steps; 3D Hand Gestures Segmentation through Mask-RCNN and classification of six different types of 3D Hand gestures by implementing feature optimization of deep CNN with Grass Hopper Optimization. There are two blocks shown in Figure 1 in which segmentation block designed for 3D depth regions to be segmented using improved Mask-RCNN whereas classification block proposed for deep features selection process where grass hopper optimization implemented after feature matrix. Our dataset includes Forward Gesture, Backward Gesture, Upward Gesture, Downward Gesture, Left Gesture, and Right Gesture. Also, the Microsoft Kinect and Leap Motion dataset [23], [24] which consists of 10 hand gestures (g1, g2, g3, g4, g5, g6, g7, g8, g9, g10) used for the experimentation purpose. The proposed methodology is revealed in Figure 1.

A. 3D HAND GESTURES SEGMENTATION

We have considered Mask-RCNN [25] for segmentation using 3D Hand gestures depth images. Mask-RCNN provides dominance as compared to other available algorithms for semantic segmentation tasks. This algorithm has different components such as backbone, region proposal network (RPN), ROI aligns, network head, and loss function, as shown in Figure 2. A brief explanation of each component is given below:

1) BACKBONE

It provides a mechanism for feature extraction from the image. It uses a CNN like ResNet50 along with Feature Pyramid Network (FPN) [26] used as a feature extractor. In CNN there are low and high-level layers, so some features like edges are extracted at the lower layer while high-level features are extracted at subsequent layers. We took a dimension of $1024 \times 1024 \times 3$ for from our 3D + RGB hand gestures dataset into a features map of size $32 \times 32 \times 2048$.

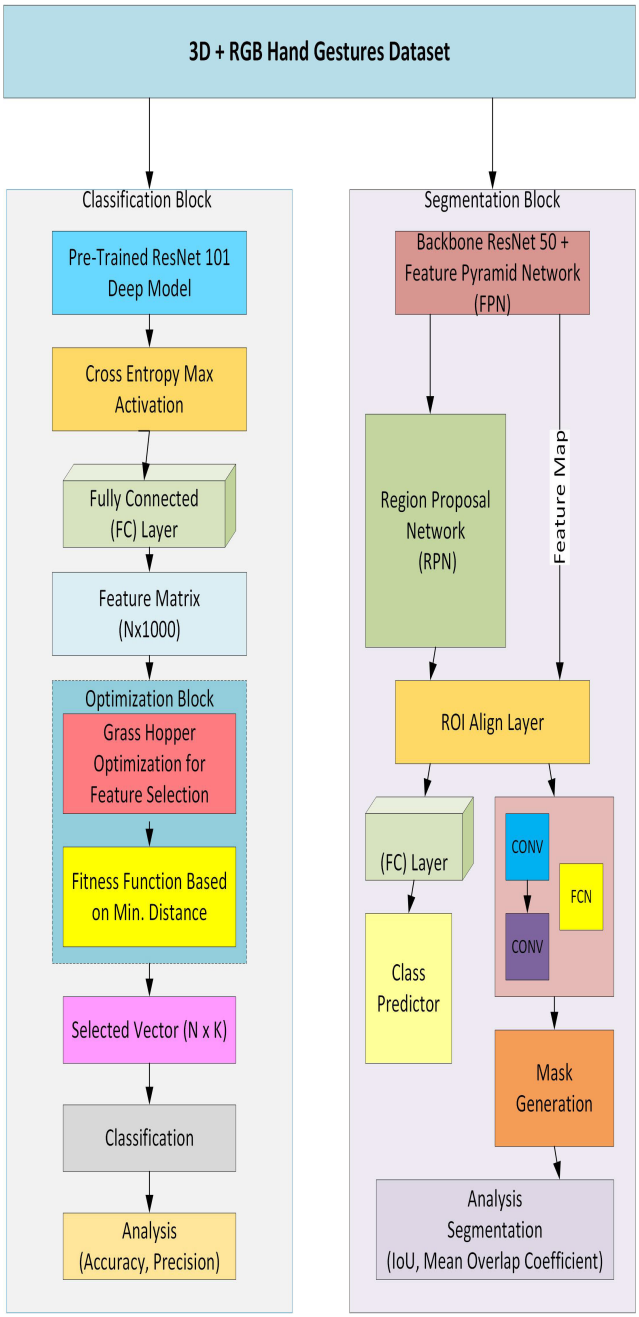

FIGURE 1. Flow diagram of proposed methodology.

2) REGION PROPOSAL NETWORK (RPN)

RPN are first introduced in Faster RCNN [27] which is designed with lightweight CNN and manlily used to replace the process of selective search, as bounding boxes generation using selective search is very slow. RPN takes input from the backbone network ResNet50 + FPN, bounding boxes are produced using these image features of the 3D hand gestures. Anchors are regions scanned by RPN. These anchors boxes disperse over the complete image to overlay maximum areas where overlap anchors are constructed with different sizes and aspect ratios. RPN predictions help to find the highest probability for the N number of anchors. Redundancy is

