# Holoscopic 3D Microgesture Recognition by Deep Neural Network Model based on Viewpoint Images and Decision Fusion

Yi Liu, *Student Member, IEEE*, Min Peng, Mohammad Rafiq Swash, *Member, IEEE,* Tong Chen, Rui Qin, Hongying Meng, *Senior Member, IEEE,*

***Abstract*—Finger microgestures have been widely used in human computer interaction (HCI), particularly for interactive applications, such as virtual reality (VR) and augmented reality (AR) technologies, to provide immersive experience. However, traditional 2D image-based microgesture recognition suffers from low accuracy due to the limitations of 2D imaging sensors, which have no depth information. In this paper, we proposed an innovative 3D microgesture recognition system based on a holoscopic 3D imaging sensor. Due to the lack of holoscopic 3D datasets, a comprehensive holoscopic 3D microgesture (HoMG) database is created and used to develop a robust 3D microgesture recognition method. Then, a fast algorithm is proposed to extract multi-viewpoint images from one holoscopic image. Furthermore, we applied a CNN model with an attention-based residual block to each viewpoint image to improve the algorithm performance. Finally, bagging classification tree decision-level fusion is applied to combine the predictions. The experimental results demonstrate that the proposed method outperforms state-of-the-art methods and delivers a better accuracy than existing methods.**

***Index Terms*—Microgesture Recognition, Holoscopic 3D Imaging, Deep Learning, Decision Fusion**

## I. Introduction

WITH the rapid development of immersive and wearable technologies, and increasing importance to user experience, human computer interaction (HCI) has become a hot research topic due to the need to empower immersive applications with touch-less interaction [1]. In recent decades, as a low-cost and nonintrusive interaction method, gestures have become an attractive sensing modality for many applications [2]. Meanwhile, virtual reality (VR) and augmented reality (AR) technologies demand touch-less and continuous 3D gesture control in many situations.

As a vital nonverbal human conversational interaction, gestures belong to human natural body language [3]. In contrast to sign language, which includes full complex languages and consists of complex grammar systems, gestures have the function of communicating a specific, single command. In the past several decades, great efforts have been made in using gestures for control. Gestures were first used to control keyboards and mice as input methods. 2D gestures are now a common input method on HCI platforms such as touch screen devices. Gesture applications have become attractive for gaming, home appliances and other Internet of Things (IoT) applications in industry.

There have been intensive developments based on 2D vision, time-of-flight (ToF) and radar sensing technologies, although 2D vision and ToF sensors suffer poor performance and a lack of accuracy. Radar technology is still rather expensive and unaffordable for this application. As a result, there is a need for reliable and robust 3D microgestures to satisfy the needs of immersive applications. For instance, the Microsoft Kinect (ToF camera) is a popular choice for entertainment and research areas that have lower accuracy requirements in capturing gestures [4]. There has been some development of ToF cameras allowing them to provide reliable distance measurements based on the speed of light; they have been used in many research areas such as computer graphics, computer vision and HCI. A radar-based sensor was developed by the Google Soli project [5], which proposed a low-power, high-resolution sensor for gesture capture [6]. This type of sensor performed dynamic gesture capture and enriched data diversification in the gesture interaction area. Nevertheless, the drawback of these sensors is the lack of an integrated system linking hardware, software and algorithms. The high cost of the sensors limits their use and application.

The growth of sensing technology has a highly beneficial impact on HCI in the design of innovative and seamless experiences. However, maintaining low cost while also obtaining full-range and high-resolution 3D scenes of static data is a difficult task. For dynamic scenes, most sensors are incapable of detecting objects in 3D scenes. Even though stereoscopic 3D cameras, laser scanning techniques, and radar sensors have been used to enhance the sensing ability for capturing high-quality data, their drawbacks are their complex data fusion and costs. As a result, there is a need for effective 3D sensing technology to empower touch-less interaction for fingers.

A holoscopic 3D (H3D) imaging system is one potential technique to address the challenges facing gesture control. H3D is a true 3D imaging principle and mimics the principle of the flys eye [7] technique to capture a true 3D scene as a module of light on a 2D surface. A 3D scene is reconstructed at different depth levels and with angle information from an H3D image [8]. The H3D system provides rich multidimensional information both visually and motionally, therein satisfying the higher demands of interactive user experiences and supporting highly accurate finger movement capture.

Recently, we designed and prepared the first holoscopic 3D

Y. Liu, M. Rafiq Swash, R. Qin, H. Meng, are with the Department of Electronic and Computer Engineering, Brunel University London, UK. e-mail: {yi.liu, rafiq.swash, rui.qin, hongying.meng}@brunel.ac.uk.

M. Peng and T. Chen are with Southwest University, China.

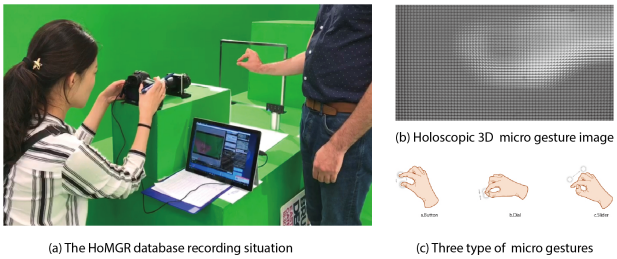


Fig. 1. Holoscopic 3D microgesture image capture. (a) recording setting, (b) obtained H3D image, (c) three types of microgestures.

microgesture (HoMG) database [9] and made it publicly available. Fig. 1 (a) shows a HoMG database recording scenario. Four positions were set up to capture a gesture from different distances and with different sides of the hand. The obtained image is shown in Fig. 1 (b), where 3D information was embedded inside of the H3D image. Fig. 1 (c) presents the three gesture types, i.e., button, dial and slider, used in the capture process. We also organized the first holoscopic microgesture recognition challenge (HoMGR 2018, http://3dvie.co.uk/) and attracted researchers from all over the world to complete in this challenge [10] [11] [12] [13]. Although significant progress has been made on the performance of microgesture recognition based on H3D imaging, there are two key problems that have not yet been solved. The first is that the pre-processing of the H3D image is insufficient because the attempts to extract detailed 3D information from H3D images were unsuccessful. The second is that the microgesture recognition rate is still unsatisfactory for real-world applications due to the limitations of the existing methods. In this paper, we focus on solving these two problems by proposing a new H3D image pre-processing method that takes advantage of 3D information and then uses advanced deep neural models for pattern recognition. In addition, we investigate several decision fusion approaches to combine predictions from multiple viewpoint recognitions.

The main contributions of this work are as follows:

- We introduced the first public HoMG dataset for holoscopic microgesture recognition research.
- We proposed a fast and robust approach for fully automated viewpoint image extraction from H3D images.
- We modified deep neural network models and applied them to each viewpoint image with attention based on residual blocks for feature extraction and classification.
- We proposed several decision-level fusion methods to combine predictions from multiple viewpoints that outperform all existing methods.

The remainder of the paper is organized as follows: Section II reviews some related works about microgesture recognition. Section III provides detailed information about our proposed framework, including fast pre-processing, viewpoint (VP) image extraction, deep neural network based image classification and decision-level fusion. Section IV presents the experimental results and discussion. Finally, section V gives the conclusion.

