iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.53487

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

Contactless System Navigation Using Dynamic Hand Gesture Recognition

Abhijeet Prasad[1], Satyajit Bhosale[2], Abdul Shaikh[3]
[1, 2, 3]*Dept. of Computer Engineering, SPPU*

Abstract: ***Hand gestures are a natural way for human-robot interaction. Vision based dynamic hand gesture recognition has become a hot research topic due to its various applications. Traditional (or conventional) methods of human-computer interaction, namely, keyboards, pointing devices, and similar physical input tools, and the more popular and commonly used touchscreens, are only as useful through the application of pressure through physical touch. The recent pandemic has driven a major argument about the substantial and often, indispensable need for contact-less operation of systems, irrespective of the scale of the industry. Say for example, supply chain systems that are handled by scores of employees in the span of a day. It is quite fundamental knowledge now, that how public systems that work on physical contact can pose as potential vectors for diseases. Hand gesture recognition can turn out to be an excellent solution to this problem; facilitating truly contact-less interaction with computing devices, and at the same time providing dependable and respectable efficiency. This approach will be further elaborated in upcoming editions of this research article.***

Keywords: Convolutional Neural Network (CNN), Long Short-Term Memory(LSTM), hand gesture recognition, short-term sampling

I. INTRODUCTION

A human gesture can be defined as a sequence of states in the short-lived motion of some part of the body, most commonly the hands and face. Consider the wave of a hand in order to open a door controlled by sensors. The gesture has an initial position and a final position, with many small states in the transition from the initial state to the final state. For example, for a left-to-right wave, the hand is initially at the left, and is moved to the right in a quick and swift movement. This very well explains the meaning of a gesture.

Existing techniques of hand gesture recognition can be divided into two broad categories on the basis of method of obtaining input – vision-based techniques and non-vision-based techniques. Vision-based techniques usually imply the use of cameras or motion sensors to detect. movement and thus extract input. No extra equipment is generally required in this approach. Non-vision-based techniques commonly use hardware equipment to extract input, such as wearable devices. These wearable devices are equipped with various types of mechanical or optical sensors, that translate movement of the body part into electrical signals.

Vision-based gesture tracking presents itself to be a more practical and feasible measure, since there is no need for the user to have an extra hardware device (e.g., wearables) for motion to be detected – as long as the motion is in the tracking frame of the sensor, it can be taken as an input for gesture recognition. It is quite common knowledge that computer systems and applications can be controlled (or navigated) by the use of remote-like devices that generate signals which are tracked by the system through sensors, and the respective operations are performed by the system. Consider a PlayStation Move motion controller for a clearer picture. But what if we could eliminate the need for these physical devices and achieve truly contact-less system control, through mere hand gestures in front of a camera or similar sensors?

Various hand gesture recognition techniques can be used for the purpose of dynamic hand gesture recognition. In this paper, we will discuss several approaches proposed by respective authors in recent years for the purpose of recognition of dynamic hand gestures. The fundamental approach seen in all the respective research reviewed is the employment of an artificial neural network as a base element in the process of performing hand gesture recognition.

An artificial neural network is very much like its biological counterpart found in human beings. It resembles the neural network in humans that is composed of a great number of fundamental units called ‘neurons’. These neurons are responsible for the intellectual functioning of the human being and facilitate in tasks like thinking, decision-making and other tasks involving the use of the brain’s computational abilities.

The basic structure of an artificial neural network is shown in Figure 1. An artificial neural network is generally composed of three distinct layers – the input layer, the hidden layer, and the output layer.

The input layer takes the subject to be worked upon and passes it further in a form that can be processed. The hidden layer may be composed of several layers within, each responsible for identifying a specific pattern or feature in the input data. The greater the number of layers, the greater is the quality of output obtained. The hidden layer produces a result and passes it on to the output layer.

The output layer helps in projecting the result obtained after the main working of the model.