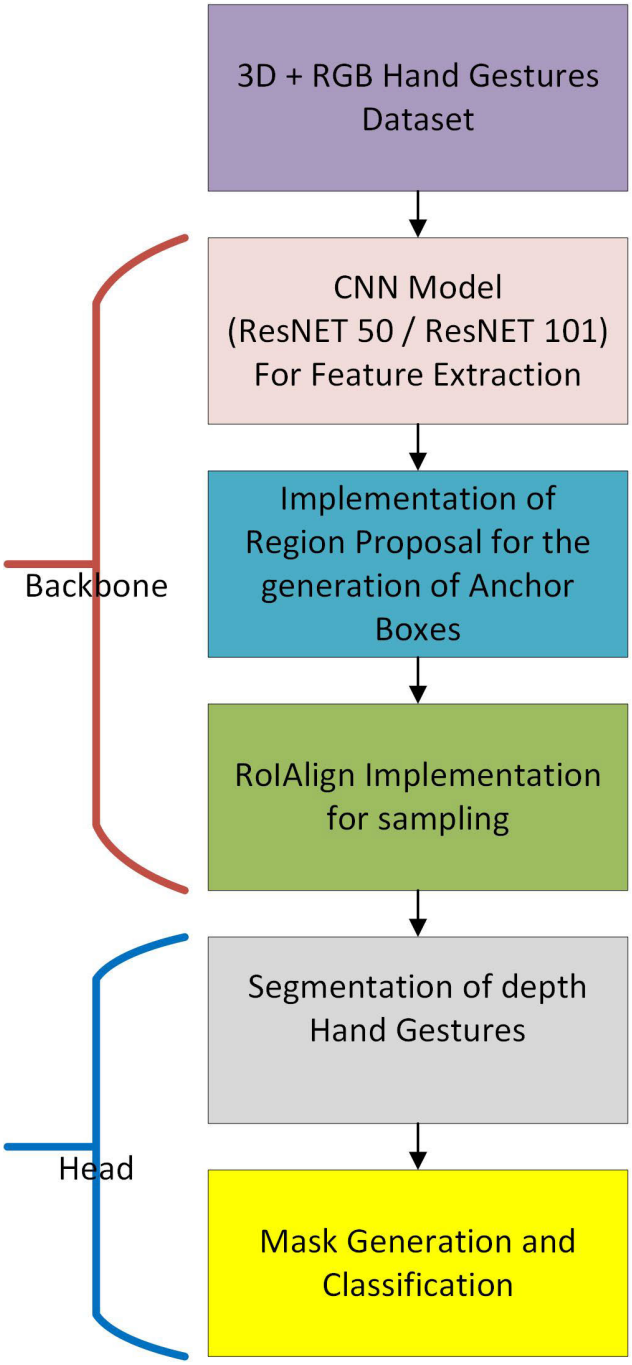

FIGURE 2. 3D + RGB hand gestures using architecture of Mask-RCNN.

reduced using these region proposals overlap, class scores are then computed by applying Non-Maximum Suppression (NMS).

The loss function of RPN is defined as follows:

$$\lambda(bi, B\,i) = \frac{1}{N}\sum_{i} Lc(bi, b\,i') + \psi\frac{1}{Nreg}\sum_{j} bi'Lreg\left(bi, B\,i'\right) \tag{1}$$

The predicted probability is demonstrated by *bi* for the anchor position *i* and describes the ground-truth value 1 or 0. The value of 1 is received when the position of the anchor is positive or negative if it is 0. Now the four points coordinates which are being used to calculate segmented regions bounding box whereas ground truth bounding box is symbolized by *Bi'*.

3) RoI ALIGN

It is used for the prediction of the precise pixel masks of the 3D hand gestures inside the image, as the size of anchor boxes are in different sizes and ratios, to align these features, RoI used RoI pooling where bilinear interpolation technique implemented to compute the precise pixel value of features [28]. The max-pooling and average pooling operations are performed for six often appraised positions with the existing features for further alteration.

4) NETWORK HEAD

The class labels are predicted at network heads where bounding box and mask are produced after the extraction of different features using RoI Align. Then these features described the exact position of bounding boxes where the fully connected layer predict the class labels. The mask of $m \times m$ projected by FCN and the loss function is described by:

$$\lambda = \lambda class + \lambda box + \lambda mask \tag{2}$$

where λ class shows a class loss, λ box demonstrates bounding boxes loss whereas λ mask symbolized during segmentation as the loss of masses generated and the overall loss is computed by λ as the sum of all these losses.

B. EFFECTIVNESS OF GRASS HOPPER OPTIMIZATION

The grasshopper optimization algorithm (GOA) is a meta-heuristic optimization technique introduced in 2017 [3]. The GOA was influenced by the long-distance and sudden migrations of groupings of mature grasshoppers. It is extensively utilized in a range of industrial settings and achieves good results due to its ease of deployment, good precision, and high effectiveness. A hybrid technique has been developed centered upon on grasshopper to improve the parameter of Support Vector Machine (SVM) models [29]. Several optimization issues in automated voltage regular systems were solved using GOA [30]. For optimizing the dispersed trajectory of UAVs in urbanized areas, a dynamical GOA was suggested [31]. The fundamental multi - objective GOA was used to tackle numerous benchmark issues with excellent results [32]. The Grasshopper optimization method prevents local optima while locating global optima in specified spaces. GOA strikes equilibrium between exploration as well as exploitation in order to discover the global optimal solutions to an optimization issue. Grasshoppers can identify the best favorable location in a specified search area. The GOA provides precise answers to unconstrained optimization problems. The optimal location determined by GOA is precise and intriguing. The GOA also produces accurate and intriguing findings for restricted optimization problems [33]. GOA can enhance the randomly initialized populations for a real-world

situation. As iterations go, the objective is enhanced, and the estimation of the global optimum gets increasingly precise according to the degree of iterations. GOA can handle real-world issues with uncertain search spaces. When addressing difficult issues requiring composites test function, GOA correctly utilized exploitation as well as exploration. GOA does have ability to surpass numerous present techniques while tackling a variety of existing or novel optimization challenges [3].

C. FEATURES SELECTION WITH EGH OPTIMIZATION

Extraction of features for 3D and RGB hand gestures estimation is a dynamic research area, especially when considered a different degree of freedom of hand. The features extracted originally hold the amount of extraneous and dispensable features which desire to be detached earlier to ultimate learning. The exclusion of inappropriate information is vital to take care of the uniformity of the suggested model. During this work for feature selection, we have applied an Enhanced Grasshopper Optimization Algorithm (EGOA) and fitness function based on the lowest distance. Using this method, the simplest features are chosen from the feature map. Primarily, features are employed as an Enhanced-Grass-Hopper (EGH) and supported a fitness function for every next iteration which is used to select target EGH. This procedure is nonstop for all features and finally, the simplest vector obtained as an output for final classification. Here we suppose 'F' for the number of features, where each feature denotes an EGH. The fitness function describes the minimum Euclidean distance where each EGH is evaluated after every interaction for which best one is selected because the target moves toward the new EGH.

