DOI: 10.7860/JCDR/2020/45065.13961

Original Article

Others Section

Dynamic Gesture Controlled User Interface Expert HCI System using Adaptative Background Masking: An Aid to Prevent Cross Infections

SEEMA RAWAT[1], PRAVEEN KUMAR[2], ISHITA SINGH[3], SHOURYA BANERJEE[4], SHABANA UROOJ[5], FADWA ALROWAIS[6]

ABSTRACT

Human-Computer Interaction (HCI) interfaces need unambiguous instructions in the form of mouse clicks or keyboard taps from the user and thus gets complex. To simplify this monotonous task, a real-time hand gesture recognition method using computer vision, image, and video processing techniques has been proposed. Controlling infections has turned out to be the major concern of the healthcare environment. Several input devices such as keyboards, mouse, touch screens can be considered as a breeding ground for various micro pathogens and bacteria. Direct use of hands as an input device is an innovative method for providing natural HCI ensuring minimal physical contact with the devices i.e., less transmission of bacteria and thus can prevent cross infections. Convolutional Neural Network (CNN) has been used for object detection and classification. CNN architecture for 3d object recognition has been proposed which consists of two models: 1) A detector, a CNN architecture for detection of gestures; and 2) A classifier, a CNN for classification of the detected gestures. By using dynamic hand gesture recognition to interact with the system, the interactions can be increased with the help of multidimensional use of hand gestures as compared to other input methods. The dynamic hand gesture recognition method focuses to replace the mouse for interaction with the virtual objects. This work centralises the efforts of implementing a method that employs computer vision algorithms and gesture recognition techniques for developing a low-cost interface device for interacting with objects in the virtual environment such as screens using hand gestures.

Keywords: Artificial neural networks, Computed unified device architecture, Convolutional neural network

1. INTRODUCTION

The way of interaction with the devices have an increasing impact on our everyday lives. Only a few modes of HCI research aim at focusing on improvement of current devices out there. A few modes existing includes through keyboard, touch screen, mouse and other helper devices. Each of them has certain limitations in adapting with powerful hardware connected to computers. Vision-based technology is an essential part of HCI. Gesture can be described as a symbol of expression or physical behaviour including body and hand gesture. Dynamic gesture includes movement of hand or body to convey some information. It can be used as a mean of communication between human and computer. Gesture recognition determines the motives of the user by recognition of the movement and gesture of body parts.

The workflow of dynamic hand gesture recognition includes detecting the hand region from inputs from input devices. Real-time hand gesture recognition is an intuitive and natural way to interact with the system as the interactions can be increased with the help of multidimensional use of hand gestures as compared to other input methods. By this method, the use of the mouse can be replaced for interacting with a personal computer. Any vision-based interface is more convenient, practical and natural because of the intuitiveness of gestures. By the implementation of a 3D application where the user may be able to move and rotate objects simply by moving and rotating the hands, all without the help of any input device, the future era of HCI can be efficiently evolved. This will be useful to promote controlling applications like media players, virtual games, browsing images, etc., in a virtual environment.

The main idea includes making computers understand and respond to human language and develop a friendly user-interface. A human can perform many gestures at a single time. In order to prevent physically interaction with the input devices, the user can just interact with the screen to navigate through various applications with the help of the defined gestures. The proposed method can make use of the defined gestures to help the users or the staff operates their system with the help of virtual keyboards and just gestures. Problems concerning transmission of the bacteria through HCI can be dealt with the minimalisation potential of cross-infection. Coding these gestures requires a complex algorithm to bring them into machine language. Hand gesture recognition has been a major attraction among researchers lately. Some researches were limited to recognition of minimal full sign language in small scale systems. Identifying temporal action states is an essential step for classification of that action. Action recognition has emerged to become the major attraction in the field of computer vision especially after the use of deep learning techniques in this area.

CNNs are highly being used in this domain. They are generally used for static image recognition and their performance is evaluated by spatial analysis. In video analysis, the amount of data which is to be processed is large with complex models, having temporal dimensions, which makes more challenging to recognise through the video [1]. Various modalities like infrared and flow, depth and RGB images are input, recognition performance is increased by the fusion of these modalities and analysis is done [2]. Multimodal hand gesture recognition also uses the same the same strategy. The most widely used fusion analysis by the CNNs is feature and decision level [2,3]. The most challenging one is the data level fusion as it requires frame registration when the data is captured by several hardwares. At first, the number of parameters is reduced by single network training. Then pixel-wise correspondences are established automatically due to various modalities. At the end with a little modification, CNN architecture is adopted.

Lee C and Xu Y have developed a recognition system based on glove-based gestures that could recognise 14 hand alphabet letters, update, in online mode, each gesture model with 10 Hz and could learn new gestures [4]. Later, Lee HK and Kim JH showcased work on dynamic or real time hand gesture recognition using Hidden Markov Model (HMM) [5]. A skin-tone segmentation technique was also designed in HSV space by Kjeldsen R and Kender J [6]. It was based on the idea that all the skin tones that appeared in images in HSV space occupied a connected volume. It got further updated to a system in which back propagation neural network was used on segmented hand images for recognition. Ueda E et al., showcased a hand pose estimation technique to be used for vision-based human interfaces [7]. Multiple images helped in the extraction of hand regions from camera systems and a 'voxel model' was constructed. A gesture recognition system with the use of image descriptor as the major feature was presented [8] and classification was done with RBF network. Another hand gesture recognition system based on finger tips was showcased by Nolker C and Ritter H [9]. It included finger joint angle identification based on 3D modal hand and was prepared using neural network.

CNNs, for gesture classification, treat the video frames as multi-channel inputs [10,11]. A 3D pooling and 3D convolutions are used to capture distinct features including temporal and spatial dimensions. Temporal Segment Network (TSN) [12] is also used to perform segmentation of video data and from optical and colour flow modalities, it extracts information. It can also be used to differentiate dependencies at multiple time scales. To increase the recognition performance, information fusion is done from various modalities by using CNNs. Data level, Feature and decision level fusions are there in deep learning. Out of all strategies for fusion, as a lot of efforts are required for data preparation due to different hardwares so data level fusion is hardly used [2,13].

A TSN approach is considered to be the building block for Motion Fused Frames (MFFs) due to the segmentation of video clips. Spatio-temporal state of every segment can be represented by MFFs. Another simple approach which uses traditional machine learning based on deep learning is described in this section. Traditional machine learning approaches like Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), K-Nearest Neighbour (KNN classifiers) or Decision Trees (DTs) can be used for feature-based works. A 3D skeletal joint coordinates are used to train DT and SVMs. On the other hand, Spine Center distances are used by the KNN classifier. Classification of the gesture side (right or left) and recognition of the type of gesture is done by the cascades of ANNs. Distinctive poses are the part of the recognition done by the SVMs while recognition of the sequence of these poses is done by the DT. Dynamic Time Warping (DTW) algorithms work on temporal sequences which may differ in speed and in some scenarios an ML algorithm complements classification scheme.