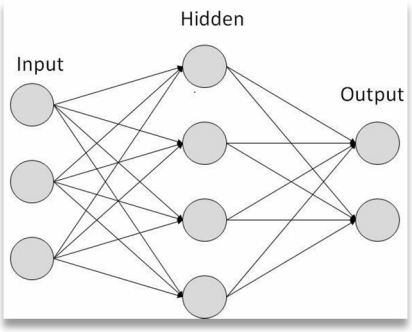


Fig 1: Basic structure of artificial neural network

II. LITERATURE REVIEW

Zhang, et al. Designed a 3-dimensional convolutional neural network model[1]. This study applies a deep learning method to recognise hand gestures. 3D Convolution neural network can be seen as a variant of 2D convolution neural network extending 2dimension filter into 3 dimensions. This 3D filter shall slide in 3 directions to extract low-level features and its output's shape is a 3-dimension space like a cuboid. They used the Jester V1.0 hand gesture dataset to train the model. According to the result of the training experiment, it got an average accuracy of 90%. Figure 2 shows the overview of system architecture for the same.

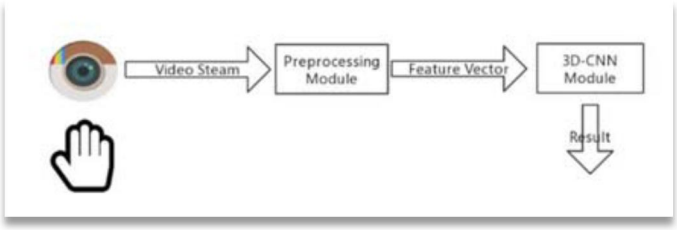


Fig. 2: System architecture for 3D CNN model

Wang et al. developed temporal segment networks (TSN) in [2], for action recognition. In the TSN model, each input video sample is divided into a number of segments and a short snippet is randomly selected from each segment. The snippets are represented by modalities such as RGB frames, optical flow and RGB differences. Convolutions neural networks (ConvNets) that are used to learn these snippets all share parameters. The class scores of different snippets are fused by the segmental consensus function to yield segmental consensus, which is a video-level prediction. Predictions from all modalities are fused to produce the final prediction. Experiment showed that the model not only achieved very good action recognition accuracy but also maintained reasonable computation cost.

Min, et al. [3] formulate gesture recognition as an irregular sequence recognition problem and aim to capture long-term spatial correlations across point cloud sequences. A novel and effective PointLSTM is proposed to propagate information from past to future while preserving the spatial structure. The proposed PointLSTM combines state information from neighbouring points in the past with current features to update the current states by a weight-shared LSTM layer. This method can be integrated into many other sequence learning approaches. In the task of gesture recognition, the proposed PointLSTM achieves state-of-the-art results on two challenging datasets (NVGesture and SHREC'17) and outperforms previous skeletonbased methods. To show its advantages in generalization, we evaluate our method on MSR Action3D dataset, and it produces competitive results with previous skeleton-based methods.

Tang, et al. [4] combined image entropy and density clustering to exploit the key frames from hand gesture video for further feature extraction, which can improve the efficiency of recognition.

Moreover, a feature fusionstrategy is also proposed to further improve feature representation, which elevates the performance of recognition. To validate our approach in a ”wild” environment, we also introduce two new datasets called Hand Gesture and Action3D datasets. Experiments consistently demonstrate that our strategy achieves competitive results on Northwestern University, Cambridge, HandGesture and Action3D hand gesture datasets.

Cheng, et al. [5] proposed a Dynamic Graph-Based Spatial-Temporal Attention (DG-STA) method for hand gesture recognition. The key idea is to first construct a fully connected graph from a hand skeleton, where the
node features and edges are then automatically learned via a self-attention mechanism that performs in both spatial and temporal domains. We further propose to leverage the spatial-temporal cues of joint positions to guarantee robust recognition in challenging conditions. In addition, a novel spatial temporal mask is applied to significantly cut down the computational cost by 99%.

Zhang, Wang, Lan [6] presented a novel deep learning network for hand gesture recognition. The network integrates several well-proved modules together to learn both short-term and long-term features from video inputs and meanwhile avoid intensive computation. To learn short-term features, each video input is segmented into a fixed number of frame groups. A frame is randomly selected from each group and represented as an RGB image as well as an optical flow snapshot. These two entities are fused and fed into a convolutional neural network (ConvNet) for feature extraction. The ConvNets for all groups share parameters. To learn long term features, outputs from all ConvNets are fed into a long short-term memory (LSTM) network, by which a final classification result is predicted. The new model has been tested with two popular hand gesture datasets, namely the Jester dataset and Nvidia dataset. The robustness of the model has also been proved with an augmented dataset with enhanced diversity of hand gestures. Uses the Jester dataset and Nvidia Dataset achieving an accuracy of around 95%. Figure 3 shows an overview of the working of the proposed model.