Scientifically, the position of F^{th} EGH is demonstrated as

$$\Delta F = S(i) + J(i) + Y(i) \tag{3}$$

where S (i) symbolizes the community of features interaction between EGHs'. It is a fundamental component through the method and calculates the Euclidean Distance (ED) among p^{th} and k^{th} EGHs' as:

$$\mathrm{Sp} = \sum S(\varphi pk)\varphi' pk \quad k = 1, k \neq p \tag{4}$$

The symbol φ pk represents ED and a unit vector known here φ' pk of *pth* EGH to *kth* EGH which can be calculated concluded the following mathematical expression:

$$\varphi p\ k = |fk - f\ p| \tag{5}$$

Here the gravity force can be demonstrated by the parameter $\mathbf{G}p$ on *pth* EGH and is mathematically calculated the following equation:

$$G_{p=} - C \times \widehat{eC} \tag{6}$$

The gravitational cost is symbolized as C and the vertical unit vector is denoted by *ec*. Now, the wind advection is denoted by $\mathbf{A}p$ is computed as:

$$Ap = u \times \widehat{ew} \tag{7}$$

Here the constant drift identified by u and due to wind pressure, a unity vector formalizes by ew. Consequently, the position of each EGH employed by $\boldsymbol{Sp}$, $\boldsymbol{Gp}$, and $\boldsymbol{Ap}$ are to update after each iteration. Therefore, after each iteration best EGH is selected as a target whereas other EGHs' change their positions near to the target for the subsequent iteration. Now, in our scenario, all the features are updated in a continuous fashion until the process ends.

Mathematically, each EGH as a portion is updated as:

$$f - \alpha \left(\sum_{k=p,k\neq p}^{N} \alpha \frac{rb\varphi - lb\varphi}{2} s\left(|fk^{\varphi} - fp^{\varphi}|\right) \left(\frac{fk - fp}{\varphi_{pk}}\right) \right. \tag{8}$$

The above equation (8) describes the upcoming position of the grass hopper based on the present position where the upper bound is represented by $\boldsymbol{rb\varphi}$ and the lower bound by $\boldsymbol{lb\varphi}$ for pth dimensions, $\boldsymbol{\alpha}$ is the adaptive parameter used for grass hopper repulsion/attraction forces with the other grasshopper, the 1st part of the equation describe the location of the grass hopper and the 2nd part $\boldsymbol{fk^{\varphi}}$ - $\boldsymbol{fp^{\varphi}}$ indicates the target grass hopper should be repelled or attracted which is linearly decreased the space utilized by the grass hopper to explore and exploit.

The Search Agent at each iteration is considered as the sigmoidal transfer function ΔSG. The probability function used this sigmoidal transfer function for updating the position of each EGH's and is represented by:

$$\Delta^{k}_{q+1}(q+1) = \left\{ \begin{matrix} 1\ if\ rand < \Delta SG + 1 \\ 0\ if\ rand \geq \Delta SG + 1 \end{matrix} \right\} \tag{9}$$

We consider Euclidian distance as a fitness function to considers only the minimum distance features from other features so that higher accuracy can be achieved during classification. Now, SVM is used to get the numerical values of classification from the final selected features.

D. KEY POINTS ESTIMATION

The human hand kinematics framework is made up with five open-loop linkages, all of which are made up of stiff linkages that belong to the hands bone as well as rotating pairings that link to the hands joint. The global movement is represented by the six degrees of freedom (DoF). The two degrees of freedom (DoFs) represent the localized adduction/abduction and also flexion/extension motions. The localized flexion/extension motions are denoted by 1-DoF. The lengthening of the connections in a kinematics framework is consistent. Inside the spectrum of the finger's workspaces, the joint orientations and joint placements can collectively transform. Within an open-loop network, the root joint through every finger act as the foundation of the referencing frame. The shared foundation for five fingers is formed by the wrist or palm joints. Forward kinematics is utilized to calculate the location and angle of the TIP based on the angle of joints at the DIP, IP, and so on. The inverse kinematics is utilized to determine the joint angles based on the TIP's location and angles. The algorithm relies upon

D-H parameter values that correspond to the DoF per each open-loop chain. This should be emphasized that the DoFs of the human hands are not completely self-reliant of one another, particularly those at the same finger. Synergies is the term used to describe the coordinated motions of the human's hand joint [34].We designed a physical model to estimate the 3rd dimension (Z-axis) from the key points from hand pose estimation. As per human hand kinematics, we analyze that the depth is equal to the inverse of the distance between key points of fingers. We have implemented the same approach for all key points. When we see our hand on an RGB image its information is in x and y coordinates, as bending the fingers in downward position then the distance between key points decreases and the depth will increase respectively. ResNet50 network is used as the backbone network for classification using the FastAI framework. It consists of a convolution + max-pooling operation. Now applied element-wise operation called batch normalization as the subsequent step so that it will not change the size of our volume. Finally, max-pooling operation implied (3×3) through a stride value of 2. After the classification of images, we sent these images to our key points model.