Various techniques are presented for dynamic as well as static gestures. Ziaie P et al., presented a method for the computation of similarities of various gestures and then through Bayesian Interface rule, assignment of probabilities [14,15]. Modification of KNN was used to estimate invariant classes consisting of geometrical attributes such as scaling, rotating and transformation of features for classification. He also presented a similar method for modified K-nearest neighbour classification algorithm which was known as locally weighted Naïve Bayes Classifier. Shrivastava R, presented a technique for the use of hand orientation and HU-moments for the extraction of features [16]. Recognition was done through Baum Welch algorithm. Another method presented by Chourasia NS et al., made use of hybrid technology feature descriptor. It combined SURF and also certain HU moments [17]. K-nearest neighbour and SVMs were used for classification purposes. Canny edge detector with the help of feature extraction was used to detect hand edges. Hunter E presented a method for the extraction of image features which used Zernike moments HMM [18]. Chaudhary A and Raheja J, presented a method for the finding the edges of finger tips by scanning all the image [19]. It resulted in the reduction of time complexity and noise removal.

Gaussian Mixture Models (GMMs) and HMM are known for solving temporal pattern recognition issues, therefore, they are widely used. A 3D skeletal joints and body parts positions are considered to be their inputs. The authors present approach for hand gesture recognition using encoders and CNNs. An estimation of hand orientation and pose serve can be done using CNN architectures. A CNN based on multimodal and multiple scale deep learning is presented. It includes two phases which are the combination of depth learning using CNNs and feature learning from RGB, PCA to derive features that are final. Gesture recognition and segmentation can be done using deep dynamic neural networks. CNN and deep belief network also help for feature learning which do not react to scaling, movement and rotation from posture images of hand. Rodriguez O et al., gave a methodology focussing on the feature extraction of gesture images through Zernike and Hu moments, using SVM for classification [20]. The use of neural networks for classification of the extracted data is another technique that was proposed by Tolba AS et al., [21]. It made use of different neural network known as learning vector quantisation. Nguyen TN et al., presented a project which focussed majorly on Principal Component Analysis (PCA) for the selection of best attributes and classification by using neural network [22]. Oyedotun OK and Khashman A gave a technique for the extraction of the shape of hands to be used as inputs which makes use of picture processing operations that compared two of the classification methods: stacked denoising autoencoder and CNN [23]. Chevtchenko SF et al., presented a methodology to improve the features that are being given as inputs to the CNN, using Gabor features, descriptors based on contours, Hu moments and Zernike moments [24]. These improvements are defined by CNN based on feature fusion. FFCNN has 3 parts: feature extraction, feature fusion and decision making. To focus on further parts and for the suppression of information which is less important, attention mechanism is applied. Ranga V et al., made use of Gabor filter with discrete wavelet transformation for feature extraction and tested it with various classifiers for the addition of a slight comparison with CNN [25]. Be and Suzuki gave a technique for the extraction of the object outlines. A set of points was derived for each element. Ramer applied another technique of polygonal approximation. It eliminated contour points those were far away from the mean curve of contour [26].

Several other methods have also been proposed before for the gesture recognition with the help of soft computational approach like ANN [27,28], genetic algorithms and fuzzy logic sets [29]. Finite State Machine (FSM) [30] and HMM [31] were some of statistical ones. ANNs have been known to solve variety of problems related to control and identification, decision making, applications for financial and data mining purposes. Taguchi K and Murakami H presented two neural network systems in which for the learning of the postures back propagation algorithm was made use [27]. Automatic sampling and data filtration improved the performance of the system. Maung THH implemented for the recognition of hand gestures, the 2D real time hand tracking [28]. However, the execution time speed for implementation that was being consumed was considerably higher. Bailador G et al., proposed Continuous Time Recurrent Neural Networks (CTRNNs) for the recognition of hand gestures in real time [32]. For each gesture class, signal predictors were created. To represent the neural parameters, genetic algorithm was used. CTRNN parameter was represented by each genetic string. The model turned out to be modular, simple and fast. Though high noise activities and movements still create a problem [33].

This paper is organised as follows, Section 2 discusses the preliminary research work related to the proposed method and defines the proposed method. Section 3 discusses the results and the last section mentions the conclusion and future prospects.

2. MATERIALS AND METHODS

In this section, the preliminary research work related to the method

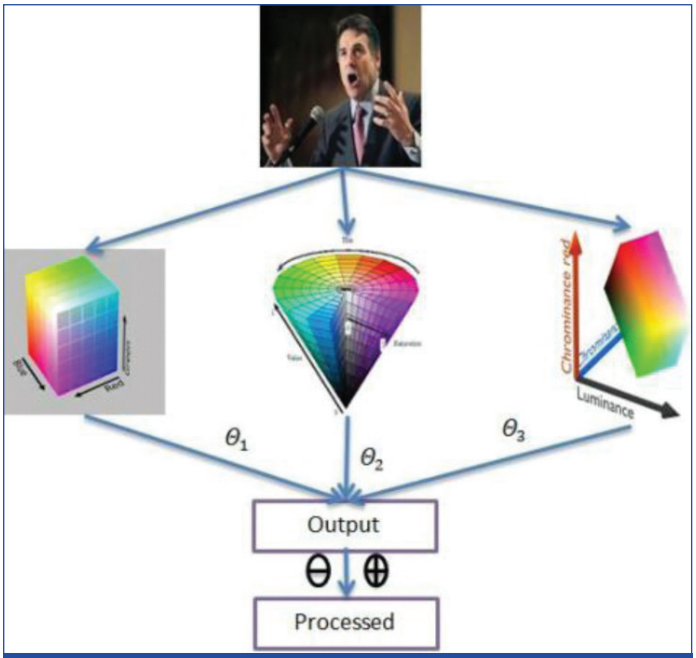
is given and then the methodology has been discussed in detail.

2.1 Preliminary Research

Skin segmentation, Neural Network, CNN and Computed Unified Device Architecture (CUDA) are discussed briefly in these sub-sections.

2.1.1. Skin Segmentation

To create any form of hand recognition system or any system that uses the recognition of a human body part, it is essential to differentiate the body part from the parts that are not. One simple way of achieving this is by applying a mask over the images that segment all the pixels, that are pixels that have a shade of skin from the ones that don't. It focuses on division of area in the given image on the basis of colour, texture or shape. Therefore, it helps us to differentiate between skin and non-skin image pixels. Skin segmentation on the basis of colour can be done by texture extraction of features and k-mean clustering techniques [34] [Table/Fig-1].



[Table/Fig-1]: Skin segmentation technique [34].

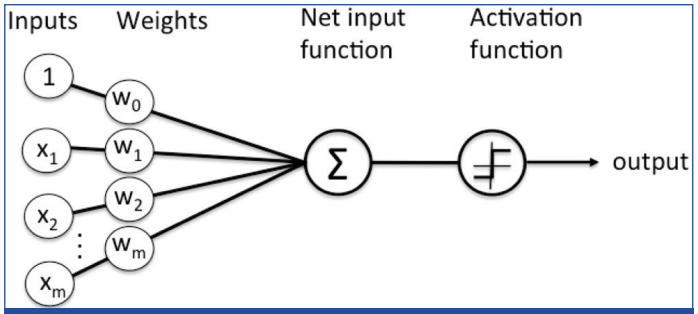
2.1.2. Neural Network

Machine learning automates and improves the computer's learning process, without being explicitly programmed, based on the experiences. It's a technique of data analysis for automating analytical building of models. It's a branch of AI (Artificial Intelligence) as the system is capable of learning from the data, pattern identification and further leads to decision making without human interference. Neural networks are multiple layered neuron networks that are used for classification and making predictions [Table/Fig-2]. It's a set of algorithms designed to identify different patterns and interpret data. The recognised patterns can be vectoral, numerical to contain real world data like sound, images or text. It also helps with clustering and classifying the data that is stored. They can be used for feature extraction that is fed as inputs to other algorithms. The layers consist of nodes where the computation occurs. It combines data inputs with coefficients, which are also known as weights, to enhance the input or amplify it to signify it. A summation of these combinations is taken and passed to an activation function for the determination of the extent of that signal's progress through the network. If it gets passed that means the neuron is activated.