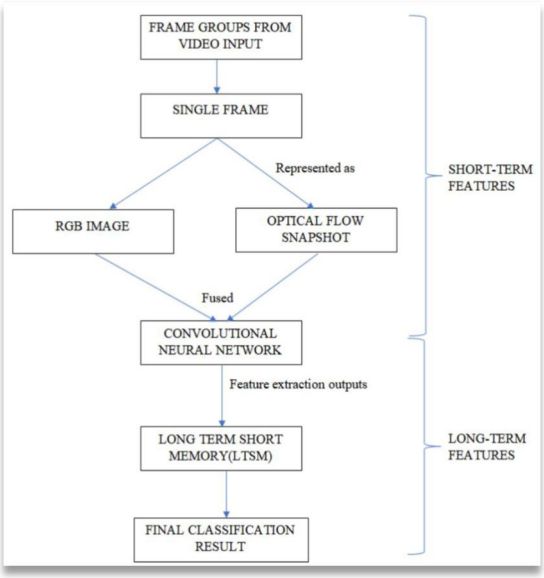


Fig. 3: Working of system described in [6]

Chung et al. [7] proposed a method for hand gesture recognition using deep convolutional neural networks(CNNs). They used a webcam to track the hand region. Firstly, they used skin color detection and morphology to remove unnecessary background information(noise) from the image, and sub-tract the background to obtain the region of interest. To avoid background influences affecting the region of interest, they used kernelized correlation filters(KCF) algorithm to track the detected region of interest. The image was then entered into the deep convolutional neural network. Two deep CNN architectures were developed in the study. The whole process was repeated in order to obtain an instant effect; with the system continuing execution as long as the hand is in the camera range. In this study, the training dataset reached a recognition rate of 99.90%.

Rahim et al [8] proposed an approach which segments hand gestures by comparing the segmentation methods of YCbCr, SkinMask and HSV(Hue, Saturation, Value). The chroma red(Cr) component is extracted from the YbCbCr model, after which operations like binarization, erosion and hole filling are carried out. Color segmentation is applied to SkinMask procedure that detects the pixels that match with the color of the hand in the frame.

By the HSV process, threshold masking determines the dominant features of the input. The Softmax classification algorithm was used for the pupose of classification of hand gestures. Convolutional neural network was used to extract features.

Xu et al [9]. proposed a recognition method of both static and dynamic hand gestures. Firstly, for static hand gesture recognition, starting from the hand gesture contour extraction, the palm centre is identified by Distance Transform (DT) algorithm. The fingertips are localized by employing the K-Curvature-Convex Defects Detection algorithm (K-CCD). On the basis, the distances of the pixels on hand gesture contour to palm centre and the angle between fingertips are considered as the auxiliary features to construct a multimodal feature vector, and then recognition algorithm is presented to robustly recognize the static hand gestures. Secondly, combining Euclidean distance between hand joints and shoulder centre joint with the modulus ratios of skeleton features, this paper generates a unifying feature descriptor for each dynamic hand gesture and proposes an improved dynamic time warping (IDTW) algorithm to obtain recognition results of dynamic hand gestures.

Sarma, Bhuyan[10] presented a model-based method for hand gesture recognition using convolutional neural network. The model was fed with trajectory-to-contour based images that were obtained from isolated trajectory gesture though segmentation and tracking of the hand motion, and the hand motion trajectory was estimated. Deep learning techniques were used to learn image features hierarchically from local to global with multiple layers of abstraction based on the sample images in the dataset. In this method, feature learning capability of CNN architecture gave quite commendable results, while tested on three different datasets.

Sriram, Nagaraj et al[11] proposed a system to perform computer mouse functions and scroll functions using a web camera or built-in camera in the computer instead of using a traditional mouse device. OpenCV, the Python library for computer vision is used; along with Mediapipe package for hand-tracking. Pynput, Autopy and PyAutoGUI packages were used for maving around the window screen of the computer for performing functions such as left click, right click and scrolling functions. The proposed model worked very well in real-world application with the use of a CPU, instead of requiring a GPU(Graphics Processing Unit). The model had an accuracy of around 95%. Figure 4 shows the overall working of the system.