E. ARCHITECTURE OF CNN FOR EGH OPTIMIZATION

The architecture of CNN consists of 104 convolution layers with a total of 344 layers. The layers use kernel sizes of 11, 5, and 3. Max pooling layers are connected using strides 4, 2, 1 with zero paddings. The feature maps from convolution layers are of size $12 \times 12 \times 8$. At the end of the architecture is a fully connected layer consist of a list of actions. All convolution layers use ReLU as an activation function. We will train our model by adding the grass hopper optimization method along with the Batch Normalization technique to speed up training. The first convolution layer filters the $100 \times 100 \times 3$ input image. For this, it uses 32 kernels of size $11 \times 11 \times 3$ with a stride of 4 pixels. The stride of 4 pixels is the distance between the receptive field centers of adjacent neurons. The second convolution layer takes as input the output of the first convolution layer (after it is response-normalized and pooled). This second layer uses 64 kernels. The third convolution layer has 64 kernels connected to the (normalized, pooled) outputs of the second convolution layer. The output layers contain the number of parameters of the action to be performed. For our model, the final layer has 6 outputs. The proposed network iterative pursues the target position. To train the network, the training sample needs to be created first. The training sample input consists of image patches and their corresponding bounding box. Transfer learning technique is utilized on the 3D hand gestures for features mapping using the previous trained ImageNet dataset. Then the activation function is used to map these features. For this purpose, we have used the cross-entropy activation function. which is demonstrated by formula as:

$$\mathrm{Hc(fi, C)} = -\sum_{m}^{M} P(v, c) log(P(v, c)) \tag{10}$$

Here classes for the overall number of classifications are symbolized by M and the class labels are denoted by c where probability P for observations V and concluded class formalized by m. We have implemented a cross-entropy activation function mutually on training and testing data. The ratio of 80, 20 percent were formalized as training/testing data and received the output in the form of two feature vectors. Then EGH applied for the optimization of these feature vectors. The Support vector machine is used to train by optimized training vector for final classification.

IV. EXPERIMENTS

The experimentation configuration consists of AWS G4dn.xlarge instance consists of 16 GB Tesla T4 GPU with 16 GB RAM with Cuda drivers. Anaconda and Jupyter Notebook was used for the development environment configured and installed with the required libraries for TensorFlow and Keras for the implementation of the machine learning framework. We used the customized Mask CNN with ResNet-50 and ResNet-101 backbone architectures implemented for 3D hand gestures detection and classification using open-source libraries on AWS GPU.

A. DATASET FOR HAND GESTURES

In this research we have used two different dataset one public Microsoft Kinect Leap Motion dataset and the other private dataset, for the private dataset, the partial dataset was collected at CONVSYS (Pvt) Ltd and the remaining part was taken at the Lab of FKEE Universiti Tun Hussein Onn Malaysia. 25 people participated in the lab for the dataset collection of 2D and 3D hand gestures. We have collected a total of 4200 images (700 Images per class for 6 hand gestures classes) and divided them into training and validation sets with 80% images for training and 20% for validation from each class. As per the calculations 3360 and 840 images for both datasets. The original size of 1024 by 1024 pixels was maintained during training and testing. The image augmentation method has been used to avoid overfitting by performing mirroring and rotation to 45°, 60°, 135°, and 270°. This technique provided 700 images of each class.

B. NETWORK TRAINING

Mask-RCNN provides the best for segmentation and it was chosen because of its capability to have less computation power which reduces time and complexity. ResNet50 was utilized in the segmentation block as the backbone network for feature extraction where as ResNet101 and ResNet50 was used for a higher number of layers and ease in classification and at the same time the proposed integration of grass hopper optimizer may result in improved accuracy of calculation.

C. EVALUATION

The proposed key point estimation method takes input from 2D images which take the pixel values from x and y. Then, the designed function returns the values of x,y,z dimensions, in a similar fashion the key points are drawn using 3D scatter

plot as shown in figure 3. It is quite obvious that key points calculated from human kinematics help to get the best degree of freedom.

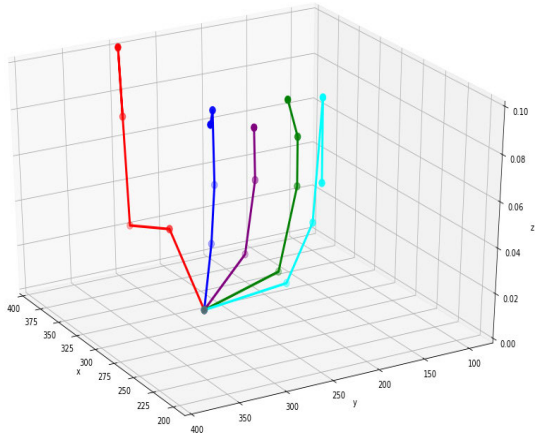

FIGURE 3. Key point estimation from 2D Hand gestures.

FIGURE 4. Binary masks constructed by using the visual location of the 3D image using two extreme points a and b to join the dividing line.

From the proposed method, the results are received from the segmentation block where masks are generated after 70 epochs of training from the training dataset shown in Figure 4. For a given set of 3D depth images, the similarity within the predicted region and ground truth is measured by Intersection Over Union (IoU) which is evaluated for segmentation block. Throughout the evaluation by utilizing two backbones ResNet50 and ResNet101. It is self-evident that the number of iterations rises, the losses decrease in an inversely logarithmic. ResNet 50 shows better convergence as shown in Figure 6 as compared with ResNet101 shown in Figure 7.

The segmentation loss has been observed during training for the identical quantity of epochs where the depth data is extremely stochastic in nature. After 70 epochs, the loss has been converged near to zero, and due to minimal variation of gestures the graph plotted on the semi-log scale along Y-axis.

The target truth bounding box on 3D depth hand gesture image accurately divided so that the ratio of pixel differentiates between the predicted item and the background.

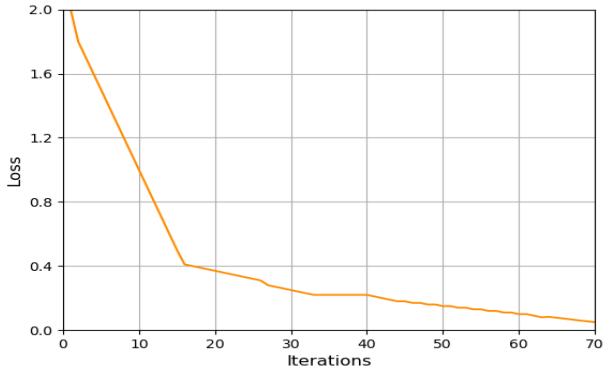

FIGURE 5. Segmentation Loss.