## II. Related Work

In the past decade, 3D hand gestures have been extensively studied with several datasets created for analysis. Cheng et al. [14] summarized four types of 3D hand gesture datasets. First, static gesture datasets usually capture the finger and palm postures in the RGB-D domain, which can represent basic symbols such as Arabic numerals. Second, trajectory gesture datasets were created to capture and detect hand or body movement using skeleton trajectory. Third, hybrid gesture datasets were produced, which contained mixed vision-based gesture data and trajectory hand postures. The last dataset is the 3D American Sign Language (ASL) dataset, which captured hand movements in a video. For trajectory gesture data, a Kinect is normally used as the sensor, and the data are the skeletal tracking of hand or human body joints. These data are good for 3D modeling and applications in depth-based hand detection and tracking. However, skeletal tracking is not sufficiently accurate for microgesture analysis [15]. The Leap Motion and Google Soli projects focus on tracking user fingers but do not seem to be usable in practical applications. Overall, although they have certain advantages, the drawbacks are obvious. Most datasets based on these sensors focus on macro body motion and sign language. There is no public microgesture dataset available for microgesture control in VR and AR applications.

For 3D hand gesture recognition, recent work has focused on RGB-D video data [16] [17] [19] or hand skeleton sequence data [18]. In 2015, Ming [16] proposed a 3D Mesh MoSIFT feature descriptor for hand activity recognition using an improved graph cuts method on hand segmentation and tracking, combined with 3D geometric characteristics and human behavior prior information. In 2018, Narayana et al. [17] proposed a FOANet deep network architecture that consists of a separate channel for every focus region (global, left hand, and right hand) and modality (RGB, depth, RGB ow and depth ow). The video level predictions from 12 channels are stacked together and fused to the gesture type. In 2019, Nguyen et al. [18] proposed a spatial-temporal and temporal-spatial hand gesture recognition network (ST-TS-HGR-NET) that consisted of three deep networks. These networks captured both spatial and temporal information of the skeleton sequences and were combined to generate better predictions. Abavisani et al. [19] used the 3D-CNN model for each modality (EGB, depth, and optical flow) of the RGB-D video data and combined them for hand gesture recognition. Although these methods cannot be directly used here because our data are images rather than videos or sequences, the ideas behind them are encouraging. For example, focus regions can highlight the key area of the gesture. The fusion scheme can combine multiple predictions to achieve improved performance.

Research on holoscopic 3D microgesture recognition has been accelerated after the first holoscopic 3D microgesture database (HoMG) was created [9]. Additionally, the HoMG database was made publicly available, and the first HoMGR challenge competition was held [9]. Created by using the H3D imaging system, the 3D microgesture database supported high-resolution static image data as well as high-quality dynamic

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/THMS.2020.3047914,


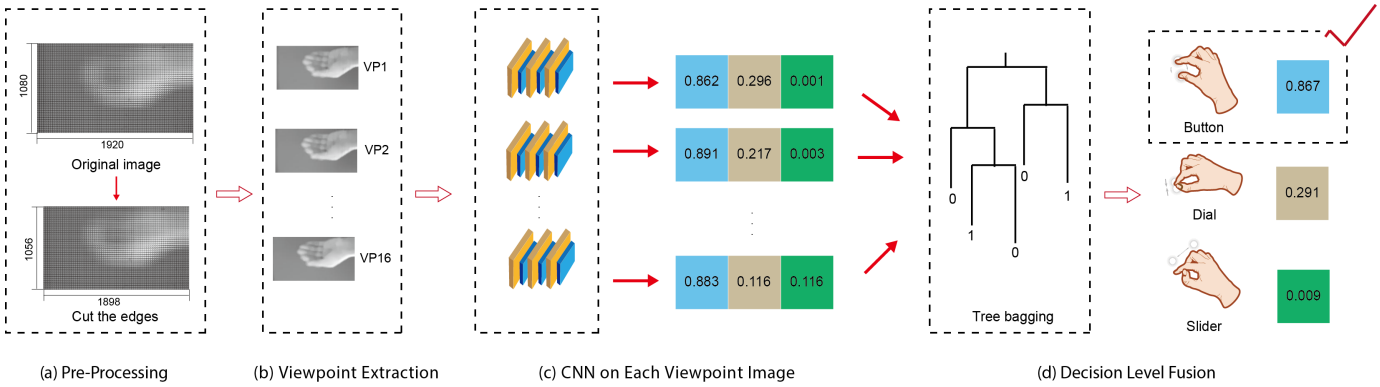


Fig. 2. Block diagram of the proposed microgesture recognition system. There are four stages: (a) pre-processing; (b) viewpoint image extraction; (c) deep learning for prediction on viewpoint images; (d) decision level fusion.

video data. It is found that the obtained 3D microgesture database has high-quality microgestures and true 3D advantages over traditional 3D capture devices. Although there are image and video subsets in the HoMG dataset, we only focus on the image subset, as the data processing methods will be quite different for video subsets. Dynamic information extraction is the key to video-based microgesture recognition systems.

For image-based microgesture recognition, efforts have been made recently [9] [10] [13] [12] [11]. Traditional 2D image feature extraction and classification methods were used in the baseline paper [9]. Zhang et al. [10] proposed a method for 2D microgesture images using CNN models with fine tuning. The method achieved the best accuracy by averaging the probabilities predicted from different models and different epochs. Lei et al. [12] proposed a bidirectional morphological filter and a fast-fuzzy C-means clustering (FCM) method [20] to reconstruct 2D images from an H3D image. This method is effective for solving the problem of blur and distortion grids in H3D images. The method ranked in second place in the challenge competition in the image subset. The main limitation of this method is that the reconstructed image has a lower resolution and loses some detail information. Sharama et al. [13] considered each microlens capturing the image at its respective angle, in contrast to the other lenses. They extracted a viewpoint image by selecting a pixel from each microlens and used a feature fusion technique on both handcrafted and deep features extracted from the neural network. The experiments showed that their proposed method outperforms the baseline by an absolute margin of 26.67%. Peng et al. [11] proposed a deep residual network with an attention mechanism. The experiments showed that the attention design can highlight the microgesture part and reduce the noise introduced from the wrist and background. An accuracy of 82.1% on the image subset was achieved.

From the methods of all the participants in the challenge on this dataset, it can be seen that there are three potential issues that may help to improve the performance of the microgesture recognition. First, because the original H3D images contain considerable noise, the appropriate image pre-processing method is crucial for extracting correct 3D information from H3D images. Although Lei et al. [12] and Garima et al. [13] have attempted to address this issue, noise, such as dark borders and geometric distortions, has not been fully removed. More importantly, none of the participants took advantage of 3D information for microgesture recognition. Second, most participants accomplished the task of microgesture recognition by using deep learning methods and obtained significantly improved results compared with the baseline method. However, most of the deep learning models were applied to the H3D images directly or to 2D images with limited view angles. The power of the deep learning models was not fully utilized. Third, Lei et al. [12] and Zhang et al. [10] used decision fusion methods to improve the recognition accuracy. This is an effective technique and should be explored further.

In what follows, we will address all these issues one by one and attempt to develop a better system for microgesture recognition.

## III. Methodology

This section introduces the detailed information about our solutions for each issue identified in existing methods and then proposes our solution. We present our method below in detail.