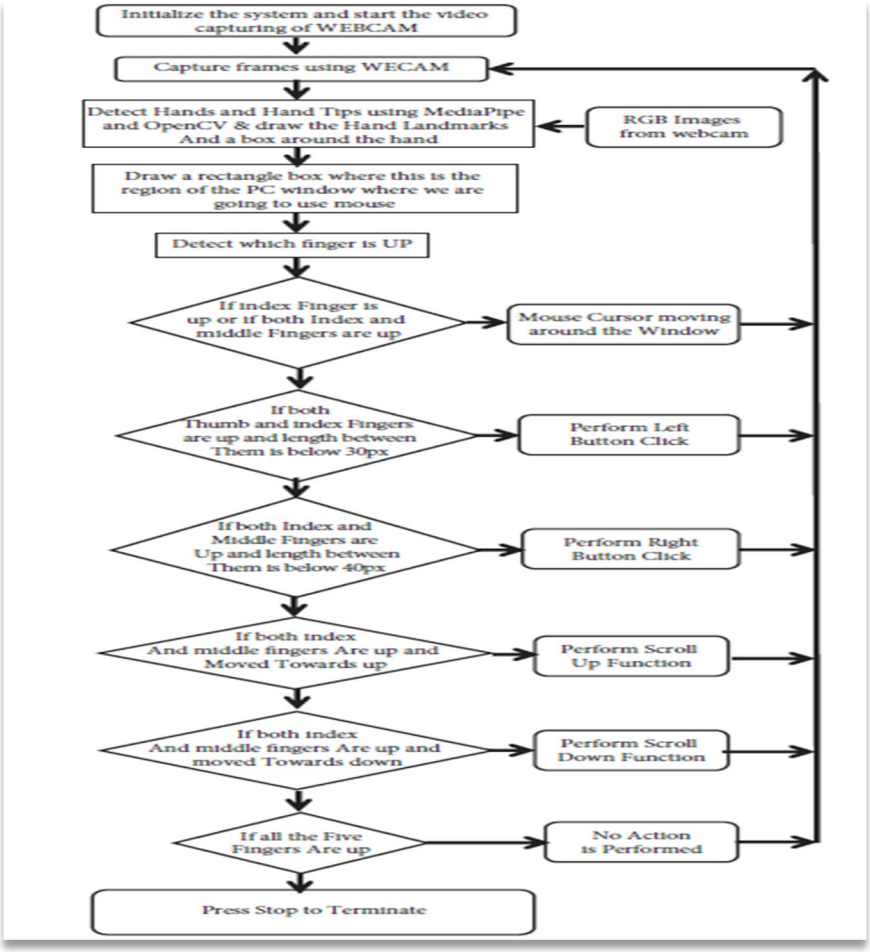


Fig. 4: Working of system described in [11]

Tran, Ho et al[12] proposed a novel virtual-mouse method for performing basic system control using RGB-D(depth) images and fingertip detection. The hand region of interest and the center of the palm are first extracted using in-depth skeleton-joint information images, and then converted into a binary image. Then, the contours of the hands are extracted and described using a border-tracking algorithm. Finally, the fingertip location is mapped to RGB images to control the mouse cursor on a virtual screen. The experimental results showed an accuracy of around 96%. This fingertip-based interface can work well in real-world environment.

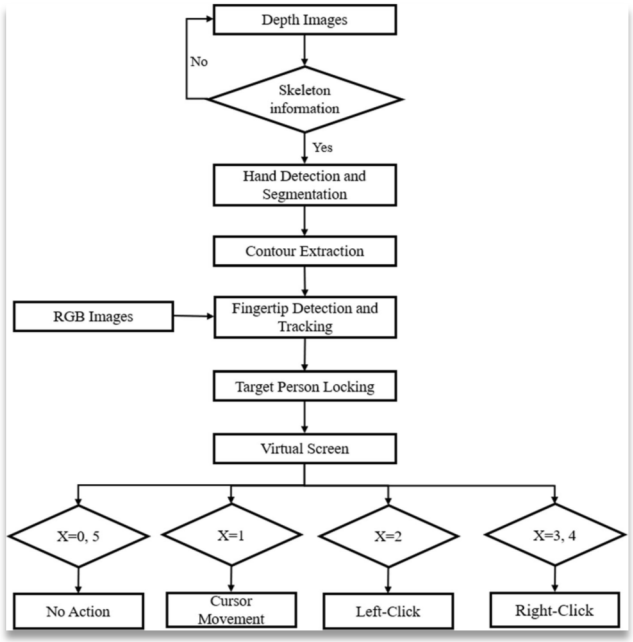


Fig. 5: Working of system in [12]

Shibly, Islam et al[13] developed a human-computer interaction system involving a virtual mouse, using computer vision and hand gestures captured using a webcam and processed with color segmentation and detection technique. The user can control basic cursor functions(namely, left-clicks, right-clicks, double-clicks, scrolling up or down) using their hands, which bear colored caps on the fingertips as markers. OpenCV is the fundamental library used for overall working of the system. The developed system achieved an average accuracy of about 78%.