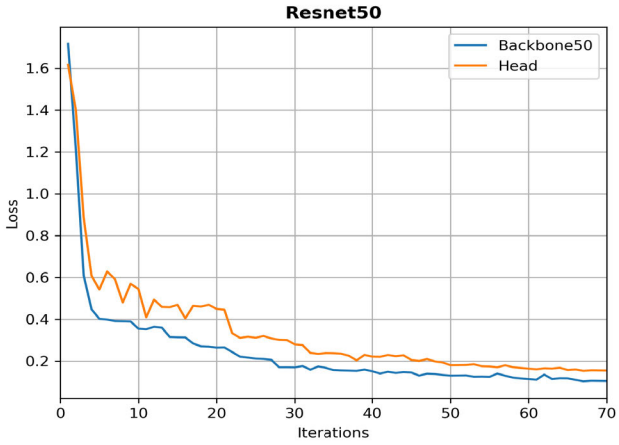

FIGURE 6. Backbone ReNet50 with 50 layers.

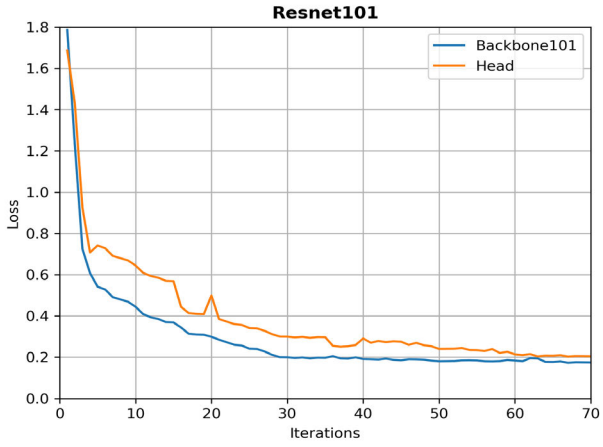

FIGURE 7. Backbone ReNet101 with 101 layers.

Table 1 describes the overlap coefficient calculated based on the two different Average Precisions (AP) which describe the segmentation on the image.

After the training on our dataset, the proposed model file has been saved in.h5 format and then random 3D images were taken for testing, Figure 8 and 9 shows the classification results on two different dataset during inference of 3D hand gestures.

TABLE 1. Overlap-coefficient (oc).

Network Model	Ap (0.75)	Ap (0.5)	Overlap Coefficient
Segmentation Block (ResNet 50)	0.9667	1	0.8816
(ResNet 101)	0.9436	0.99	0.8716

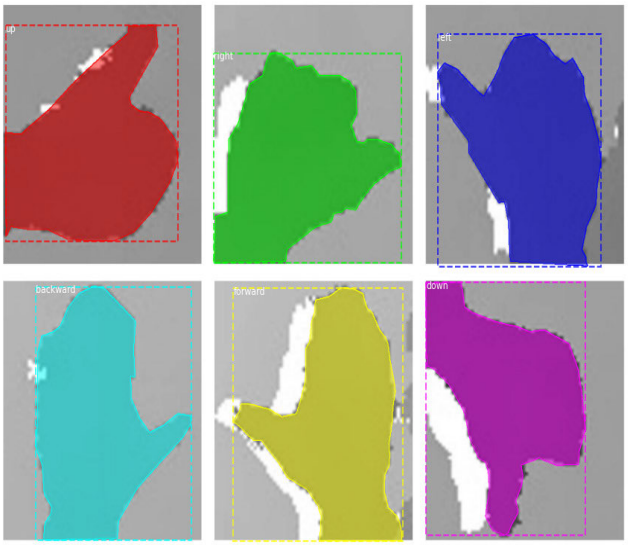

FIGURE 8. Classification of 6 types of 3D hand gestures using private dataset.

FIGURE 9. Classification of 10 types of 3D hand gestures using public microsoft kinect leap motion dataset.

We have used 140 images from the private dataset where each class of six different 3D hand gestures from the validation dataset. Similarly, the Microsoft public data set contains 140 images of each class, there are total of 10 types of 3D hand gestures in which we divided the dataset into training, validation and testing (90, 22 & 28) images respectively. The horizontal axis shows the predicted labels while at vertical axis shows the ground truth. The concluded classification accuracy achieved on private dataset is 99.05% and 99.29% on public dataset.

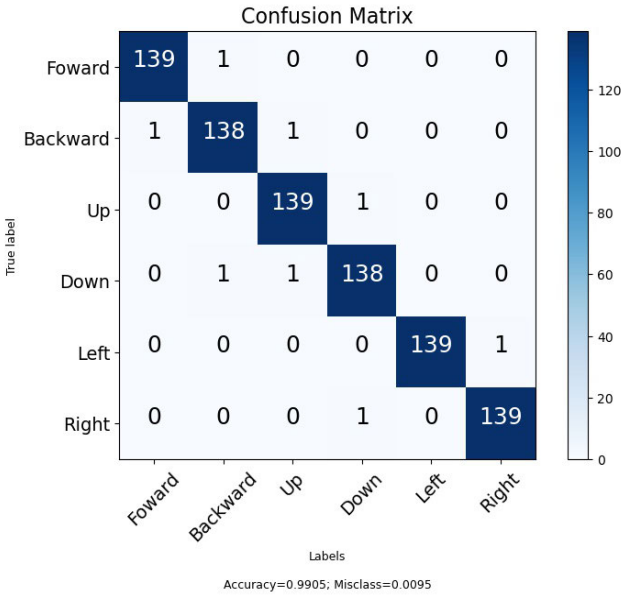

FIGURE 10. 3D hand gestures classification on private dataset, the diagonal values are the true positives labels which are successfully classified.