### A. Overview

Fig. 2 shows the proposed framework, which includes four main stages. First, in the pre-processing stage, the original H3D images are cropped along the four boundaries to localize the element images (EIs). An EI is a local area in the H3D image that was captured by one of the microlens arrays. The captured H3D images might have various offsets depending on the positions of the microlens array inside of the camera. This is a preparation step for ViewPoint (VP) image extraction, where the VP image is a 2D image of the scene from a particular viewing angle. Second, multiple shifted VP images are extracted from one H3D image, where each VP image has different shifts from horizontal and vertical positions at the angle of view. Three simple and efficient patch-based rendering approaches were proposed by Georgiev and Lumsdaine [21], and were used by Yang et al. [22] in a holoscopic image coding scheme. However, our proposed method here is simpler, and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/THMS.2020.3047914,


the EIs can be cropped out automatically. Third, the CNN model with an attention block is used to extract features from each VP image and generates predictions (e.g., the probability that it belongs to each type of microgesture from its fully connected layer). Finally, the decision fusion methods are used to combine the predicted decision values from each VP image and produce the final prediction of the type of microgesture.

### B. Pre-processing

In the recording process of the original H3D image, an object is captured through an array of microlenses, where each microlens captures a perspective 2D element image of the object from a specific angle. The final captured image contains the intensity and directional information of the corresponding 3D scene in 2D form. This 2D element image is called the EI, which is a small grid area in the H3D image. The standard pre-processing process can be found in [23], which includes lens correction, distortion correction, EI extraction, viewpoint extraction, etc. Most processing needs to be done manually. The first pre-processing step is to create an automated method to detect the edges of EIs and crop out the EIs from the original H3D image. Fig. 3 shows an example of a holoscopic 3D

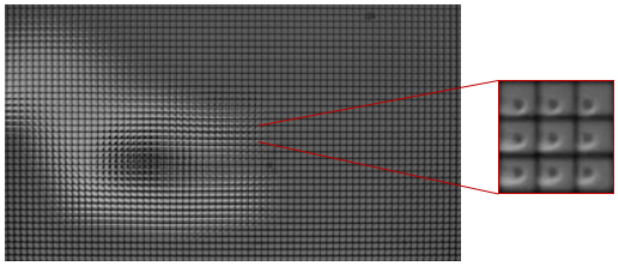

Fig. 3. H3D micro-gesture images consist of multiple 2D element images (EIs). Here, 9 EIs are enlarged from the original H3D image.

microgesture image that consists of many 2D EIs. Roughly, each EI is an approximately square area with small values (dark areas) at the edge. However, some boundaries are not straight lines due to the distortion of the microlens, especially those near the H3D image boundary associated with the microlens far from the centre. Barrel distortion is caused by spatial imaging in a narrow space, and results in obvious distortion in the corner and edges. Although this distortion is not easy to notice by the human eye, it greatly affects the extraction process [7].

In this work, all the EIs are cropped out based on straight lines, and the distortion will be addressed later by the algorithm for VP image extraction. On the boundaries of the H3D image, some EIs are not fully captured, so only completed EIs will be cropped out and used later for VP image extraction.

The cropping algorithm is based on the detection of the minimum values of the rows and columns of an H3D image. For an H3D grayscale image $H(i,j), i = 1,2,\cdots,I_o, j = 1,2,\cdots,J_o$ with a resolution of $I_o \times J_o$, all the values are summarized to $H_c$ and $H_r$ according to column and row as shown in equations 1 and 2, respectively. Then, the local

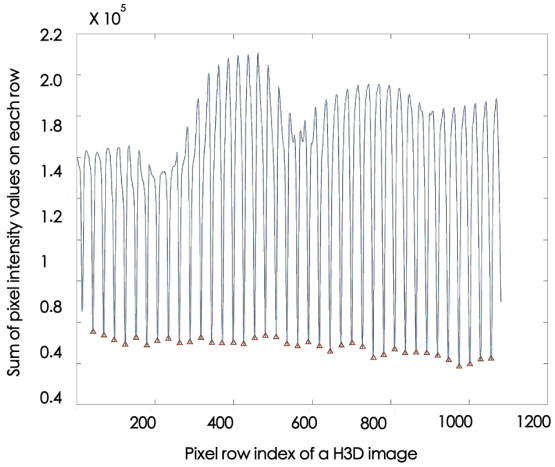


Fig. 4. The minimal values of the summarized rows of a H3D image. The 38 points marked with small red triangles are the selected boundaries of the EIs.

minima of the functions $H_c$ and $H_r$ are are selected as the boundaries of the EIs. For example, if the H3D image has a resolution of 1920×1080, we can obtain two vectors, as shown in Fig. 4 and Fig. 5. In Fig. 4, there are 40 local minima detected where the left and right minimum points were removed as they are the edge of an incomplete EI. Therefore, 38 EI edges were finally chosen. In the same manner, in Fig. 5, there are 70 local minima detected where the left and right minimum points were removed, Thus, EI edges were finally chosen. In the end, 68 × 38 EIs were cropped out from one H3D image. This method is a fast algorithm that can quickly produce all the EI images.

$$H_c(i) = \sum_{j=1}^{J_o} H(i,j), i = 1,2,\cdots,I_o \tag{1}$$

$$H_r(j) = \sum_{i=1}^{I_o} H(i,j), j = 1,2,\cdots,J_o \tag{2}$$

In practice, camera calibration is not executed perfectly because the microlens cannot be placed perfectly on the horizontal and vertical lines of the H3D image. There can be a small angle $\delta$ between the boundary of the EIs and the horizontal and vertical lines of the H3D image, as shown in Fig. 6. A small shift image $H_\delta$ of the original H3D image can be used in the above methods, and then, the best $\hat{\delta}$ can be obtained by minimizing the local minima of the summary of row and column pixels, as shown in Eq. 3:

$$\hat{\delta} = \arg\min_{\delta}(\sum_{i=1}^{I_o} H_c^{\delta}(i) + \sum_{j=1}^{J_o} H_r^{\delta}(j)) \tag{3}$$

In this way, the best cropping of the H3D image $I_c$ is obtained with $\hat{\delta}$.

### C. Holoscopic 3D Viewpoint Extraction and Shift

From all the obtained EIs, VP images can be extracted. The VP image is a clear image (e.g., without the grid boundaries

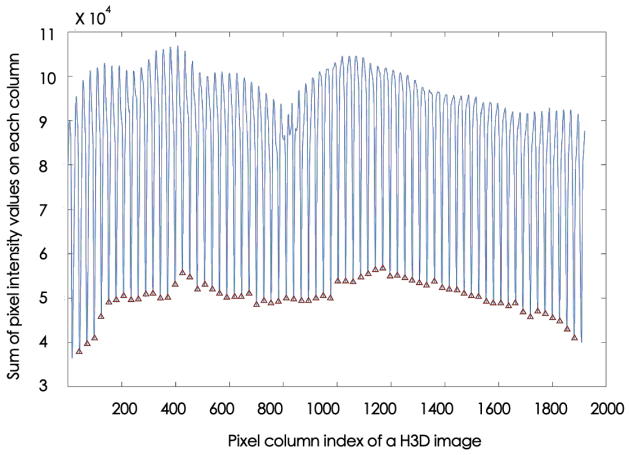


Fig. 5. The minimal values of the summarized columns of a H3D image. The 68 points marked with small red triangles are the selected boundaries of the EIs.