III. PROJECT WORKING DETAIL

A. Main modules in our project are

1) Tracking_module.

2) Main_Driver.

B. The library used in Tracking modules are:

1) Open CV

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.

In our project we have used OpenCV for capturing the video stream from camera, general GUI interface etc. Using OpenCV we have highlighted some of backend process by displaying it on to the monitor like showing live hand tracking, current gesture, and hand skeleton.

2) Mediapipe

MediaPipe is an open-source framework for building pipelines to perform computer vision inference over arbitrary sensory data such as video or audio. Using MediaPipe, such a perception pipeline can be built as a graph of modular components.

Mediapipe is the main library in our project. We have used it for hand-tracking. It provides the accurate x, y coordinate of the landmarks. Landmarks are the pre assumed point on the hand. Further we have used this data for our gesture recognition module which is present in Main Driver module.

3) Math

Math is the python-based library which contain the most the mathematical formula. We have used the math library for different operations like find the distance with 2 point and FPS calculation.

C. The library used in Main_Driver.

1) Numpy

NumPy is a Python library used for working with arrays.
It also has functions for working in domain of linear algebra, Fourier transform, and matrices.
NumPy was created in 2005 by Travis Oliphant. It is an open-source project, and you can use it freely.
NumPy stands for Numerical Python.
We have used it for performing normalization on tracking data and for handling the landmarks data. The tracking module generates the data for every frame of the video in the form of array, hence it is necessary to use NumPy for handling all the generate data.

2) Time

Time is a python based library which contains the method for recording the time. We have used to calculate the fps.

3) Pyautogui

PyAutoGUI is a cross-platform GUI automation Python module for human beings. Used to programmatically control the mouse & keyboard. pip install pyautogui. We have used it for executing the navigation operation based on classified gesture.
Working
The main driver module has the class called Gesture controller. The Gesture controller class classify the gesture and perform the OS navigation operation accordingly.
We classify the hand gesture by identifying which fingers of hand are up and which motion the hand is performing.
Which fingers are currently up is processed by the method called fingersUP() from tracking module based on the data generated by mediapipe. And we stored these data in the form of array which contain 0 and 1.
Eg:[0,1,0,0,0] ----------> This means index finger is up. After we get the fingers data we execute the navigation operation.
For executing the navigation operation, we are using the pyautogui library. This library provides the methods like left click right click or any keyboard shortcut, etch from which we can interact with system.

a) Video Stream Acquisition

We are using the OpenCV to get access of the system webcam. Using it we capture the video feed frame by frame. The frames are sent as parameters to the hand detection function for detecting if the image contains hand or not. If then it generates the data related to that particular image.

b) Hand Detection

Once an image frame is acquired using the hand detection function we search for user's hand. After detection the function generates the related data for tracking. Hand landmarks are the predefined points located on hand at different locations. Using these points, we can track the hands location as well as the location of individual fingers.

c) Landmarks Position

Once the landmarks have been generated, we need to extract the x and y coordinates of each point which are located on the hands. In this stage we also try to track the hand's movement. Using OpenCV we show the visual representation of the hand tracking,

d) Gesture Classification

After generating all the required data from the above stages. We start with the classification of the gesture. Here we classify the gesture which is currently begin performed by the user in that captured image frame.

Based on which fingers have been raised up we try to classify the gesture. For example: [0,1,0,0,0] here the array of size 5 represents the different individual finger. Value at index 1 it represents the index finger and value 1 denote that index finger is raised up and value 0 denote that the finger is folded down.

e) System Navigation Operation Execution

After classifying the gesture, we execute the corresponding system's navigation operation. For example, raising the only the index finger system will execute system's navigation operation like cursor movement based on the movement of the user's hand. For executing the system's navigation operation, we are using the autopy and pyautogui library.