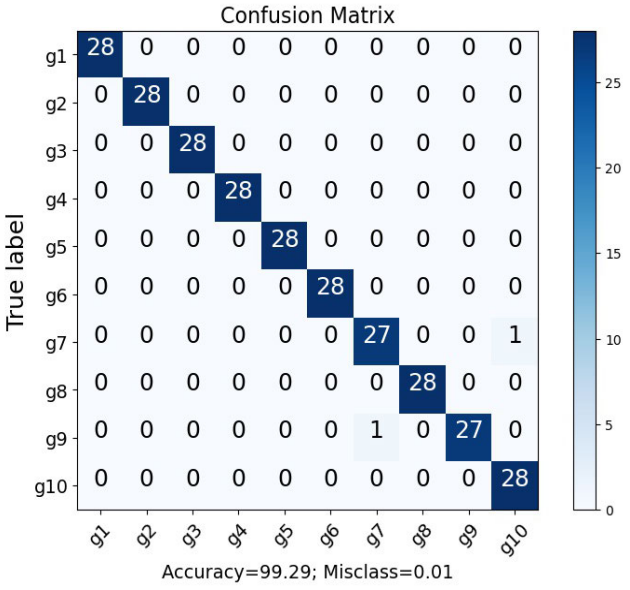

FIGURE 11. 3D Hand gestures classification on microsoft kinect leap motion Dataset, the diagonal values are the true positives labels which are successfully classified.

D. COMPARATIVE ANALYSIS

For the challenge of hand gesture identification utilizing depth data provided by different researchers, there are various approaches using different custom and pubic dataset on 3D and RGB. These techniques include creating filters, cluster,

TABLE 2. Comparative analysis.

Paper	Method	Segmentation/ Accuracy (%)
H. Sahbi et al. [11]	LWG-CNN	86.78
Molchanov et al. [22]	CNN, 2 sub-nets	77.5
Molchanov et al. [35]	Recurrent 3D CNN	98.6
Zhu et al. [15]	3D CNN + LSTM	98.89
Zhang et al. [16]	ConvLSTM	55.98 / 95.13
N. Nishida et al [17]	MRNN	97.8
Kingan et al. [2]	PointNet +Att.Mod	94.2
Proposed	**Mask-RCNN + GH**	**88.16 / 99.05**
RGB + Depth		**88.19/ 99.29**

and identification the hand in the region of concern as well as hand-forearm segment, a bespoke hand gesture dataset with multiple stage hand segmentation with three similar datasets (Viva Challenge Dataset, SKIG, JGD) are examined for comparative study on 3D point cloud dataset and a 2D images dataset. The environment consists of NVidia 1050 and 1080 GPUs with TensorFlow and Keras. Table 2 describes the comparative analysis between our proposed method with other researchers.

E. DISCUSSION

In gesture recognition generally, backgrounds are less informative wherever the object presence has less consideration during classification or segmentation of gestures. So, gestures recognition may be effective when complex backgrounds carry negative instances. Therefore, the constructed feature maps with the help of optimized features become vital for gesture recognition methods. Mask Regional Convolutional Neural networks are well designed for the required feature extraction especially when considering the segmentation, these networks are more suitable for variable size and ratio of hand gestures. Therefore, the integration of the grass hopper optimization method and may provide an excellent framework for robust gesture recognition, classification, and segmentation. Fine-tuning on pre-trained models ResNet50 and ResNet101 are being fine-tuned to prevent overfitting for relatively small datasets. In the case of hand gestures recognition, the degree of freedom plays a vital role if considering 2D to 3D conversions, similarly, human kinematics provides a better solution for understanding how the depth is equal to the inverse of the distance between key points of fingers. The proposed method not only provides the segmentation but also provided better classification accuracy. The comparative analysis shows the dominance both in terms of segmentation and classification between the previous methods suggested by other researchers. [2] used precise hand-forearm segmentation algorithm using principal component analysis (PCA) in which first calculated the angels between the arm and depth dimensions then scale the data to separate the hand data from the forearm data, lastly PCA applied to calculate the symmetry axis of the arms. The non-predictive features reside in memory during training cycles which affects the accuracy of the model. Sorting out these features with appropriate feature selection algorithms results in better accuracy. [11] used light weight graph convolutional neural network based on skeletal hand gestures where two constraints utilized for training parameters which only improved the 3D hand pose skeletal estimations with the objective function used to reduce the abrupt noise during training.

As we have considered only 6 hand gestures from private dataset and 10 hand gestures from public dataset which may be used to control the UAV, so as the results there are some difficulties found during experimentations 1) The recognition of similar gestures is very difficult as we considered originally three gestures and the remaining three are of opposite direction 2) computational time increases after integrating the grass hopper optimization. 3) AWS GPU took 14 hours for training using G4 instances for 70 epochs. 4) 3D hand gestures from the whole scene are not isolated if the annotation has any wrong bounding boxes 4) All the key points are not reserved by uniform down-sampling where static gesture contains useless information during down sampling, thus such kind of gestures are also difficult to recognize. 5) The sequence of hand gestures with less visibility due to depth is not fully recognized in the experiments. If both the global and local features are recognized at the same time then it can improve the recognition accuracy. Alternative skills such as normalization may affect the estimation of key points information for recognition of 3D hand gestures.

V. CONCLUSION

A new method consists of a segmentation and classification framework which is proposed for 3D + RGB hand gestures segmentation and classification using grass hopper optimization technique with a fitness function. The evaluation of performance is made on the proposed method which has different phases; first, the analysis of Mask-RCNN performance evaluation with the help of overlap coefficient. Then from the classification block on the proposed method, the computed classification accuracy shows dominance as compared with the previous techniques proposed by another researcher. The complete results from the proposed architecture work strongly for both 3D and RGB hand gestures segmentation and classification. The overlap-coefficient is computed as 88.16% from our validation dataset for 3D hand segmentation and 99.05% classification accuracy and 99.29% using Microsoft Kinect Leap Motion dataset. From the results, we concluded that the hybrid Mask-RCNN with Grass Hopper Optimizer provides dominance based on 3D hand gestures segmentation which increased in the overlap coefficient rate using 4200 images (700 images per class). Moreover, we also accomplish to know that the classification

step using a grass hopper improves the performance and provides the best-selected features for classification.

REFERENCES

[1] J. Malik, I. Abdelaziz, A. Elhayek, S. Shimada, S. A. Ali, V. Golyanik, C. Theobalt, and D. Stricker, "HandVoxNet: Deep voxel-based network for 3D hand shape and pose estimation from a single depth map," 2020, *arXiv:2004.01588*. [Online]. Available: http://arxiv.org/abs/2004.01588

[2] R. Mirsu, G. Simion, C. D. Caleanu, and I. M. Pop-Calimanu, "A pointnet-based solution for 3D hand gesture recognition," *Sensors*, vol. 20, no. 11, pp. 1–13, 2020, doi: 10.3390/s20113226.

[3] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: Theory and application," *Adv. Eng. Softw.*, vol. 105, pp. 30–47, Mar. 2017, doi: 10.1016/j.advengsoft.2017.01.004.

[4] C. A. C. Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art," *Comput. Methods Appl. Mech. Eng.*, vol. 191, nos. 11–12, pp. 1245–1287, Jan. 2002.

[5] J. Malik, A. Elhayek, F. Nunnari, K. Varanasi, K. Tamaddon, A. Heloir, and D. Stricker, "DeepHPS: End-to-end estimation of 3D hand pose and shape by learning from synthetic depth," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 110–119, doi: 10.1109/3DV.2018.00023.

[6] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt, "GANerated hands for real-time 3D hand tracking from monocular RGB," 2017, *arXiv:1712.01057*. [Online]. Available: http://arxiv.org/abs/1712.01057

[7] J. Malik, A. Elhayek, F. Nunnari, and D. Stricker, "Simple and effective deep hand shape and pose regression from a single depth image," *Comput. Graph.*, vol. 85, pp. 85–91, Dec. 2019, doi: 10.1016/j.cag.2019.10.002.

[8] T. Chatzis, A. Stergioulas, D. Konstantinidis, K. Dimitropoulos, and P. Daras, "A comprehensive study on deep learning-based 3d hand pose estimation methods," *Appl. Sci.*, vol. 10, no. 19, pp. 1–27, 2020, doi: 10.3390/app10196850.

[9] G. Yoder and L. Yin, "Real-time hand detection and gesture tracking with GMM and model adaptation," in *Advances in Visual Computing* (Lecture Notes in Computer Science), vol. 5876, 2009, pp. 387–396, doi: 10.1007/978-3-642-10520-3_36.

[10] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt, "Real-time hand tracking under occlusion from an egocentric RGB-D sensor," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1163–1172, doi: 10.1109/ICCV.2017.131.

[11] H. Sahbi, "Skeleton-based hand-gesture recognition with lightweight graph convolutional networks," Tech. Rep., 2021, pp. 1–10.

[12] Y. Yuan, S.-E. Wei, T. Simon, K. Kitani, and J. Saragih, "SimPoE: Simulated character control for 3D human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7159–7169.

[13] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Y. Chang, K. M. Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge, and J. Yuan, "Depth-based 3D hand pose estimation: From current achievements to future goals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2636–2645, doi: 10.1109/CVPR.2018.00279.

[14] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, T. Sharp, E. Soto, D. Sweeney, J. Valentin, B. Luff, A. Topalian, E. Wood, S. Khamis, P. Kohli, S. Izadi, R. Banks, A. Fitzgibbon, and J. Shotton, "Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–12, Jul. 2016, doi: 10.1145/2897824.2925965.

[15] J. S. Supančič III, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan, "Depth-based hand pose estimation: Methods, data, and challenges," *Int. J. Comput. Vis.*, vol. 126, pp. 1180–1198, Nov. 2018, doi: 10.1007/s11263-018-1081-7.

[16] Y. Nizam, M. N. H. Mohd, and M. M. A. Jamil, "Human fall detection from depth images using position and velocity of subject," *Proc. Comput. Sci.*, vol. 105, pp. 131–137, Jan. 2017, doi: 10.1016/j.procs.2017.01.191.

[17] A. M. Tan, H. Yang, Y. Sen Li, Z. M. Bi, M. Zhang, and Z. T. Wang, "Study on the chemical constituents of Ligularia dictyoneura," *Chin. Pharm. J.*, vol. 39, no. 7, pp. 498–499, 2004.

[18] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun, "Realtime and robust hand tracking from depth," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1106–1113, doi: 10.1109/CVPR.2014.145.

[19] C.-Y. Fan, M.-H. Lin, T.-F. Su, S.-H. Lai, and C.-H. Yu, "3D hand skeleton model estimation from a depth image," in *Proc. 14th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2015, pp. 489–492, doi: 10.1109/MVA.2015.7153237.

[20] M. Santello, M. Flanders, and J. F. Soechting, "Postural hand synergies for tool use," *J. Neurosci.*, vol. 18, no. 23, pp. 10105–10115, 1998.

[21] A. Daffertshofer, C. J. C. Lamoth, O. G. Meijer, and P. J. Beek, "PCA in studying coordination and variability: A tutorial," *Clin. Biomech.*, vol. 19, no. 4, pp. 415–428, 2004, doi: 10.1016/j.clinbiomech.2004.01.005.

[22] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 1–7.

[23] A. Memo, L. Minto, and P. Zanuttigh, "Exploiting silhouette descriptors and synthetic data for hand gesture recognition," in *Proc. Ital. Chapter Conf. Smart Tools Apps Comput. Graph. (STAG)*, 2015, pp. 15–23, doi: 10.2312/stag.20151288.

[24] A. Memo and P. Zanuttigh, "Head-mounted gesture controlled interface for human-computer interaction," *Multimedia Tools Appl.*, vol. 77, no. 1, pp. 27–53, 2018.

[25] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[26] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[27] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[28] M. A. Khan, M. A. Khan, F. Ahmed, M. Mittal, L. M. Goyal, D. Jude Hemanth, and S. C. Satapathy, "Gastrointestinal diseases segmentation and classification based on duo-deep architectures," *Pattern Recognit. Lett.*, vol. 131, pp. 193–204, Mar. 2020, doi: 10.1016/j.patrec.2019.12.024.

[29] I. Aljarah, A. M. Al-Zoubi, H. Faris, M. A. Hassonah, S. Mirjalili, and H. Saadeh, "Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm," *Cogn. Comput.*, vol. 10, no. 3, pp. 478–495, 2018, doi: 10.1007/s12559-017-9542-9.

[30] B. Hekimoglu and S. Ekinci, "Grasshopper optimization algorithm for automatic voltage regulator system," in *Proc. 5th Int. Conf. Electr. Electron. Eng. (ICEEE)*, May 2018, pp. 152–156, doi: 10.1109/ICEEE2.2018.8391320.

[31] J. Wu, H. Wang, N. Li, P. Yao, Y. Huang, Z. Su, and Y. Yu, "Distributed trajectory optimization for multiple solar-powered UAVs target tracking in urban environment by adaptive grasshopper optimization algorithm," *Aerosp. Sci. Technol.*, vol. 70, pp. 497–510, Nov. 2017, doi: 10.1016/j.ast.2017.08.037.

[32] S. Z. Mirjalili, S. Mirjalili, S. Saremi, H. Faris, and I. Aljarah, "Grasshopper optimization algorithm for multi-objective optimization problems," *Appl. Intell.*, vol. 48, no. 4, pp. 805–820, 2018, doi: 10.1007/s10489-017-1019-8.

[33] A. G. Neve, G. M. Kakandikar, and O. Kulkarni, "Application of grasshopper optimization algorithm for constrained and unconstrained test functions," *Int. J. Swarm Intell. Evol. Comput.*, vol. 6, no. 3, p. 2, 2017, doi: 10.4172/2090-4908.1000165.

[34] R. Li, H. Wang, and Z. Liu, "Survey on mapping human hand motion to robotic hands for teleoperation," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Feb. 8, 2021, doi: 10.1109/TCSVT.2021.3057992.

[35] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, p. 3.

FAWAD SALAM KHAN (Member, IEEE) received the B.S. degree in computer engineering from SSUET, Pakistan, in 2002, the M.E. degree in computer engineering from NEDUET, Pakistan, in 2011, and the M.S. degree in computer science from IIUI, Pakistan, in 2020. He is currently pursuing the Ph.D. degree in electrical engineering with Universiti Tun Hussain Onn Malaysia. He is currently affiliated with Pakistan's top AI company CONVSYS (Pvt) Ltd., as the Director and a CEO heading different AI, machine, and deep-learning projects of Silicon Valley, USA; Malaysia; and the U.K. He is a Professional Member of PEC Pakistan. He has more than 17 years of experience in industry and academia. He is also teaching various computer engineering and sciences subjects in different universities in Pakistan. He has received various gold and silver medals for different AI projects in Malaysia.

MOHD NORZALI HAJI MOHD (Senior Member, IEEE) received the Diploma degree in computer engineering from Toyama Maritime College, Japan, in 2000, and the B.Eng. and M.Eng. degrees from Fukui University, Japan, in 2002 and 2004, respectively, and the Ph.D. degree from the Department of Information Sciences and Biomedical Engineering, Kagoshima University, Japan. He is currently working as a Senior Lecturer, a Faculty Lab Manager (HoD), and a Former Industrial Training Coordinator with the Department of Computer Engineering, Faculty of Electrical and Electronic Engineering (FKEE), Universiti Tun Hussein Onn Malaysia (UTHM).

DUR MUHAMMAD SOOMRO received the bachelor's and master's degrees in power electrical engineering from Mehran University of Engineering and Technology, Pakistan, in 1990 and 2002, respectively, and the Ph.D. degree in electrical engineering from UTM, Malaysia, in 2011. He is currently an Associate Professor affiliated with the Faculty of Electrical and Electronics Engineering (FKEE), University Tun Hussein Onn Malaysia. He is the author or coauthor of multiple papers, book chapters, and conferences proceedings.

SUSAMA BAGCHI (Member, IEEE) received the B.E. degree in electronics and communication engineering (ECE) from the University of ent from the University Burdwan, India, in 2002, and the M.Sc. degree in modern communication technologies with business management from the University of Sussex, U.K., in 2007. She is currently pursuing the Ph.D. degree in electrical and electronic engineering with Universiti Tun Hussein Onn Malaysia (UTHM). She has eight years of engineering management experience in MNCs. Since 2020, she has been a Graduate Research Assistant with the Faculty of Electrical and Electronic Engineering (FKEE), UTHM.

M. DANIAL KHAN received the B.S. degree in electrical engineering from HITEC University Taxila, Pakistan. He is currently working at CONVSYS (Pvt) Ltd., Islamabad, Pakistan, as a Design Engineer. He has four years of experience in designing AI solutions for Silicon Valley, the U.K., and Malaysia. His major expertise is in python, C++, TensorFlow, Keras, and CNN modeling. He has completed various projects such as automated AI fitness trainer, fatigue detection system, and fashion design AI pipeline using deep learning. He is also involved in the training of AI, ML, and deep learning to customers in Pakistan and abroad.