At a higher level, the training process defines a cost function for its minimisation it uses gradient descent optimisation. Changing the weight or the neuron bias as an effect on all other neurons along with their subsequent layer activations.

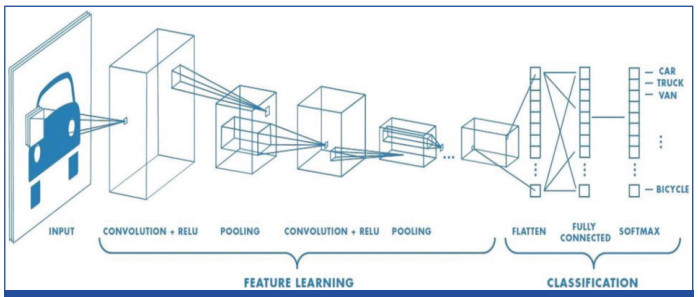
2.1.3. Convolutional Neural Network (CNN)

ANN helps with the classification of audio, text or images. CNNs are used for image classification [Table/Fig-3]. CNNs are the networks



[Table/Fig-2]: Neural network [26].

that usually share their parameters. It is most commonly used in recommendation systems and NLP (Natural Language Processing) as it detects the main features automatically without any interference. Parameter sharing helps it with running on any device.



[Table/Fig-3]: Architecture of Convolutional Neural Networks (CNN) [35].

A convolutional network consists of many sequential layers and each layer is responsible for transforming one volume with the help of differentiable functions to another. A CNN also known as CNN is a type of neural network most popularly used for analysing images, although image analysis, has been the most widespread use of CNN's they can be used for other data analysis or classification purposes. Most generally we can think of a CNN as an ANN that specialises in being able to pick out and/or make sense of patterns making it the ideal option for image classification.

Since a convolutional network is basically just a neural network the thing that differentiates the two is that the hidden layers in a CNN are replaced by what is called convolutional layers. CNN may or may not have other nonconvolutional layers but the basis of CNN is the convolutional layers. Just like any other layer a convolutional layer receives inputs and then transforms the input in some way and transfer the output to the next layer, the difference being the transformation being a convolutional operation. At each step of convolutional network, there is a filter and these filters are what that provides the network with the ability to detect patterns. Initially, the network works on smaller tasks such as detecting corners, edges or colours but as network gets trained it, starts to filter out more sophisticated patterns such as eyes, ears, etc.

2.1.4. CUDA

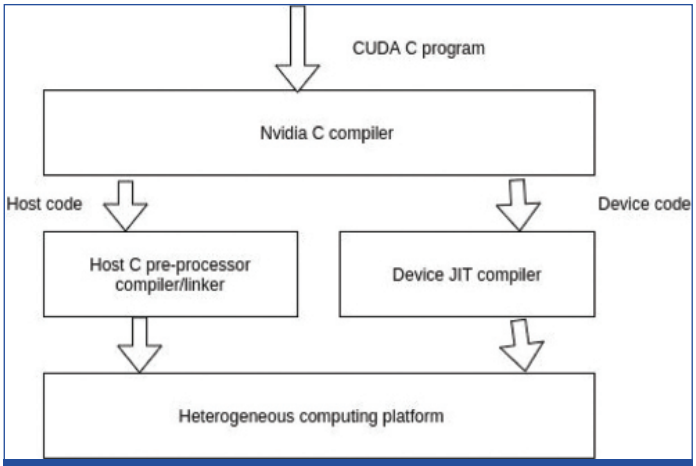
CUDA stands for Computed Unified Device Architecture [36]. It was introduced in 2006 and is a parallel computational platform developed by Nvidia. It lets the software programs in performing complex calculations that require the use of both the CPU and GPU. Instead of using only the CPU, it shares its load from the processing with the GPU. This helps in increasing the performance of the programs. It harnesses the GPU power in order to do so. Large amounts of information take large amounts of time for its execution with the help of sequential computers. Like the rendering of the pixels, the conversion of RGB pixel values to grayscale. Sequential processing of these takes a lot of time and is inefficient as the same operation is performed on the pixel on a single processor CPU. Thus, these can be parallelly processed as the processing for each pixel is different. Thus, GPU comes into picture, when it's a data intensive task. Data parallelism has many other advantages than just for the processing of computer graphics and images. There are several algebra libraries that require high performance for complex operations and with the

help of the of GPU processing power, it is possible.

General Purpose Computation on Graphics Processing Units Platforms (GPGPU) makes use of CUDA the most. It can only run on Nvidia graphics hardware and is proprietary unlike OpenCL. For development of applications for Nvidia hardware, programmers have an option to either write their code on the same platform or OpenCL as most of the video cards support OpenCL. Various coding languages can be used on CUDA. C, Python, Fortran and C++ compilers are also provided by Nvidia.

2.1.4.1. Program Structure of CUDA

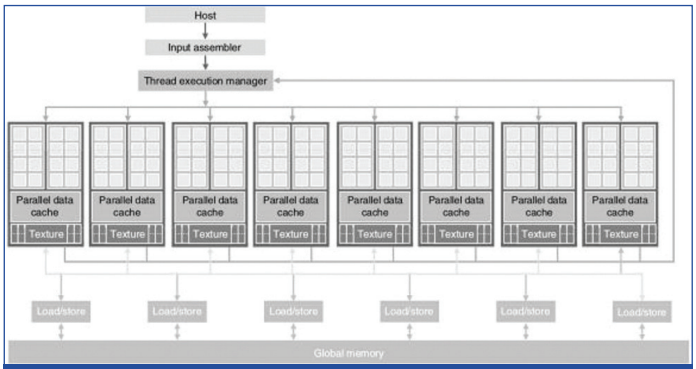
CUDA program has code which can be used for both CPU and GPU. The host is the CPU with the device being the GPU. The traditional compiler for C can be used for the host code compilation but there are API functions in the device code which needs to be understood so another compiler is needed for that. Nvidia C Compiler (NVCC) is used for this purpose. The NVCC does the separation of the device code from the host code after processing by using CUDA keywords. Kernels are the keywords that are used to mark the parallel data functions which are run the device code. NVCC later on compiles the device code and it gets executed further on the GPU [Table/Fig-4].



[Table/Fig-4]: Program structure of CUDA.
CUDA- Computed Unified Device Architecture

2.1.4.2. CUDA Capable GPU Design

Like that in the mouse or keyboards, GPUs have no virtual memory, any means to address device or interrupts. When there is no single program, multiple data, CPUs can be used to handle the problem efficiently. CUDA capable GPU architecture [Table/Fig-5]:



[Table/Fig-5]: Architecture of CUDA capable GPU [36].

Each of the 16 streaming multi-processor has 8 streaming processors. There's 128 SPs in total. With each having a Multiply and Addition unit an additional MU. Every streaming processor is threaded massively so that it can run 1000 threads per application. Though there are various other APIs for GPUs by several companies like AMD, CUDA and Nvidia GPUs are considered to be the best for application areas which consists of deep learning. Over the years since 2006, CUDA has developed its scope and improved its Nvidia GPUs. Several applications in which it has been adopted that need high computational performance are:

- Data Science and Analytics
- Defence and Intelligence
- Computational Finance
- Ocean, Climate and Monitoring of Weather
- Machine Learning and Deep learning
- Medical Imaging
- Security and Safety Purposes

2.2. Proposed Method

Under this section, the proposed approach is discussed in detail. The goal of the project initially was to develop a working dynamic gesture recognition expert system that has its applications in a wide variety of scenarios from Laptop UI control, games as well as motion recognition. For minimising the physically interaction with the input devices the user can just interact with the screen to navigate through various applications with the help of the defined gestures. The model can be customised into making use of virtual keyboards as an input method for the users for typing and searching purposes with the help of the defined gestures. The proposed method can make use of the defined gestures to help the users operate their system with the help of virtual keyboards and with just the help of the gestures for motion tracking.

Since the project calls for extremely large amounts of data to be processed the problem was that the machine learning model to be trained was extremely computationally expensive. This was a major roadblock. The model called for high robust enough model because of two major problems, background noise as well as the processing required for such huge images still was not sufficiently powerful.

For the background noise removal, the background was altogether removed by using a mask frame. This is a technique commonly used in photoshop and works by using multiple frames from the same perspective to stitch an image together. The only difference was that a mask frame was used to switch-off any pixels that are not the hand by matching them with the mask frame. This allowed for extremely sharp boundaries and almost perfectly avoided any activating problem caused due to the noise or objects in the background.

After the classification done by the model, it required to know what the user wants the gesture to do and perform the following actions i.e., UI control. To achieve this python libraries for mouse, keyboard and other APIs have been used to integrate what the gestures are meant to do. The predictions made by the CNN are sent to the file actions.py where it is used in mapping of the desired actions and sent over to the windows to act upon.

2.2.1. Requirements

Python is an advanced, interpreter-based programming language, made by "Guido van Rossum". It is very popular for its easy readability of code and compactness of line of codes. Python comes with a massive standard library which can is usable for various applications for example NLP, machine learning, data Science etc.

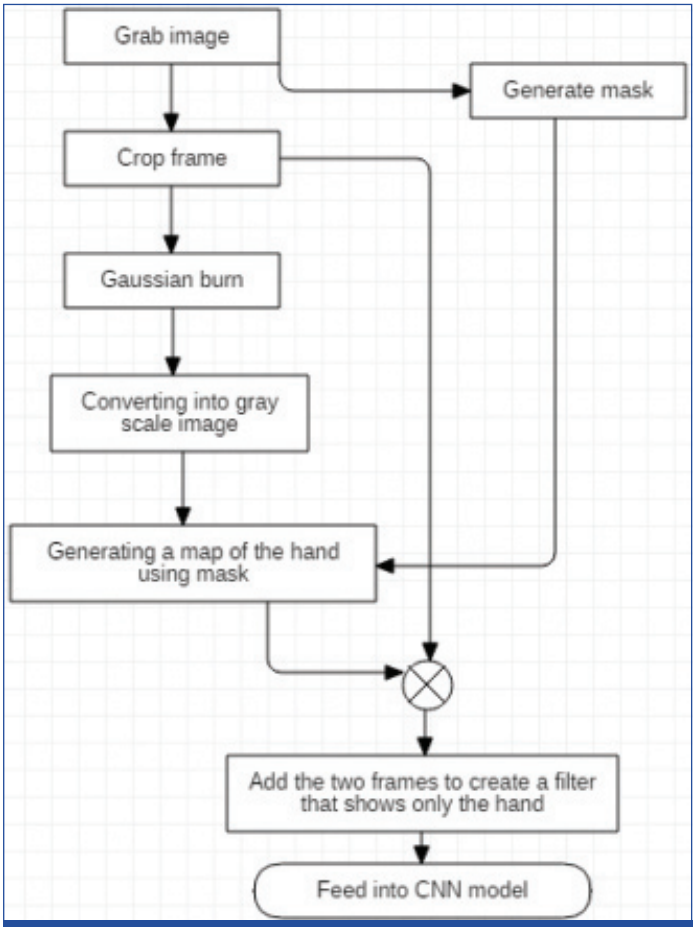
1. Python
2. CUDA: for retraining and adding gestures.
3. Libraries
    i. Keras
    ii. scikit-learn
    iii. OpenCV
    iv. pandas
    v. mouse
    vi. Numpy
    vii. joblib
4. Webcam

2.2.2. Flow chart of Image Pre-processing [Table/Fig-6]

3. EXPERIMENTAL RESULTS

3.1. Dataset

The best fit for this model that was available during the time of research was a dataset by Twenty Billion Neurons. The company is led by highly skilled and educated data scientists to create a life-size intelligent assistant, Millie. It is recognised as a CB Insights Top 100 AI start-ups of 2019 and a Gartner Cool Vendor in AI Core Technologies in 2018. One of the datasets made public by 20bn towards their goal of AI advancements is the 20bn-Jester dataset. It is one of the largest if not the largest dataset of densely labelled video clips that show humans performing predefined hand gestures. This dataset created by large number of crowd workers provides the much-needed diversity while having the required volume and variety of gestures.



[Table/Fig-6]: Flow chart for Image pre-processing.

3.1.1. Format

The video is fed as a huge tgz archive split into 1 gb maximum parts. A 22.8 GB is being the overall download size. The archive contained 1 to 148092 numbered directories. Each one of them associated with a video and contained 100 px height and variable width of jpg images. The extraction of jpg images was at the rate of 12 frames per second. 00001.jpg was the filename at which jpg started. The length of the real videos determined the number of jpg images.

This section discusses the outcome of the proposed method. The goal of the project initially was to develop a working dynamic gesture recognition expert system that has its applications in a wide variety of scenarios from Laptop UI control, games as well as motion recognition. The workflow of dynamic hand gesture recognition includes detecting the hand region from inputs from the input device which is the webcam here.

Processing of large data was computationally expensive and the background noise were the major problems faced.

After the analysis done by segmenting the images classes, it was found that there was a similarity in several shapes of the gestures which could have confused the CNN model. To reduce this, segmented image masks were used. Seven CNN models were trained and tested, all having a slight variation in the convolutional, pooling layers to improve the performance of the system. The proposed models make use of activation function in the convolution layers.

3.2 Combined Dynamic Gestures [Table/Fig-7-9]

[Table/Fig-7]: Five defined gestures.

Combined gesture type	Gesture type	Motion track
Gesture 1	One finger	Mouse control
Gesture 2	Two fingers	Scroll
Gesture 3	Three fingers	Volume control
Gesture 4	Palm	Pause
Gesture 5	Y	Mappable

[Table/Fig-8]: Trajectories for the defined gestures.

[Table/Fig-9]: Images and masks used to train the skin colour classifier.

3.3. Analysis of CNN Models [Table/Fig-10-31]

Convolutional networks consist of three parts mainly which are convolution, pooling and dense layers. All the features maps and filters are present in the convolution layer. Pooling is done to reduce the size of the features that have been obtained in the layers used before. It reduces the number of parameters that are being used and subsequently reduces the computation of network. Dense layer as the name suggests is densely connected with weight matrix and bias vector. It is a normal feed forward network layer. CNN model 1 structure consists of: Convolutional layer- max pooling- dense (relu)- dense (SoftMax)

After examination, the convolutional model 1 had a low training accuracy rate and validation accuracy rate that is a higher bias and variance. Also, the zigzag of the validation graph represents the low durability of the results. Therefore, another convolutional layer was added to the model.