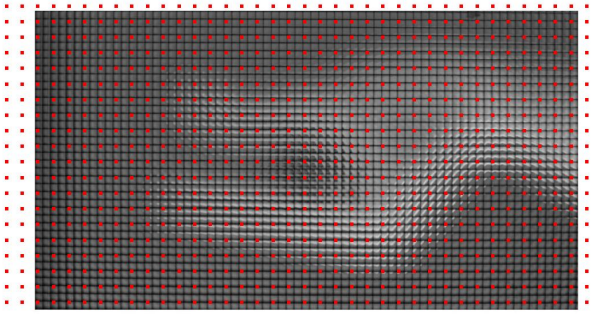

Fig. 6. The horizontal and vertical lines of H3D images are adjusted for the EI extraction.

of the EIs), and it can be extracted from the pixels of all the EIs. The basic principle of the recording process is to map the object to an image through a microlens array, where each microlens has intensity and directional information from the specific capture angle. Fig. 7 shows the relationship between 5 EIs and 3 focus layers. The captured H3D image includes a microlens array of 5 lenses, with 3 pixels per lens. In this particular example, there are 3 planes per slice, with an image size of 5 pixels, which is the VP image. In the recording stage, each microlens of local pixel position is involved direction as shown in Fig 7(a). For each VP image, the reconstruction integrates all the pixels from the same location under different microlenses. All the VP images, such as VP1, VP2 and VP3, are orthographic images as shown in Fig. 7 (b), and they are reconstructed by all pixels from the same location in the 5 EIs. It should be mentioned that the focus plane of each VP image might be different, as shown in Fig. 7 (b). Holoscopic 3D imaging changes the focus plane on which all the light rays converge to the ideal virtual depth plane. However, the viewpoint rendering pixels are used to refocus at different depth planes.

In holoscopic 3D imaging, each EI, which is captured by a microlens, contains a pixel from each layer of the 3D scene. In the same way, all EIs contribute to creating a single-aperture holoscopic 3D scene in the space.

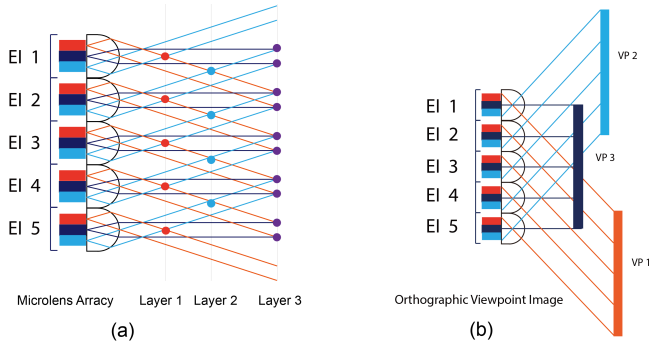


Fig. 7. The relationship between the EIs and focus layers of the holoscopic 3D capturing system. (a) microlens array recording system, (b) orthographic viewpoint images from different perspectives.

The H3D viewpoint extraction process selects appropriate pixels at the same location from every EI of the H3D image to reconstruct an orthographic viewpoint image. The principle of the proposed viewpoint extraction method is illustrated in Fig. 8, where there are 3×3 pixels in each EI, and the 9 EIs constitute an omni-directional H3D image.

In general, for a well-cropped H3D image $I_c$ with $n \times m$ EIs, each EI can be represented as EI $(p, q)$ where $p = 1$ to $P$ and $q = 1$ to $Q$. The VP image $VP(p, q)$ will have dimensions of $n \times m$, as there is one pixel extracted from each EI. The values of $P$ and $Q$ are decided by the resolution of the cropped H3D image $I_c$ because $I_c$ resolution will be $(m \times P, n \times Q)$.

The equation for the VP extraction can be written as follows:

$$VP_{p,q}(i, j) = I_c((i-1)P + p, (j-1)Q + q) \qquad (4)$$

where $i = 1, \cdots, m$ and $j = 1, \cdots, n$ are the coordinates of the VP image and $p$ and $q$ are the index of the horizontal and vertical positions of the VP images, respectively. The VP image $VP_{p,q}(i, j)$ has a resolution of $n \times m$ pixels.

In principle, $P \times Q$ VP images can be extracted from one H3D image, where every pixel in all the EIs can be picked up. However, this does not work in practice due to several issues. First, there is a very small difference between two adjacent pixels in one EI, and substantial additional information will not be provided by picking up all the VP images. Second, the intensity value of one pixel might vary due to lighting conditions and random noise. This will reduce the quality of the VP images. Third, there are barrel distortion effects at the boundaries of each EI when the distances between the object and the microlens are large.

In our proposed method, we address all these issues to obtain high-quality VP images. First, we only extract a small number of VP images that have large differences between each other. Second, our VP images are extracted from patches instead of individual pixels from each EI. Third, only the central area in one EI is selected and used for VP image extraction to avoid distortions. The patches are shifted in the horizontal and vertical directions, and only a small number of viewpoint images are extracted.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/THMS.2020.3047914,



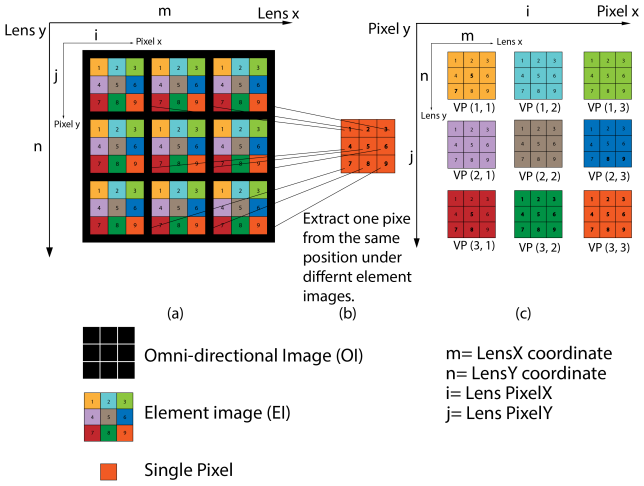


Fig. 8. Illustration of the principle of H3D image viewpoint image extraction. (a) 3×3 pixels under each microlens, (b) one VP image extracted from the same position under different microlenses, (c) nine VP images extracted from 3×3 EIs.

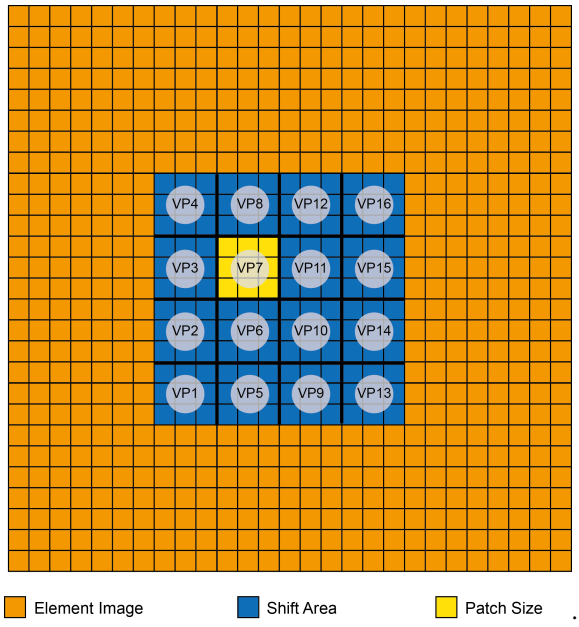


Fig. 9. This is one EI image with a resolution of 27×27. The boundary pixels are avoided, and only the central pixels are selected. Each patch has 3×3 pixels. In the end, only 16 patch areas are selected from this EI.

Fig. 9 shows one EI where the boundary pixels are not used and only the central pixels are selected for VP image extraction. The central pixels are composed of 16 patches, each of which has 3×3 pixels. All the 3×3 pixels contribute to one pixel in the VP image. From each patch, one VP image is reconstructed, and a total of 16 viewpoint images are extracted.

### *D. Convolutional neural network*

A convolutional neural network (CNN) is a biologically inspired model and is very successful in image-related recognition tasks [24]. An important component of the CNN is the shared weight and subsampling. Generally, a deep convolutional neural network is formed by stacking multiple convolution layers (conv) and subsampling layers [25]. Fig. 10 shows the whole structure of a specific CNN model (ResNet-50) with attention-based residual blocks embedded. The network receives images with the same size. After processed by a convolution kernel, each small neighborhood in the input layer will form a value in a feature map (each plane in the layer). The $i^{th}$ feature map $C^i$ can be expressed as:

$$C^i = f(x * W^i + b^i) \tag{5}$$

where $f$ is the activation function, $x$ is the input VP image, and $W$ and $b$ are the weight of the convolution kernel and bias, respectively. Each feature map shares the same $W$ and $b$. In a convolution layer, there is normally more than one convolution kernel; thus, multiple feature maps are calculated. The $i^{th}$ feature map $P^i$ in the pooling layer can be calculated by using

$$P^i = f(\beta * S(C^i)) + \alpha) \tag{6}$$

$\beta$ and $\alpha$ are the coefficient and bias, respectively. $S(\cdot)$ denotes the subsampling operation for a convolutional feature map. It can be written as:

$$S(C^i) = maxC^i_{s,l} \quad \|s\| \leq \frac{N_s}{2}, |l| \leq \frac{N_s}{2}, s, l \in Z^+ \tag{7}$$

where $N_s$ is the subsampling size.

The fully connected layer (Fc) is a multilayer perception feed-forward neural network, and the output layer can be written as:

$$p(j|F;\theta) = \frac{e^{\theta_j^T F}}{\sum_{i=1}^{J} e^{\theta_i^T F}} \qquad 1 \leq j \leq J \tag{8}$$

where $p(j|F;\theta)$ denotes the probability that the input feature $F$ belongs to class $j$, $\theta$ is the weight vector between the output layer and the previous layer, and $J$ is the number of classes.

Many CNN structures, such as GoogLeNet [26] and ResNet [27], have been trained on ImageNet [24] and have achieved superior performance. Here, a pretrained ResNet model is used, and fine tuning is carried out on the model with our dataset. In addition, we modified the existing CNN model by adding an attention-based residual block.

Fig. 11 shows the attention-based residual block, where the dotted-line area is the attention branch, which can focus on the finger microgesture and reduce the noise introduced by the wrist and background. For input $x$, the overall output is $O(x)$.

$$O(x) = F(x) + F(x) \cdot A(x) + x \tag{9}$$

$A(x)$ represents the spatial attention mask. This attention branch here has been used in our previous work [11]. It is a bottom-up, top-down structure used to learn the interesting area of a gesture image, as shown in Fig. 12. From Fig. 12, it can be seen that the attention design places more attention to the gesture area in higher level layers. We believe that this is special for microgesture recognition.

The CNN model adopted in this attention design is applied for all VP images for microgesture recognition. The output probabilities of the CNN were produced. From each sample,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/THMS.2020.3047914,


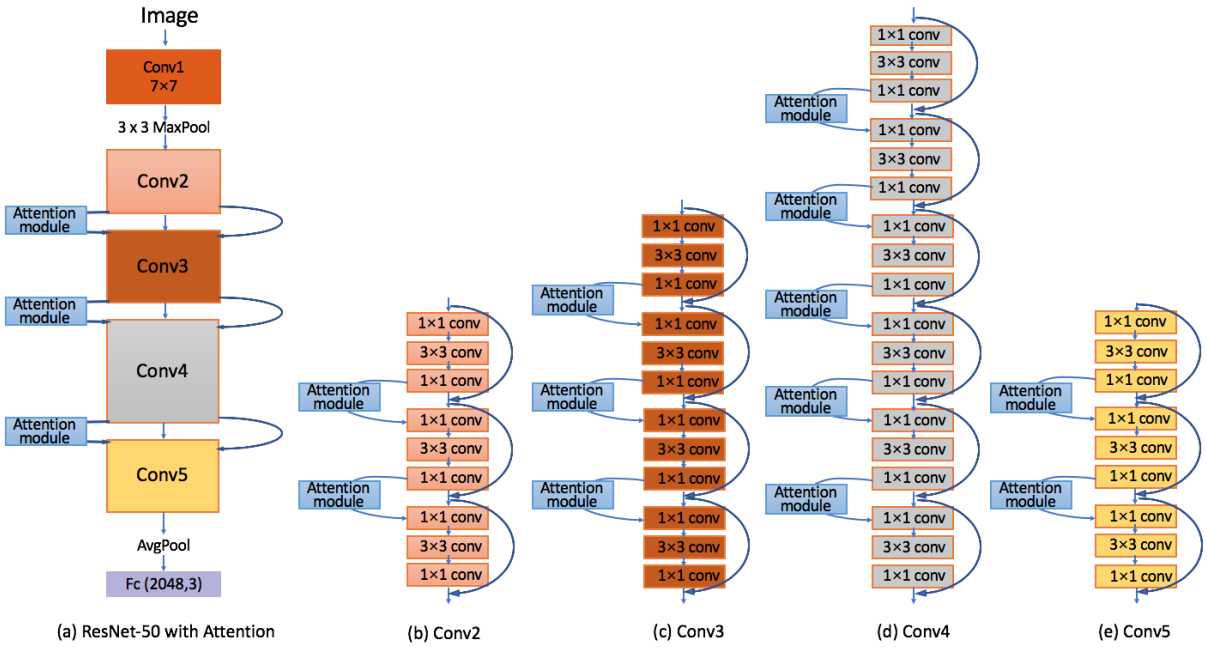


Fig. 10. The whole structure of ResNet-50 CNN model with attention-based residual blocks embedded. (a) overall architecture with main blocks; (b) Conv2 layers; (c) Conv3 layers; (d) Conv4 layers; (e) Conv5 layers.