IV. CONCLUSION AND FUTURE SCOPE

Through this review paper we have attempted at exploring various ways that can be used in recognising hand gestures made by humans. It has been observed that there is a fundamental common technique employed for this purpose – artificial neural networks. An advanced type of artificial neural networks, namely convolutional neural network serves as the main model used to process visual imagery. Although the base technique used by all the respective authors mentioned in this review paper is the same, there are considerable differences in the results obtained. This is due to the fact that despite of a similar base model, the overall approach used differs, changing the entire direction of how the working proceeds. Also, with respect to the concept of contactless system control using virtual mouse, the technique has been found to be quite economic and computationally light weight, not requiring the use of a dedicated graphics processing system for its functioning. Although the research mentioned in this review paper has produced commendable results, there remains a scope for improvement on various aspects.

V. ACKNOWLEDGMENT

We would like to express our sincere gratitude to all those who have contributed to the development and completion of our research project on contactless system navigation using dynamic hand gesture recognition. First and foremost, we would like to extend our deepest appreciation to our supervisor, Mrs.- S. S. Raskar, for their guidance, support, and valuable insights throughout the entire research process. Their expertise and encouragement have been instrumental in shaping our project and pushing us to excel.

We are also thankful to the faculty members of Computer Engineering at SPPU for providing us with the necessary resources, facilities, and opportunities to conduct our research. Their commitment to fostering an environment conducive to learning and innovation has been invaluable.

We would like to express our heartfelt thanks to our research team members for their collaborative efforts and dedication. Each team member has played a crucial role in carrying out experiments, collecting data, and analyzing results. Their commitment and hard work have contributed significantly to the success of this project.

REFERENCES

[1] Wenjin Zhang and Jiacun Wang, "Dynamic hand gesture recognition based on 3d convolutional neural network models" in 2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), pages 224–229. IEEE, 2019.

[2] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool, "Temporal segment networks for action recognition in videos", in IEEE transactions on pattern analysis and machine intelligence, 41(11):2740–2755, 2018.

[3] Yuecong Min, Yanxiao Zhang, Xiujuan Chai, and Xilin Chen, "An efficient pointlstm for point clouds based gesture recognition", In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5761–5770, 2020.

[4] Hao Tang, Hong Liu, Wei Xiao, and Nicu Sebe. "Fast and robust dynamic hand gesture recognition via key frames extraction and feature fusion", Neurocomputing, 331:424–433, 2019.

[5] Yuxiao Chen, Long Zhao, Xi Peng, Jianbo Yuan, and Dimitris N Metaxas, "Construct dynamic graphs for hand gesture recognition via spatial-temporal attention", arXiv preprint arXiv:1907.08871, 2019

[6] Wenjin Zhang, Jiacun Wang, and Fangping Lan, "Dynamic hand gesture recognition based on short-term sampling neural networks", IEEE/CAA Journal of Automatica Sinica, 8(1):110–120, 2020.

[7] Hung-Yuan Chung, Yao-Liang Chung, and Wei-Feng Tsai, "An efficient hand gesture recognition system based on deep cnn", in 2019 IEEE International Conference on Industrial Technology (ICIT), pages 853–858. IEEE, 2019.

[8] Md Abdur Rahim, Abu Saleh Musa Miah, Abu Sayeed, and Jungpil Shin, "Hand gesture recognition based on optimal segmentation in human-computer interaction", in 2020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII), pages 163-166, IEEE, 2020.