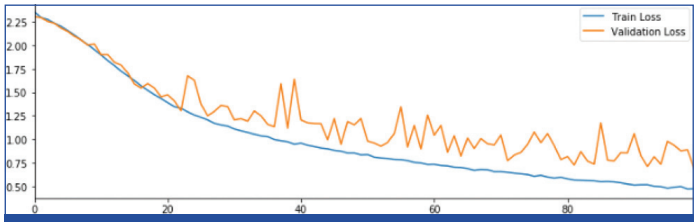
CNN model 2 structure consists of: Convolutional layer- max pooling-

another conv layer- max pooling- dense (relu)- dense (SoftMax).

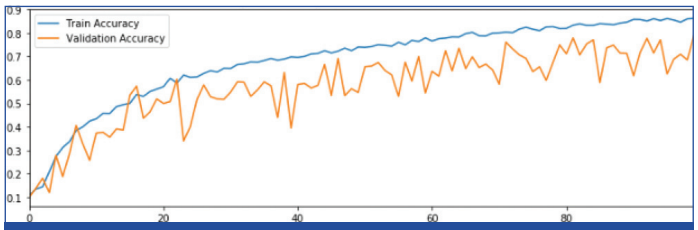
From the above, it was noted that the model had low validation accuracy but a high training accuracy rate that is high variance and low bias. The zigzag is still there, so the robustness is low. Thus, a new convolutional layer was added to avoid the overfitting of the model.

```
[INFO]:Convolutional Model 1 created...
[INFO]:Convolutional Model 1 compiled...
[INFO]:Convolutional Model 1 training....
[INFO]:Convolutional Model 1 trained....
[INFO]:Train Accuracy:0.866
[INFO]:Validation Accuracy:0.787
```

[Table/Fig-10]: Convolutional model 1 Loss and accuracy in train and validation dataset.



[Table/Fig-11]: Loss in the datasets after model 1.



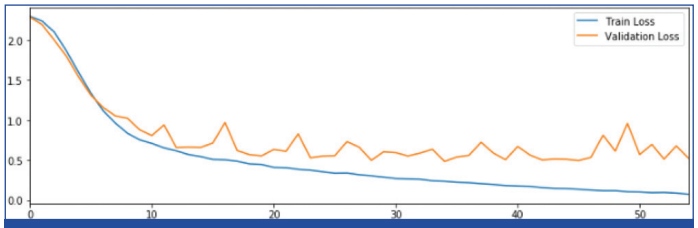
[Table/Fig-12]: Accuracy in the datasets after model 1.

CNN model 3 structure consists of: Convolutional layer- max pooling- conv layer 2- max pooling- conv layer 3- max pooling- dense (relu)- dense (SoftMax).

From the above, an increase in the validation accuracy rate was seen but the issue of overfitting still existed. For that another convolutional layer was added.

```
[INFO]:Convolutional Model 2 created...
[INFO]:Convolutional Model 2 compiled...
[INFO]:Convolutional Model 2 training....
Epoch 00055: early stopping
[INFO]:Convolutional Model 2 trained....
[INFO]:Train Accuracy:0.991
[INFO]:Validation Accuracy:0.859
```

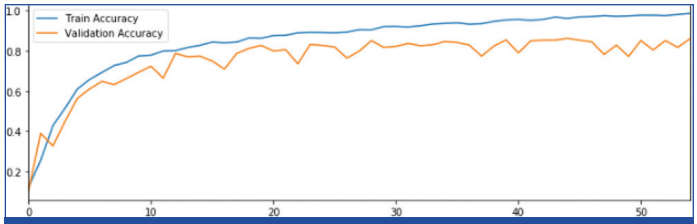
[Table/Fig-13]: Convolutional model 2 Loss and accuracy in train and validation datasets.



[Table/Fig-14]: Loss in the datasets after model 2.

CNN model 4 structure consists of: Convolutional layer- max pooling- dropout- conv layer 2- max pooling- dropout- conv layer 3- max pooling- dropout- dense (relu)- dense (SoftMax).

The validation accuracy got increased but the overfitting issue can still be seen that is high variance. CNN model 5 structure consists of: Convolutional layer- max pooling- dropout- conv layer 2- max pooling- dropout- conv layer 3- max pooling- dropout- conv layer



[Table/Fig-15]: Accuracy in the datasets after model 2.

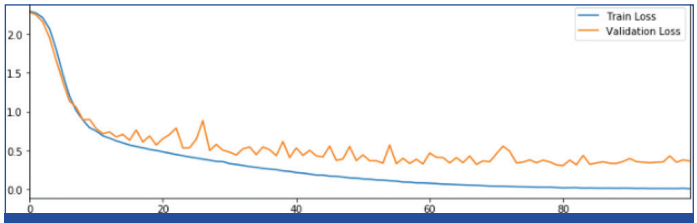
4- max pooling- dropout- dense (relu)- dense (SoftMax).

Low robustness and overfitting problem were solved but still the model had not so good training and validation performance. Dropout layers were used between the fully connected ones.

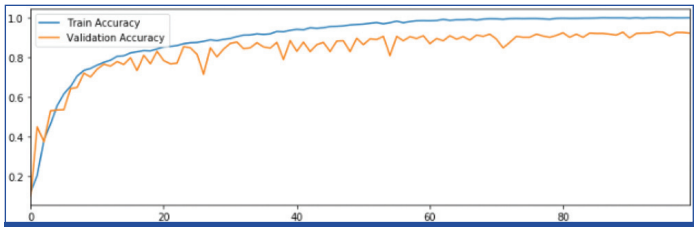
CNN model 6 structure consists of: Convolutional layer- MaxPooling- conv layer 2- max pooling- conv layer 3- max pooling- conv layer 4- max pooling- dense (relu)- dropout -dense (SoftMax).

```
[INFO]:Convolutional Model 3 created...
[INFO]:Convolutional Model 3 compiled...
[INFO]:Convolutional Model 3 training....
[INFO]:Convolutional Model 3 trained....
[INFO]:Train Accuracy:1.000
[INFO]:Validation Accuracy:0.922
```

[Table/Fig-16]: Convolutional model 3 Loss and accuracy in train and validation datasets.



[Table/Fig-17]: Loss in the datasets after model 3.



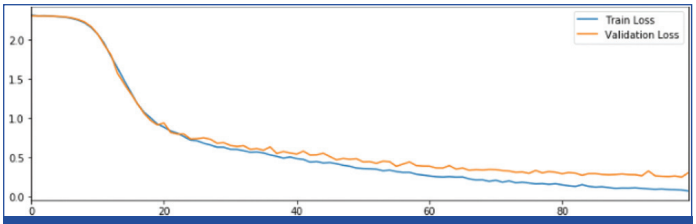
[Table/Fig-18]: Accuracy in the datasets after model 3.

The training and validation performance improved. The number of filters was fine-tuned as more and more filters were used in the deeper layers. Also, the size of the filter was also fine-tuned.

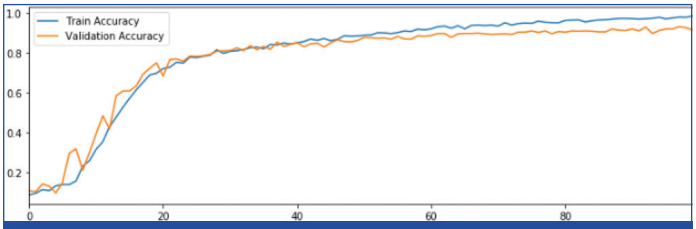
CNN model 7 structure consists of: Convolutional layer- max pooling- Batch normalisation- dropout- conv layer 2- max pooling- Batch normalisation- dropout- conv layer 3- max pooling- Batch normalisation- dropout- conv layer 4- max pooling- Batch normalisation- dropout- dense (relu)- dropout -dense (SoftMax). To prolong the training time of the deep neural networks, batch normalisation is done. Batch

```
[INFO]:Convolutional Model 4 created...
[INFO]:Convolutional Model 4 compiled...
[INFO]:Convolutional Model 4 training....
[INFO]:Convolutional Model 4 trained....
[INFO]:Train Accuracy:0.985
[INFO]:Validation Accuracy:0.914
```

[Table/Fig-19]: Convolutional model 4 loss and accuracy rates in train and validation datasets.



[Table/Fig-20]: Loss in the datasets after model 4.



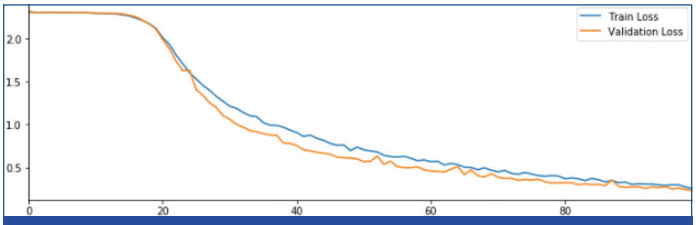
[Table/Fig-21]: Accuracy in the datasets after model 4.

normalisation is another layer which was added to the model 4.

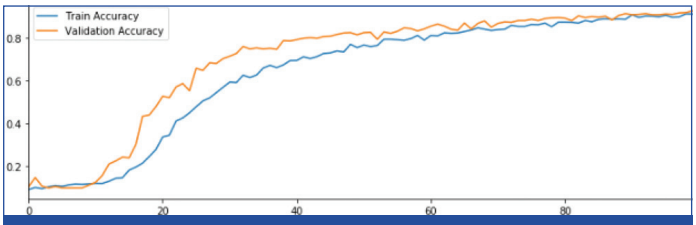
The tests were run on a server with configuration Intel(R) Core i5-7300HQ @ 2.50 GHz CPU, 8 GB of RAM, and a NVIDIA 1050ti GPU.

```
[INFO]:Convolutional Model 5 created...
[INFO]:Convolutional Model 5 compiled...
[INFO]:Convolutional Model 5 training....
[INFO]:Convolutional Model 5 trained....
[INFO]:Train Accuracy:0.967
[INFO]:Validation Accuracy:0.927
```

[Table/Fig-22]: Convolutional model 5 loss and accuracy rates in the datasets.



[Table/Fig-23]: Loss in the datasets after model 5.

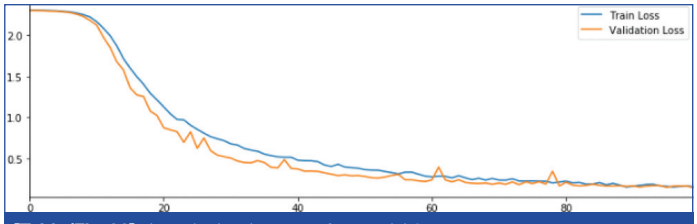


[Table/Fig-24]: Accuracy in the datasets after model 5.

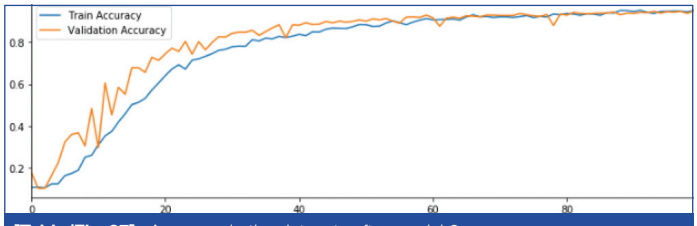
The proposed method had used the cross-validation method. Therefore, the division adopted for training is 85% and testing is 15%, respectively. From the 85% of data that has been used for training, 5% was reserved for the validation during training. Also, ten rounds of training and testing were used. The training and test sets were permuted in a random manner. The holdout metrics were derived by getting the average of the results of the 10 rounds, using: accuracy, recall and F1 score.

```
[INFO]:Convolutional Model 6 created...
[INFO]:Convolutional Model 6 compiled...
[INFO]:Convolutional Model 6 training....
[INFO]:Convolutional Model 6 trained....
[INFO]:Train Accuracy:0.990
[INFO]:Validation Accuracy:0.952
```

[Table/Fig-25]: Convolutional model 6 loss and accuracy rates in the datasets.



[Table/Fig-26]: Loss in the datasets after model 6.



[Table/Fig-27]: Accuracy in the datasets after model 6.

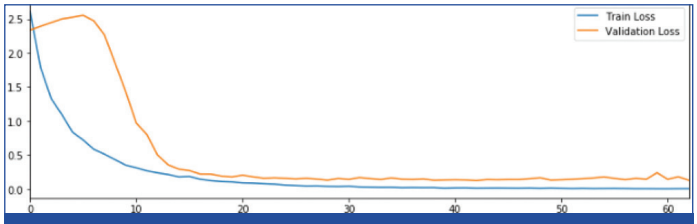
3.4 New Understandings

By working on this project there were multiple learning experiences. Initially, any work done on machine learning was on smaller datasets and hence were trained using generic hardware and using CPUs for normal computation. However, working with datasets which are in orders of magnitude larger than anything that had previously worked on, needed a different approach.

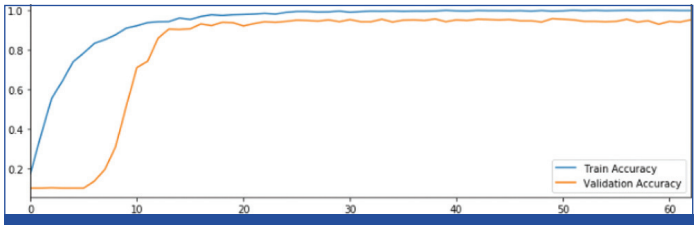
This made a requirement for higher tier hardware or using CUDA for efficient workflow. Using CUDA , it was possible to train the model in about 60 minutes which would otherwise have taken at least a day using a low-end CPU. This was extremely helpful as it taught

```
[INFO]:Convolutional Model 7 created...
[INFO]:Convolutional Model 7 compiled...
[INFO]:Convolutional Model 7 training....
Epoch 00063: early stopping
[INFO]:Convolutional Model 7 trained....
[INFO]:Train Accuracy:1.000
[INFO]:Validation Accuracy:0.952
```

[Table/Fig-28]: Convolutional model 7 loss and accuracy rates in the datasets.



[Table/Fig-29]: Loss in the datasets after model 7.



[Table/Fig-30]: Accuracy in the datasets after model 7.

how to efficiently work on larger datasets and how it is a different ballgame compared to anything basic. This also provided a deeper insight about various mathematical and logistic currently faced in the infancy of any field i.e., computer vision.

Computer vision asks for multiple masking and data pre-processing so this allows for many advantages like faster computation, smaller datasets, less complex models as well more accurate classification and detection. Even the background removal technique which has been dealt is common with people who work with images using softwares

like photoshop. By combining inputs from the two fields, it was possible to elevate and to tackle one of the biggest problems faced by any object detection algorithm i.e., isolating subjects from the background.

	CNN 1	CNN 2	CNN 3	CNN 4	CNN 5	CNN 6	CNN 7
Train accuracy	86%	99%	99%	98.5%	96.7%	99%	99%
Validation accuracy	78%	85%	92%	91.4%	92.7%	95.2%	95.2%

[Table/Fig-31]: Results obtained with the used models.
CNN- Convolutional Neural Network

Another major new learning was during the tuning of the hyperparameters. In the initial stages of training a model, each layer should be carefully set and have the required activation function. Another important factor is training duration including batch size and epochs, a balance is required between epochs, batch size and the early stopping parameters.

Controlling infections has turned out to be the major concern of the healthcare environment. Several input devices such as keyboards, mouse, touchscreens can be considered as a breeding ground for various micro pathogens and bacteria. In order to prevent physically interaction with the input devices, the user can just interact with the screen to navigate through various applications with the help of the defined gestures. The proposed method can make use of the defined gestures to help the users or the staff operate their system with the help of virtual keyboards and just gestures. Problems concerning transmission of the bacteria through HCI can be dealt with the minimalisation the potential of cross-infection. Gesture controlled interface can be customised further to be accessed more effectively and easily for infection control.

4. CONCLUSION(S)

The workflow of dynamic hand gesture recognition includes detecting the hand region inputs from input devices. As any vision-based interface is more convenient, practical and natural because of the intuitiveness of gestures. So, by this method the use of the mouse can be replaced for interacting with a personal computer.

In the healthcare environment, controlling of cross infections is currently one of the main concerns and most of the input devices such as keyboards, mouse, touchscreens are considered as a breeding ground for various micro pathogens and bacteria. In order to prevent physically interaction with the input devices, the user can just interact with the screen to navigate through various applications with the help of the defined gestures and even expand or customise the gestures according to the usage.

The main problem of the image background and the noise in a gesture recognition system in the region of interest has been dealt. To distinguish hand region from the background noise skin, segmentation through neural network has been done which was further fed into morphological operations. It helps in the removal of irrelevant image objects before feeding them to the classification method. The CNN then extracts only the important features (gestures) with the help of layers of convolution and pooling. It helps in increasing the accuracy of the model. A separate operation for the combination of segmentation masks and the original images improves the information about the fingers. The proposed CNN model that has been used gives better success rates at a comparatively computation cost that is low. The accuracy rates were quite similar to the relative architectures for gesture recognition that had earlier been used but this has a lower computational cost [Table/Fig-32]. The image pre-processing methodology that had been used removes the unwanted information and thus improves the extraction of features by the CNN model.

This model has been used to create an expert system which takes the gesture input from the user through web cam and is able to navigate through the computer screen. It performs various actions in place of mouse and controls the cursor movements. It can be used to control the volume of the system by either increasing or decreasing it with the help of hand gestures. Through this model can further be used to implement dynamic gesture recognition in embedded devices. With the help of other colour segmentation techniques and deep learning architectural models, this dynamic gesture recognition expert system can be made to work on other methodologies for data pre-processing that would be comparatively faster. The proposed method can make use of the defined gestures to help the users or the staff operate their system with the help of virtual keyboards and just gestures. The users can make use of simple hand gestures to review, search and read literature or even make use of virtual on-screen keyboard for the same. Gesture controlled interface can be customised further, to be accessed more effectively and easily for infection control.

4.1. Future Prospects

Virtual reality: It is used in device stimulated environments that are capable of stimulating physical presence in real world out there along with imaginary ones. Some of the stimulations consists of sensory information, with the help of headphones. While the advanced ones consist of force feedback generally in gaming environments. **Sign language:** It's a form of natural language and the most ancient ones as far as civilisation is concerned. It was there before the spoken languages. It is being used worldwide by hearing impaired people in the field of sports, work places and religious practices. It enhances the significance of spoken language and improves thoughts and expressions.

For medical purposes: Gesture controlled interface can be customised further to be accessed more quickly, effectively and easily for infection control. Operating rooms of medical facilities can have gesture-controlled interfaces to reduce the time taken in scrubbing and washing of the hands by the staff. This way the transmission of the micro-pathogens can be controlled too.

	Proposed method-dataset	Otiniano-rodriguez KC et al., [20]	Nguyen TN et al., [22]	Tolba AS et al., [21]	Oyedotun OK and Khashman A [23]	Chevtchenko SF et al., [24]	Ranga V et al., [25]
Accuracy	91.21%	96.27%	94.3%	90.83%	91.33%	98.06%	97.01%

[Table/Fig-32]: Comparison with related work [20-25].

For embedded devices: Intelligent robots used for detection, security and other services are based on the embedded systems. And being able to interact with the humans is the major aim of these robots. Such type of an expert system can make it more convenient for interactions between humans and robots. Recognition hand gestures for various video streams on embedded systems can be done. Real-time algorithm for gesture recognition and the given embedded device can be used to further control such intelligent robots.

Infrared gesture sensing: Most of the embedded devices make use of user interfaces which don't require any touch. It's considered to be one of the most innovative and creative ways for interaction with the electronic devices. Active proximity infrared motion sensor uses light sensor technology to sense gestures within a range. Phase and position-based sensing is also done to calculation the object location, timing and direction of the motion.

Any vision-based interface is more convenient, practical and natural because of the intuitiveness of gestures. By the implementation of 3D application where the user may be able to move and rotate objects simply by moving and rotating the hands, all without the help of any input device, the future era of HCI can be efficiently evolved. This will be useful to promote controlling applications like media players, virtual games, browsing images etc. in a virtual environment.

Acknowledgement

This research was funded by the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University through the Fast-

track Research Funding Program.

REFERENCES

[1] Asadi-Aghbolaghi M, Clapés A, Bellantonio M, Escalante HJ, Ponce-López V, Baró X, et al. A survey on deep learning based approaches for action and gesture recognition in image sequences. 12th IEEE International Conference on Automatic Face Gesture Recognition. 2017;476-83. Doi: 10.1109/FG.2017.150.

[2] Feichtenhofer C, Pinz A, Zisserman A. Convolutional two-stream network fusion for video action recognition. Computer Vision and Pattern Recognition, 2016. Doi: 10.1109/CVPR.2016.213.

[3] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. Advances in neural information processing systems. 2014;568-76.

[4] Lee C, Xu Y. Online, interactive learning of gestures for human robot interfaces, Carnegie Mellon University, The Robotics Institute, Pittsburgh, Pennsylvania, USA, 1996.

[5] Lee HK, Kim JH. An HMM-Based Threshold model approach for gesture recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1999;21(10):961-73. Doi: 10.1109/34.799904.

[6] Kjeldsen R, Kender J. Finding skin in colour images. In IEEE Int. Conf. on autom. Face and Gesture Recognition, 1996.

[7] Ueda E, Matsumoto Y, Imai M, Ogasawara T. Hand pose estimation for vision-based human interface. IEEE Transactions on Industrial Electronics. 2003;50(4):678-84. Doi: 10.1109/TIE.2003.814758.

[8] Ng CW, Ranganath S. Real-time gesture recognition system and application. Image Vision Comput. 2002;20(13-14):993-1007. Doi: 10.1016/S0262-8856(02)00113-0.

[9] Nolker C, Ritter H. Visual recognition of continuous hand postures. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews. 2001;31(1).

[10] Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. Learning spatiotemporal features with 3d convolutional networks. IEEE International Conference Computer Vision (ICCV). 2015;4489-97. Doi: 10.1109/ICCV.2015.510.

[11] Varol G, Laptev I, Schmid C. Longterm temporal convolutions for action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017. Doi: 10.1109/TPAMI.2017.2712608.

[12] Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, et al. Temporal segment networks: Towards good practices for deep action recognition. European Conference on Computer Vision, Springer. 2016;20-36. Doi: 10.1007/978-3-319-46484-8_2.

[13] Kopuklu O, Gunduz A, Kose N, Rigoll G. Real-time hand gesture detection and classification using convolutional neural networks. Institute for Human-Machine Communication, TU Munich, Germany, 2019. Doi: 0.1109/FG.2019.8756576.

[14] Ziaie P, Muller T, Knoll A. A novel approach to hand-gesture recognition. in IEEE, Technische Universitat München, 2008. Doi: 10.1109/IPTA.2008.4743760.

[15] Ziaie P, Knoll A. An invariant-based approach to static Hand-Gesture Recognition. 18th International Conference on Artificial Reality and Telexistence, 2008.

[16] Shrivastava R. A hidden Markov model based dynamic hand gesture recognition system using OpenCV. In IEEE International Advance Computing Conference (IACC), Ghaziabad, India, 2013. Doi: 10.1109/IAdCC.2013.6514354.

[17] Chourasia NS, Dhote K, Saha S. Analysis of hand gesture recognition using sign language through computer interfacing. IJESIT. 2014;3(3):631-37.

[18] Hunter E. Posture estimation in reduced model gesture input systems. In Internal Workshop on Automated Face and Gesture Recognition, 1995.

[19] Chaudhary A, Raheja J. A vision based geometrical method to find fingers positions in real time hand gesture recognition. Journal of Software Academy Publisher. 2012;7:. Doi: 10.4304/jsw.7.4.861-869.

[20] Otiniano-Rodríguez KC, Camara-Chávez G, Menotti D. Hu and zernike moments for sign language recognition. In International Conference on Image Processing, Computer Vision, and Pattern Recognition, 2012.

[21] Tolba AS, Elsoud MA, Elnaser OA, "LVQ for hand gesture recognition based on DCT and projection features. Journal of Electrical Engineering. 2009;60(4):204-08.

[22] Nguyen TN, Huynh HH, Meunier J. Static hand gesture recognition using principal component analysis combined with artificial neural network. Journal of Automation and Control Engineering. 2015;3:. Doi: 10.12720/joace.3.1.40-45.

[23] Oyedotun OK, Khashman A. Deep learning in visionbased visionbased. Neural Computing and Applications. 2017;28(12):3941-51. Doi: 10.1007/s00521-016-2294-8.

[24] Chevtchenko SF, Vale RF, Macario V, Cordeiro FR. A convolutional neural network with feature fusion for realtime hand posture recognition. Applied Soft Computing. 2018;73:748-66. Doi: 10.1016/j.asoc.2018.09.010.

[25] Ranga V, Yadav N, Garg P. American sign language fingerspelling using hybrid discrete wavelet transform-gabor filter and convolutional neural network. Journal of Engineering Science and Technology. 2018;13(9):2655-69.

[26] Pinto RF, Borges CDB, Almeida AMA, Paula IC. Static hand gesture recognition based on convolutional neural networks. Hindawi Journal of Electrical and Computer Engineering. 2019;12. Doi: 10.1155/2019/4167890.

[27] Taguchi K, Murakami H. Gesture recognition using recurrent neural networks. in SIGCHI Conference on Human Factors in Computing Systems: Reaching through Technology, 1999.

[28] Maung THH. Real-time hand tracking and gesture recognition system using neural networks. World Academy of Science, Engineering and Technology. 2009;50:466-70.

[29] Li X. Gesture Recognition Based on Fuzzy C-Means Clustering Algorithm, Department of Computer Science, The University of Tennessee Knoxville, 2003.

[30] Verma R, Dev A. Vision based hand gesture recognition using finite state machines and fuzzy logic. in IEEE International Conference on Ultra Modern Telecommunications and Workshops, Petersburg, Russia, 2009. Doi: 10.1109/ICUMT.2009.5345425.

[31] Lien C, Huang C. The model-based dynamic hand posture identification using genetic algorithm. SpringerMachine Vision and Applications. 1999;2(3):107-21. Doi: 10.1007/s001380050095.

[32] Bailador G, Roggen D, Troster G. Real time gesture recognition using continuous time recurrent neural networks. In 2nd ICST International Conference on Body Area Networks, 2007. Doi: 10.4108/bodynets.2007.149.

[33] Mekala P, Fan J, Lai WC, Hsue CW. Gesture recognition using neural networks based on HW/SW Cosimulation Platform. Hindawi Publishing Corporation, 2013. Doi: 10.1155/2013/707248.

[34] Chen ZH, Kim JT, Liang J, Zhang J, Yuan YB. Real-time hand gesture recognition using finger segmentation. 2014. Doi: 10.1155/2014/267872.

[35] [Online]. Available: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148.

[36] [Online]. Available: https://www.geeksforgeeks.org/running-python-script-on-gpu/.

PARTICULARS OF CONTRIBUTORS:

1. Associate Professor, Department of Information Technology, Amity University, Gautam Budh Nagar, Sector-125, Noida, Uttar Pradesh, India.
2. Associate Professor, Department of Computer Science Engineering, Amity University, Tashkent, Uzbekistan. (On deputation from Amity University Noida, UP, India)
3. Scholar, Department of Computer Science Engineering, Amity University, Gautam Budh Nagar, Sector-125, Noida, Uttar Pradesh, India.
4. Scholar, Department of Computer Science Engineering, Amity University, Gautam Budh Nagar, Sector-125, Noida, Uttar Pradesh, India.
5. Associate Professor, Department of Electrical Engineering, College of Engineering, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia (On leave from Gautam Buddha University, Uttar Pradesh, India).
6. Assistant Professor, Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyad, Saudi Arabia.

NAME, ADDRESS, E-MAIL ID OF THE CORRESPONDING AUTHOR:

Fadwa Alrowais,
Department of Computer Sciences, College of Computer and Information Sciences,
Princess Nourah Bint Abdulrahman University, Airport Road, Riyad PO 84428,
Kingdom of Saudi Arabia.
E-mail: fmalworais@pnu.edu.sa

PLAGIARISM CHECKING METHODS: [Jain H et al.]

- Plagiarism X-checker: May 15, 2020
- Manual Googling: Jun 19, 2020
- iThenticate Software: Jul 31, 2020 (9%)

ETYMOLOGY: Author Origin

AUTHOR DECLARATION:

- Financial or Other Competing Interests: None
- Was Ethics Committee Approval obtained for this study? NA
- Was informed consent obtained from the subjects involved in the study? Yes
- For any images presented appropriate consent has been obtained from the subjects. Yes

Date of Submission: **May 14, 2020**
Date of Peer Review: **Jun 08, 2020**
Date of Acceptance: **Jun 19, 2020**
Date of Publishing: **Aug 01, 2020**