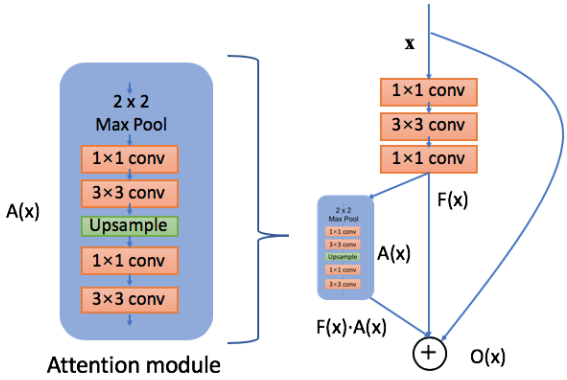


Fig. 11. Detailed architecture of attention-based residual block that was integrated into the CNN architecture.

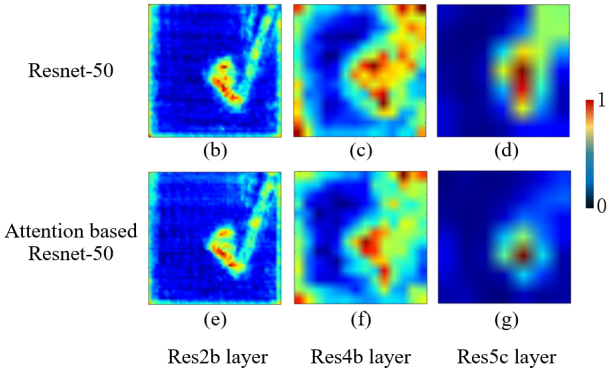


Fig. 12. The first row and second row represent the feature maps learned by ResNet-50 with and without attention-based residual block respectively at the Res2b layer (low-level), Res4b layer (middle-level) and Res5c layer (high level).

the probabilities of three gestures are computed and used for decision-level fusion.

### *E. Decision Fusion*

Decision fusion, also referred to as a mixture of experts [28], is a method that can be used to improve the recognition rate by combining all the decisions together. In this work, some ensemble functions based on voting [29] and trainable methods [30] have been explored for combining predictions from multiple VP images efficiently. Specifically, some simple fusion methods such as the bagging learning strategy with REPTree, are used in the multiple-viewpoint predictions.

Assume that there are $J$ classes for all the H3D images and that each H3D image has $K$ VP images, The CNN models will be applied to all VP images separately to produce all the prediction probabilities $\{p_{j,k}\}$, where $\{j = 1, 2, \cdots, J\}$ is the index of the class and $\{k = 1, 2, \cdots, K\}$ is the index of the VP images.

**Majority Voting.** For instance, voting strategies assume that each classifier gives a prediction with the probabilities belong to each class. Therefore, the predicted label $V_k$ from a single classifier can be represented as follows:

$$V_k = \arg\max_j(p_{j,k}), \quad k = 1, 2, \cdots, K \tag{10}$$

The predicted label $V_M$ of the majority voting can be written as:

$$V_M = \arg\max_j(\sum_{k=1}^{K} I_{\{V_k=j\}}) \tag{11}$$

where the index function $I_\Delta$ is 1 if the set $\Delta$ is non-empty and 0 otherwise.

**Averaging.** The average fusion method can be applied to the multiple classifiers under the condition that each output of the classifiers is expressed as probabilities. The decision output $V_A$ for the averaging fusion can be written as

$$V_A = \arg\max_j (\frac{1}{K} \sum_{k=1}^{K} p_{j,k}) \tag{12}$$

**Product.** The product probability fusion method calculates the product of experts by multiplying individual probabilities. Similar to the averaging probability fusion method, the product probability fusion output $V_P$ can be written as

$$V_P = \arg\max_j (\prod_{k=1}^{K} p_{j,k}) \tag{13}$$

**Bagging Classification Tree.** Trainable mixtures of experts have the ability to learn from individual classifier outputs to form a higher level of expertise. In this work, bagging learning with REPTree has been explored to enhance the multi-viewpoint results. The bagging learning strategy was introduced by Breiman [31] to reduce the variance of a predictor. It is a successful method for improving classification performance. The reduce error pruning tree (REPTree) [32] method is a fast decision tree learning method that is based on the information gain. The main steps of the trainable mixture of experts approach are as follows.

Assume that we have $N$ instances. For each instance, the numbers of VP images and classes are $K$ and $J$, respectively. Therefore, the feature dimension of each instance is $K \times J$. First, a training set is sampled (with replacement) from all instances to generate a classifier. Specifically, REPTree algorithms are used as the learning system. Then, as in the first step, the number of trials $T$ is replicated to form the $T$ classifiers. Finally, for an instance, the classification result is voted on by every classifier for the class with the most votes.

## IV. Experiments and Evaluation

### A. HoMG database

The holoscopic microgesture (HoMG) database was recorded for this research and is publicly available at our website (http://3dvie.co.uk/). For the data collection, 40 participants were selected, and the recordings were conducted under 2 different backgrounds, with 2 hands (e.g., left and right), 2 distances (e.g., far and close) and 3 microgestures (e.g., Button, Dial and Slide). Therefore, 24 videos were recorded for each participant. The length of a video is between 2 and 20 seconds, with a frame rate of 25 fps, and resolution of $1920 \times 1080$. In total, 960 videos are included in the database.

The HoMG database has been made publicly available [9] for microgesture recognition competition (http://3dvie.co.uk/), and was divided into two subsets: image-based and video-based microgesture subsets. Additionally, it was divided into training, development and testing subsets. Detailed information about the HoMG database is shown in Table I. In this paper, the work is only done for the image-based subset, where each microgesture is represented by an H3D image.

TABLE I
NUMBER OF SAMPLES IN EACH PARTITION OF THE HOMG DATABASE. "B" STANDS FOR BUTTON, D STANDS FOR DIAL AND S STANDS FOR SLIDE.

| Subset | Training | | | Development | | | Testing | | |
|---|---|---|---|---|---|---|---|---|---|
| | B | D | S | B | D | S | B | D | S |
| Image | 5507 | 5534 | 5722 | 2266 | 2188 | 2106 | 2665 | 2267 | 2359 |
| Video | 160 | 160 | 160 | 80 | 80 | 80 | 80 | 80 | 80 |

### B. VP Extraction Parameters

For an original H3D image, its resolution is 1920×1080. The resolution of the element image from each microlens is approximately 27×27. However, at the edge of the H3D image, there are some EIs that cannot have full resolution due to the completion of the microlens. Therefore, these pixels of the H3D image were cropped out. Specifically, 68×38 full EIs were cropped out from one H3D image after rotation of the image and after making straight cropping lines.

After edge cropping, the H3D image should be estimated in depth and refocused to extract the VP images. A small patch area of 3×3 was chosen from the central area of each EI and then shifted in the horizontal and vertical directions. The shift value also leads to the depth transformation. We obtained 16 points by extracting the viewpoint from the different refocusing layers, as shown in Fig. 9. In the human eye, slight movement from the viewing of continuous VP images can be observed. In the end, for each H3D microgesture image, 16 2D VP images were extracted from different depths, as shown in Fig. 13.

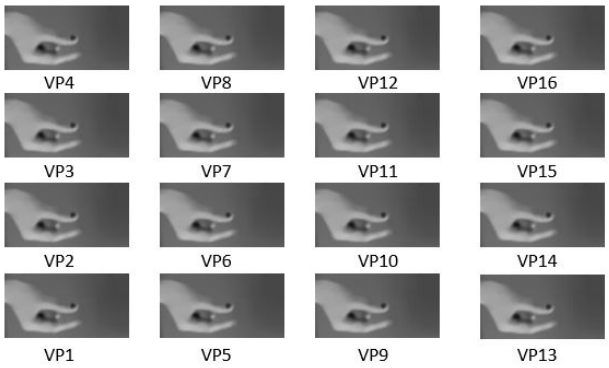


Fig. 13. The 16 viewpoint images extract from an H3D image (e.g. the image in Fig. 3).

### C. Implementation Details of the CNN Model and Fusion

Considering that the microgesture is a small-local representation in an image. We adopted the attention mechanism to highlight the finger microgesture area.

Before training, we used the transfer learning strategy to initialize the network with the ImageNet database and obtain a pretrained model [27]. In the stage of training for microgesture recognition, fine tuning was performed for the pre-trained model on the HoMG database. The average values of all pixels of the training set have been subtracted from the input

TABLE II
CLASSIFICATION ACCURACY(%) OF CNN MODELS ON EACH VP IMAGES ON THE TESTING SET OF THE HOMG DATABASE.

| No.VP | Acc | No.VP | Acc | No.VP | Acc | No.VP | Acc |
|---|---|---|---|---|---|---|---|
| **VP1** | 86.50 | **VP5** | 86.69 | **VP9** | **86.81** | **VP13** | 86.79 |
| **VP2** | 86.47 | **VP6** | 86.76 | **VP10** | 86.61 | **VP14** | 86.62 |
| **VP3** | 85.68 | **VP7** | 86.33 | **VP11** | 86.28 | **VP15** | 86.27 |
| **VP4** | 85.74 | **VP8** | 86.01 | **VP12** | 86.1 | **VP16** | 86.42 |

grayscale image, and the input image was further divided by the variance of all pixels of the training set. This is a normalization process. To increase the robustness of the network, each VP image was resized to 256×256 and then cropped out of the four corners to a size of 224×224. Moreover, data augmentation, such as color shift (maximum value of 20) and image rotation (maximum degree of 10), is applied to the training set with a probability of 0.5. The dropout ratio of the last weight layer was set to 0.5. The batch size was set to 9 with a momentum of 0.9 and weight decay of 0.0005. The initial learning rate was set to 0.001, decreasing 10 times every 10 epochs. Our training process was implemented on the Caffe framework [33] with an Nvidia Titan X GPU.

After the training, each VP image in the dataset was input to the trained model to obtain a decision output of the classification layer. Specifically, each value in the output vector represents the probability belonging to the three types of microgestures. Because each instance in the dataset has 16 VP images, we obtained 16×3 output values as the probabilities of the instance belonging to each type of microgesture based on each VP image.

In our mixture of experts procedure, we set the number of bagging trails as 10000, and in each trail, 50% of the instances in the training set were sampled to generate a classifier. For REPTree training [32], the minimum total weight of the instances in a leaf is set to 2, and the amount of data used for pruning is set to 3.

### *D. Experimental Results and Comparison*

Table II shows the experimental results using CNN models on separate VP images. In total, 16 VP images were extracted from one H3D image. Each of them has been applied in the proposed CNN models and the associated classification probabilities have been produced. This accuracy is the percentage of correctly classified microgestures in the testing set after the models were trained on the training and development subsets. From this table, it can be seen that similar performance has been achieved for each individual VP image. The best result of 86.81% was achieved for viewpoint 9, which is a very high accuracy.

Table III shows the results achieved by our proposed method in comparison with other state-of-the-art methods. First, we combined all 16 VP images and applied the CNN model and achieved an accuracy of 86.71%. The last four methods of the table denote the methods combining the CNN outputs based on a mixture of experts approach. We can see from Table III that our proposed pre-processing methods used for H3D images combining the CNN with a mixture of experts approach obtained a significant performance improvement on microgesture recognition. Specifically, the proposed methods, which combine the CNN with the bagging classification tree approach, achieved an improvement of approximately 40% in recognition accuracy compared to the baseline method. The method also outperforms the method of Zhang et al. [10] (87.15% vs 86.70%). In addition, compared to the method of Peng et al. [11], the CNN model achieved an accuracy improvement of approximately 5%. It was even slightly better than the method by Zhang et al. [10], although the voting method was used in their work (86.71% vs 86.70%). Consequently, the proposed pre-processing methods used for H3D images are validated to be effective. Moreover, the last four methods in the table show that the mixture of experts approach makes a great contribution to the improved recognition rate. Notably, the proposed trainable mixture of experts based on the bagging classification tree is superior to the voting and the probability fusion method.

TABLE III
CLASSIFICATION ACCURACY (%) COMPARISON BETWEEN THE PROPOSED METHOD AND ALL THE EXISTING METHODS ON THE TESTING SUBSET OF THE HOMG DATABASE. "A" MEANS ATTENTION BLOCK. "M.V." MEANS "MAJORITY VOTING".

| Author | Methods | Accuracy |
|---|---|---|
| Liu et al. [9] | LBP+k-NN | 50.90 |
| Liu et al. [9] | LPQ+SVM | 52.6 |
| Sharma et al. [13] | CNN+LPQ(Max Vote) | 77.57 |
| Peng et al. [11] | A-Resnet | 82.10 |
| Lei et al. [12] | FCM+GoogLeNet | 84.28 |
| Zhang et al. [10] | ResNet152+DenseNet161+SeResNet50+M.V. | 86.70 |
| This work | 16VPs+A-ResNet | **86.71** |
| This work | 16VPs+A-Resnet+M.V. | **87.04** |
| This work | 16VPs+A-Resnet+(Mean Probability Fusion) | **87.04** |
| This work | 16VPs+A-Resnet+(Product Probability Fusion) | **87.03** |
| This work | 16VPs+A-Resnet+(Bagging Classification Tree) | **87.15** |

## V. CONCLUSION

Image-based finger microgesture recognition is a difficult challenge in unconstrained environments. In this paper, we proposed innovative 3D microgesture recognition methods based on a holoscopic 3D imaging system that outperformed all state-of-the-art methods. A comprehensive holoscopic 3D database is produced, particularly for 3D microgestures, and made publicly available. The proposed method includes a fast and robust pre-processing method, designed and developed for H3D images for data preparation by automatically extracting the element images as well as VP images in a simpler manner. This innovative pre-processing approach can clean and prepare visual data to achieve effective learning and detection.

A pretrained CNN model with attention mechanics is applied to each VP image to obtain the predicted probabilities of each gesture. Finally, some mixture of expert methods based on voting strategies and trainable models have been explored to achieve better classification results. The achieved recognition accuracy outperformed all state-of-the-art methods. The accuracy of 87% might be good for some applications already. The main reason is that the attention-based network can learn to focus on the area of interest for each viewpoint image, and

the decision fusion method efficiently ensembles the classification results of each viewpoint. This also demonstrated that the holoscopic 3D imaging system provides a new dimension for 3D microgesture recognition, as it captures colour, texture and motion of the real-world scene at full resolution.

## ACKNOWLEDGMENT

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPU used for this research, as well as the support of Aidrivers for sponsoring this research paper.

## REFERENCES

[1] M. G. Helander, T. K. Landauer, and P. V. Prabhu, Eds., *Handbook of Human-Computer Interaction*, 2nd ed. New York, NY, USA: Elsevier Science Inc., 1997.

[2] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artif. Intel. Review*, vol. 43, no. 1, pp. 1–54, 2012.

[3] M. Studdert-Kennedy, "Hand and Mind: What Gestures Reveal About Thought," *Lan. and Spe.*, vol. 37, no. 2, pp. 203–209, 1994.

[4] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE Multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

[5] L. E. Potter, J. Araullo, and L. Carter, "The leap motion controller: A view on sign language," in *in Proc. 25th Aus. Comput. Human Inter. Conf.:Augme. Appli. Innove. Collab.*, Sep. 2016, pp. 175-178.

[6] F. Garcia, D. Aouada, T. Solignac, B. Mirbach, and B. Ottersten, "Real-time depth enhancement by fusion for RGB-D cameras," *IET Computer Vision*, vol. 7, no. 5, pp. 1–11, October 2013.

[7] M. R. Swash, A. Aggoun, O. Abdulfatah, B. Li, J. C. Fernandez, and E. Tsekleves, "Holoscopic 3D image rendering for autostereoscopic multiview 3D Display," *in Proc. IEEE Int. Conf. Multi. Syst. and Broad.*, May. 2013, pp. 1-4.

[8] M. R. Swash, O. Abdulfatah, E. Alazawi, T. Kalganova, and J. Cosmas, "Adopting multiview pixel mapping for enhancing quality of holoscopic 3D scene in parallax barriers based holoscopic 3D displays," *in Proc. IEEE Int. Conf. Multi. Syst. and Broad.*, April. 2014, pp. 1-4.

[9] Y. Liu, H. Meng, M. R. Swash, Y. F. A. Gaus, and R. Qin, "Holoscopic 3D microgesture database for wearable device interaction," *in Proc. IEEE 13th Int. Conf. Auto. Face Gest. Recog., May. 2018, pp. 802-807.*

[10] W. Zhang, W. Zhang, and J. Shao, "Classification of holoscopic 3D microgesture images and videos," *in Proc. IEEE 13th Int. Conf. Auto. Face Gest. Recog.*, May. 2018, pp. 815–818.

[11] M. Peng, C. Wang, and T. Chen, "Attention based residual network for microgesture recognition," *in Proc. IEEE 13th Int. Conf. Auto. Face Gest. Recog.*, May. 2018, pp. 790-794.

[12] T. Lei, X. Jia, Y. Zhang, Y. Zhang, X. Su, and S. Liu, "Holoscopic 3D microgesture recognition based on fast preprocessing and deep learning techniques," *in Proc. IEEE 13th Int. Conf. Auto. Face Gest. Recog.*, May. 2018, pp. 795-801.

[13] G. Sharma, S. Jyoti, and A. Dhall, "Hybrid neural networks based approach for holoscopic microgesture recognition in images and videos," *in Proc. IEEE 13th Int. Conf. Auto. Face Gest. Recog.*, May. 2018, pp. 808-814.

[14] H. Cheng, L. Yang, and Z. Liu, "A survey on 3D hand gesture recognition," *IEEE Trans. Circu. A Syst for Video Tech.*, vol. 26, no. 9, pp. 1659-1673, Sept. 2016.

[15] V. Frati, "Using Kinect for hand tracking and rendering in wearable haptics," *in Proc. IEEE World Hap. Conf.*, Jul. 2011, pp. 317-321.

[16] Y. Ming, "Hand fine-motion recognition based on 3D Mesh MoSIFT feature descriptor," *Neurocomputing*, vol. 151, no. 2, pp. 574–582, 2015.

[17] P. Narayana, J. R. Beveridge, and B. A. Draper. "Gesture recognition: Focus on the hands," *in Proc. IEEE 28th Int. Conf. Comput. Vision and Patte. Recogn.*, Jun. 2018, pp. 5235-5244.

[18] X.S. Nguyen, L. Brun, O. Lzoray and S. Bougleux. "A neural network based on SPD manifold learning for skeleton-based hand gesture recognition," *in Proc. IEEE 28th Int. Conf. Comput. Vision and Patte. Recogn.*, Jun. 2019, pp. 12036-12045.

[19] M. Abavisani, H. R. V. Joze, and V. M. Patel. "Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training," *in Proc. IEEE 28th Int. Conf. Comput. Vision and Patte. Recogn.*, Jun. 2019, pp. 1165-1174.

[20] T. Lei, X. Jia, Y. Zhang, L. He, H. Meng, and A. K. Nandi, "Significantly fast and robust fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 5, pp. 3027-3041, Oct. 2018.

[21] A. L. Todor G. Georgiev, "Focused plenoptic camera and rendering," *J. Elec. Imag.*, vol.19, no.2, pp. 1-11, 2010.

[22] L. Yang, P. An, D. Liu, R. Ma, and L. Shen, "Three-dimensional holoscopic image-coding scheme using a sparse viewpoint image array and disparities,"*J. Elec. Imag.*,vol. 27, no. 033030, pp. 1-15, 2018.

[23] A. Aggoun, E. Tsekleves, M. R. Swash, D. Zarpalas, A. Dimou, P. Daras, P. Nunes, and L. D. Soares, Immersive 3D holoscopic video system, *IEEE MultiMedia* , vol. 20, no. 1, pp. 2837, Jan 2013.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advan. Neural Info. Pro. Syst.*, pp. 1–9, 2012.

[25] C. Chung, S. Patel, R. Lee, L. Fu, S. Reilly, T. Ho, J. Lionetti, M. D. George, and P. Taylor, "Implementation of an integrated computerized prescriber order-entry system for chemotherapy in a multisite safety-net health system," *American J. Health syst pharm.*, vol. 75, no. 6, pp. 398–406, 2018.

[26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *in Proc. IEEE 28th Int. Conf. Comput. Vision and Patte. Recogn.*, Jun. 2015, pp. 1-9.

[27] O. Kuchaiev and B. Ginsburg, "Factorization tricks for LSTM networks," 2017. [Online]. Available: http://arxiv.org/abs/1703.10722

[28] S. E. Yuksel, J. N. Wilson, and P. D. Gader, "Twenty years of mixture of experts," *IEEE Trans. Neural Net. Learn. Syst.*, vol. 23, no. 8, pp. 1177–1193, 2012.

[29] A. Sinha, H. Chen, D. G. Danu, T. Kirubarajan, and M. Farooq, "Estimation and decision fusion: a survey," *Neuro.*, vol. 71, no. 13-15, pp. 2650-2656, 2008.

[30] M. Gashler, C. Giraud-Carrier, and T. Martinez, "Decision tree ensemble: small heterogeneous is better than large homogeneous," *in Proc. 7th Int. Conf. on Machine Learn. and Appli.*, Dec. 2018, pp. 900-905.

[31] L. Breiman, "Bagging predictors - Springer," *Machine learning*, vol. 140, pp. 123-140, 1996.

[32] S. Kalmegh, "Analysis of WEKA data mining algorithm REPTree , simple cart and RandomTree for classification of Indian news," *Int. J. Innov. Sci. Eng. Tech.*, vol. 2, no. 2, pp. 438-446, 2015.

[33] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe," *in Proc. 22st ACM Int. Conf. Multimedia*, 2014, pp. 675-678.