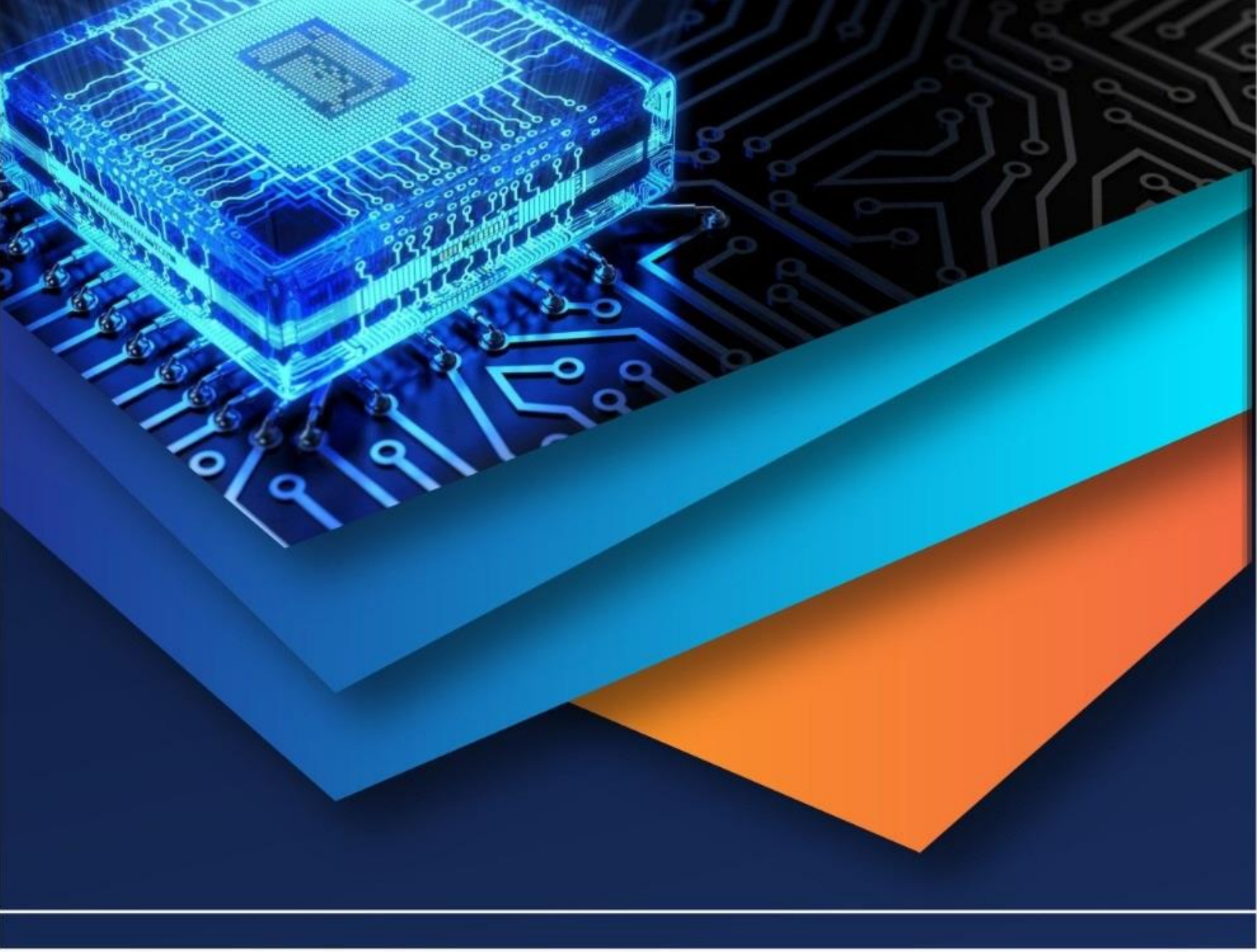
[9] Jun Xu, Hanchen Wang, Jianrong Zhang, and Linqin Cai, "Robust hand gesture recognition based on rgb-d data for natural human-computer interaction", IEEE Access, 2022.

[10] Debajit Sarma and MK Bhuyan. Hand gesture recognition using deep network through trajectory-to contour based images. In 2018 15th IEEE India council international conference (INDICON), pages 1–6. IEEE, 2018.

[11] S. Sriram, B. Nagaraj, J. Jaya, S. Shankar, P. Ajay, "Deep Learning-Based Real-Time AI Virtual Mouse System Using Computer Vision to Avoid COVID-19 Spread" in Hindawi Journal of Healthcare Engineering, Volume 2021, Article ID 8133706, 2021.

[12] Dinh-Son Tran, Ngoc-Hyunh Ho, Hyung-Jeong Yang, Soo-Hyung Kim, Guee Sang Lee, "Real-time virtual mouse system using RGB-D images and fingertip detection", Multimedia Tools and Applications, 80:10473-10490, 2021.

[13] Kabid Hassan Shibly, Surat Kumar Dey, Md. Aminul Islam, Shahriar Iftekhar Showrav, "Design and Development of Hand Gesture Based Virtual Mouse", 1st International Conference on Advances in Science, Engineering and Robotics Technology 2019(ICASERT 2019).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY