iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI: https://doi.org/10.22214/ijraset.2021.35742

Machine Learning Models for Finger Bend Evaluation using Implemented Low cost Flex Sensor

Pratyush Kaware[1], Vedant Kayande[2], Aditya Khambete[3], Nida Bakereywala[4]
[1, 2, 3]*Electronics and Telecommunications Department,* [4]*Applied Science & Humanities Department (Mathematics), Sardar Patel Institute of Technology, Mumbai.*

Abstract: In this paper a cost-effective sensor has been implemented to read finger bend signals, by attaching the sensor to a finger, so as to classify them based on the degree of bent as well as the joint about which the finger was being bent. This was done by testing with various machine learning algorithms to get the most accurate and consistent classifier. Finally, we found that Support Vector Machine was the best algorithm suited to classify our data, using we were able predict live state of a finger, i.e., the degree of bent and the joints involved. The live voltage values from the sensor were transmitted using a NodeMCU micro-controller which were converted to digital and uploaded on a database for analysis.
Keywords: Machine Learning, flex sensor, Support Vector Machine

I. INTRODUCTION

The need for reading hand signals could be immense field of medicine or the chemical industry where it might be hazardous for humans to do on their own. These hand signals can then be used to emulate an artificial hand which would do the work in dangerous environments for us. There are various ways to read hand signals mainly image processing, in which input is in the form of images, or using some form Electromyography (EMG)[1] sensor that gives voltage readings based on muscle movements or other methods involving the use of accelerometers touched upon in the next section. Though each of these methods have their own flaws such as image processing is heavy on computing and EMG sensors or IMU sensors that measure acceleration are very expensive do have their fair share of noise which needs filtering.

In our method we have implemented a sensor similar to a flex resistor which changes its value of resistance based on the amount the bend on the sensor. We attached this sensor to a finger and obtained the voltage values depending on the bend of the finger. This method can be used to get readings of all the fingers by attaching a sensor to each. We trained a machine learning model to create classifier which gave us the joints which were bent and the degree of rotation. This classifier can then classify the readings quickly unlike the image processing method and is cost-effective as well. Though in our method there is restriction in predicting the curl of a finger only towards and away from the palm direction.

Moving onto the overview of our paper, Section II shows various ways finger movement tracking can be done and what devices can be used for it. We also touch upon various machine learning techniques that can be used for this. In Section III, all the components used have been shown in detail as well as the working of the implemented sensor has also been explained. In Section IV, the data procurement procedure as well as some data visualisation has been done. In Section V, various machine learning models have been used on the procured data and the results have been noted. In section VI, the results have been analysed and justified. Section VII is the conclusion.

II. LITERATURE SURVEY

There have been various ways of tracking finger movements most notably with expensive sensors such as an IMU sensor which detects the acceleration values in all 3 dimensions which in turn gives the yaw, pitch and roll. A glove can be made that can give an accurate measurement of finger movement using an IMU sensor on various joints [2]. Another way to do this is to use image processing which would then track the finger and its rotation although such methods require special cameras to get accurate enough readings [3]. Although such method does have an advantage of not needing to connect any sensor to the subject's finger. Finally, there is a way of doing this by using flex and stretch sensors which are commercially available that when stretched give a change in their output value based on the stretch. This can be attached to a finger and readings can be noted [4]. Although such commercial stretch sensors do provide a predictable change in their output depending on their stretch, low-cost flex resistor can be built whose output might not be as predictable but which can be decoded using various machine learning algorithms.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

There are no studies that combine the use of flex resistors and Machine learning for movement tracking. Flex resistors provide a cheap, lightweight, robust, and effective way for movement tracking [5]. The use of Flex resistors ranges from making artificial touch surfaces [6] to using it across in the medical fields for various applications. This paper mainly focuses on the use of Flex resistors to simulate a normal hand which could further develop many applications. As this paper explores the use of clustering in Machine learning algorithms and it being a fundamental part of driving the project, KNN [7] has been the one of the primitive choices as a classifier. The paper also explores other viable algorithms. In order to understand the basic mechanism of how flex resistors work in simulating an arm or a hand, [8] helps points this paper to a direction. It discusses how using flex sensors was a part of the design of a robotic arm. It was just a part of an elaborate setup including Zigbee and a voltage divider circuit.

III. HARDWARE COMPONENTS

A. Flex Sensor

We made the flex sensor using two pieces of aluminium foil and graphite paper where, the graphite paper was sandwiched between the two pieces making sure they did not touch each other as shown in Fig. 1. The two pieces had the length to cover the three joints of the index finger: Metacarpophalangeal joint, Proximal Interphalangeal joint and the Distal Interphalangeal joint.

Fig. 1 Flex Resistor

When there is no bent on the sensor the value of resistance between the two outer aluminium foils is very high. The flex resistor changes its resistance depending on the bent of all the three layers. As the three layers get close to each other as shown in Fig. 2, the value of the resistance between the two conductors decreases.

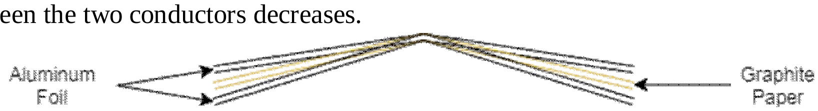


Fig. 2 Bent Flex Resistor

If we apply a constant voltage to the flex resistor and connect the other end to an Analog to digital converter, we can get changes in the voltage depending on the bent. Although, the bending of a finger is more complex than shown in Fig. 3, some pattern can be found in the readings depending on how much the finger is bent.

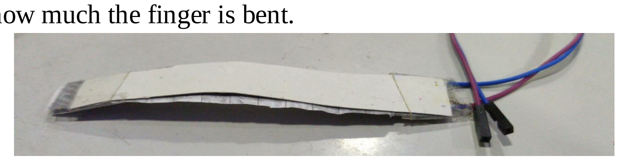


Fig. 3 Fabricated Flex Resistor

The above in Fig. 3 is the flex resistor we made. We substituted the graphite paper with a normal paper on which we scribbled on both sides with a pencil.

B. Micro-controller

We have used Node-MCU with ESP8266 WiFi chip (shown in Fig. 4) which also has a 10-bit Analog to Digital Converter (ADC).

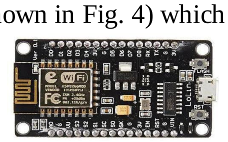


Fig. 4 ESP8266 micro-controller

The flex sensor was provided 3.3V voltage through the pins on the Node-MCU and the other end of the flex resistor was connected to the ADC pin of the micro-controller as shown in Fig. 5.

Fig. 5 Connections to Node-MCU

The readings were then uploaded to a Real-time Database using Google's Firebase as the database. These values were then used to train the Machine Learning model.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

IV. DATASET

The sensor was attached to the finger as shown in Fig. 6. The sensor needs to be attached tightly to the finger such as unwanted bends don't occur and when the finger is bent, the sensor also bends in a similar fashion. We also made markings as to where each joint is under the sensor so as to get consistent readings while re-attaching the sensor.

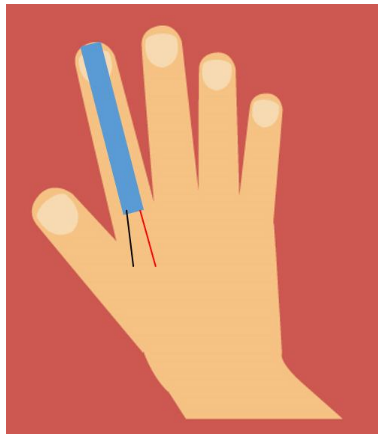


Fig. 6 Connections on finger

The red terminal of the flex sensor needs to be attached to a constant voltage, then the other end could be connected to the ADC pin of the micro-controller. The hold and sample time for the ADC was 2ms which is enough to get an accurate reading. The values were then mapped from 0 to 1024. This was then uploaded to the database using the NodeMCU's Wi-Fi module. The database we used for this project was Google's Firebase real-time database. The value database received a new value every 100ms.

For acquiring the training data, we first kept the finger in a Neutral state (as shown in Fig. 7-A) and took 100 readings of that state and then downloaded it from the database. For the next state, we moved the finger about the middle joint of the finger (as shown in Fig. 7-B) and then took 100 readings of this state. Similarly, we did this with both the joints involved (as shown in Fig. 7-C) and took 100 readings of this state.

(A) Neutral position (B) Curl around middle joint (C) Curling around both joints

Fig. 7 Finger Positions

The motion about the uppermost joint near the tip of the finger was not recorded as this motion is highly constrained in an average individual and the change in values in that state was insignificant. A significant change is observed only when the middle joint is already somewhat curled. These readings were then downloaded from the database log in json format. This data was then labelled for as N, M and F, where N refers to the Neutral readings while in state Fig. 7-A, M refers to readings taken while in states as shown in Fig. 7-B and F refers to readings take in states as shown in Fig. 7-C. All the values are integers i.e., no decimal point values.

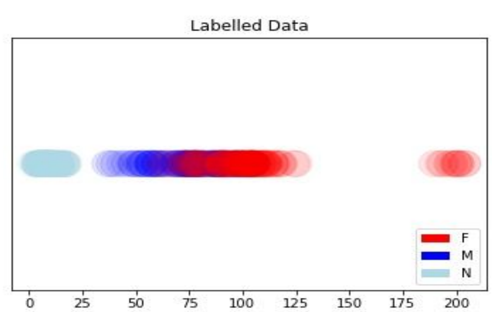


Fig. 8 Data plot

The minimum and maximum values for the data labelled N is 2 and 7. The minimum and maximum values for the data labelled M is 37 and 105. The minimum and maximum values for the data labelled M is 62 and 204. This shows that there exists an overlap in data labelled M and F in the range 62 to 105. The zoomed plot of this subset of data can be seen in Fig. 9.

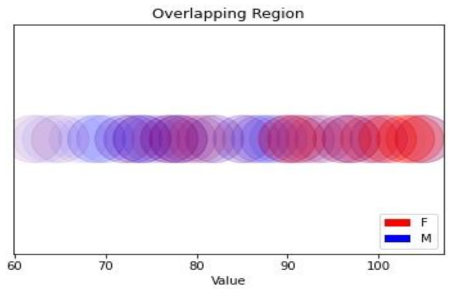


Fig. 9 Overlap plot

The data points are plotted with bigger symbols to emphasize the overlap, though even without them this proves that there is no definite range in which each label lies. This goes on the show us that the system is non-linear. So, to train based on these data we would need models that can classify non-linear systems. This non-linearity can be explained by the flex sensor not being taut in some areas becomes of the angle with the joint.

V. SELECTION OF MODEL

Our observations while training on these models and why we selected Support Vector Machine as our final model is explained below. The following models were tested with our dataset and gave the best results for our data, explained in detail below.

A. *Random Forest*

Random Forest is an algorithm which uses many decision trees in order to make a decision so as to minimize the number of misclassifications that may occur due to a decision trees not being consistent with new data. We used 10 estimators. This model gave us an accuracy of 81%.

Although this was an improvement over the Decision Tree model which had Accuracy of 79%, still the misclassification rate was high in this. From the confusion matrix shown in Table I, below it can be seen that the misclassification took place between M and F in the overlap area.

TABLE I

Confusion Matrix for Random Forest

Predicted Labels	True Labels		
	F	M	N
F	18	10	0
M	11	25	0
N	0	0	36

The columns labels represent true label and the row labels represent false labels. Moreover because of the small areas of a state, the decision boundary plot shown in Fig. 10 shows that there might be some over-fitting.

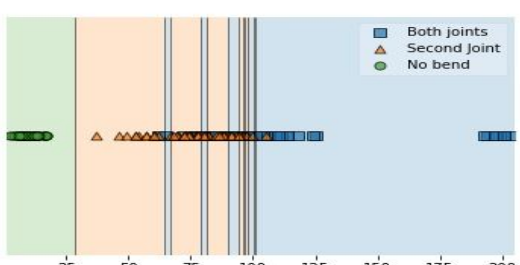


Fig. 10 Decision Boundary plot for Random Forest

B. K Nearest Neighbours

This model works by looking at the nearest values label to assign the label to a test value. This inherently works with non-linear data. We used 15 neighbours. This model gave us an accuracy of 84%. This has better accuracy than RF model. From the confusion matrix shown in Table II below it can be seen that the misclassification between M and F is very less than the RF model.

TABLE III

Confusion Matrix for K Nearest Neighbours

Predicted Labels	True Labels		
	F	M	N
F	20	11	0
M	5	26	0
N	0	0	28

The columns labels represent true label and the row labels represent false labels. Although the accuracy is high, there is still a small region indicating some over-fitting in the decision boundary plot shown in Fig. 11. The overfitting goes away with increasing the number of neighbours but the accuracy goes down drastically.

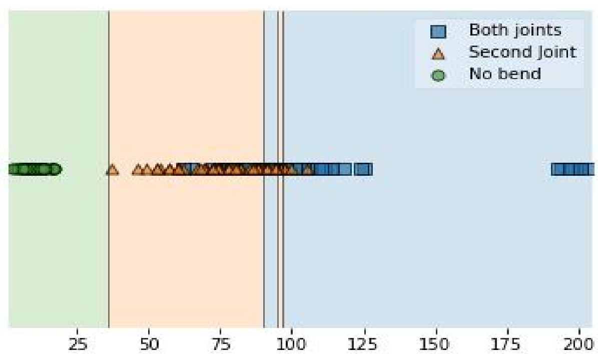


Fig. 11 Decision Boundary plot for K Nearest Neighbours

C. Support Vector Machine

Support vector machine model works by creating hyperplanes that classify the data based on labels. But the linear kernel would not work for our data as there is some overlap. So, we used Radial Basis Function kernel which can be used to classify non-linear data. This method works by creating a soft margin by calculating the misclassifications around that margin. The RBF kernel is similar to a weighted KNN model. This model gave us an accuracy of 85%.

From the confusion matrix shown in Table III, below it can be seen that the misclassification between M and F is very less similar to the KNN model.

TABLE IIIII

CONFUSION MATRIX FOR SUPPORT VECTOR MACHINE

Predicted Labels	True Labels		
	F	M	N
F	20	8	0
M	9	21	0
N	0	0	32

The columns labels represent true label and the row labels represent false labels. Although the accuracy is similar to the KNN model, this is a better model as there is no overfitting which can be observed from the boundary plot shown in Fig. 12 as it has a clean boundary. This shows that this model will work with novel data as well.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

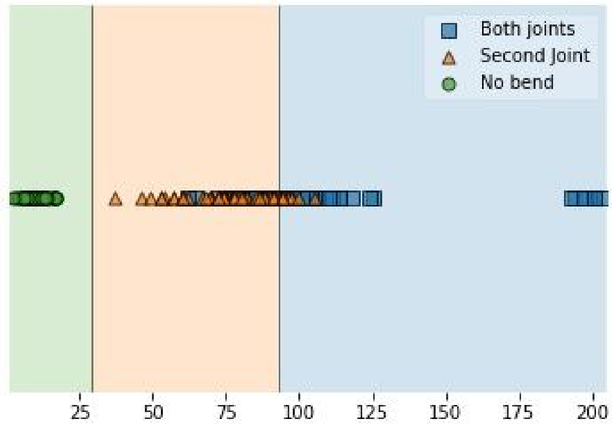


Fig. 12 Decision Boundary plot for Support Vector Machine

The decision boundary plot shown in Fig. 12 shows that the states have a clean boundary between each other. This boundary helps us calculate the degree of rotation of the finger by first classifying the value in label and then dividing the value obtained by the range of the label to get the degree of rotation. For e.g., the range of M or middle joint is 29 to 93 as obtained from the boundaries using the SVM model. So, for the value of 67 the degree of rotation would be 0.59 of state M.

VI. RESULT ANALYSIS

The results by training with 10-fold cross validation with for the 3 models with the best results can be seen in Table IV. The results show that the best results are obtained using Support Vector Machine model with rbf kernel with an accuracy and f1 score of 0.85.

TABLE IVV
MODEL COMPARISON

Model	Results		
	Accuracy	Misclassification	F1 score
Random Forest	0.81	0.19	0.83
K Nearest Neighbours	0.84	0.16	0.84
Support Vector Machine	0.85	0.15	0.85

The Area under the curve in Receiver Operating characteristics for neutral state N was 1 for the models. So, the models were mainly differentiated based on their ROCs for M and F states or middle joint of both joints state. From the ROCs in Fig. 13 we can see that Support Vector Machine model has the best AUCs.

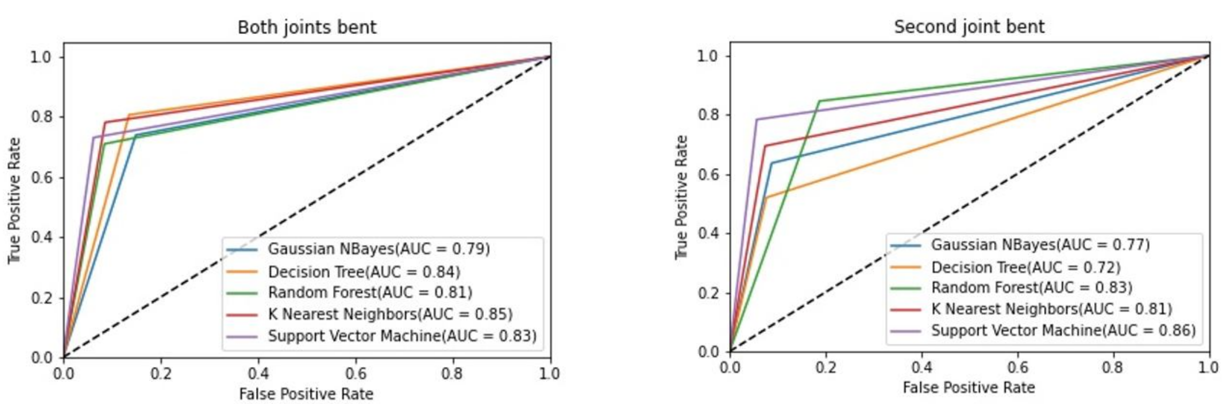


Fig. 13 ROC plots for the models

VII. CONCLUSION

A cost-effective flex sensor can be implemented which can be used to measure bend angle with a good enough efficiency using machine learning techniques. Various models were built and tested to find the best model which is most suitable for the data from the implemented flex sensor. Support Vector Machine Model gave the best results for the data procured. Although there might be some limitations using this sensor, mainly the smallest degree of angle bend that the model can detect or can be detectable. These results can be further improved using deep learning models.

REFERENCES

[1] K. Ishikawa, M. Toda, S. Sakurazawa, J. Akita, K. Kondo and Y. Nakamura, "Finger Motion Classification Using Surface-Electromyogram Signals," 2010 IEEE/ACIS 9th International Conference on Computer and Information Science, 2010, pp. 37-42, doi: 10.1109/ICIS.2010.131.

[2] J. Connolly, J. Condell, B. O'Flynn, J. T. Sanchez and P. Gardiner, "IMU Sensor-Based Electronic Goniometric Glove for Clinical Finger Movement Analysis," in IEEE Sensors Journal, vol. 18, no. 3, pp. 1273-1281, 1 Feb.1, 2018, doi: 10.1109/JSEN.2017.2776262.

[3] L. Sbernini, A. Pallotti and G. Saggio, "Evaluation of a Stretch Sensor for its inedited application in tracking hand finger movements," 2016 IEEE International Symposium on Medical Measurements and Applications (MeMeA), 2016, pp. 1-6, doi: 10.1109/MeMeA.2016.7533809.

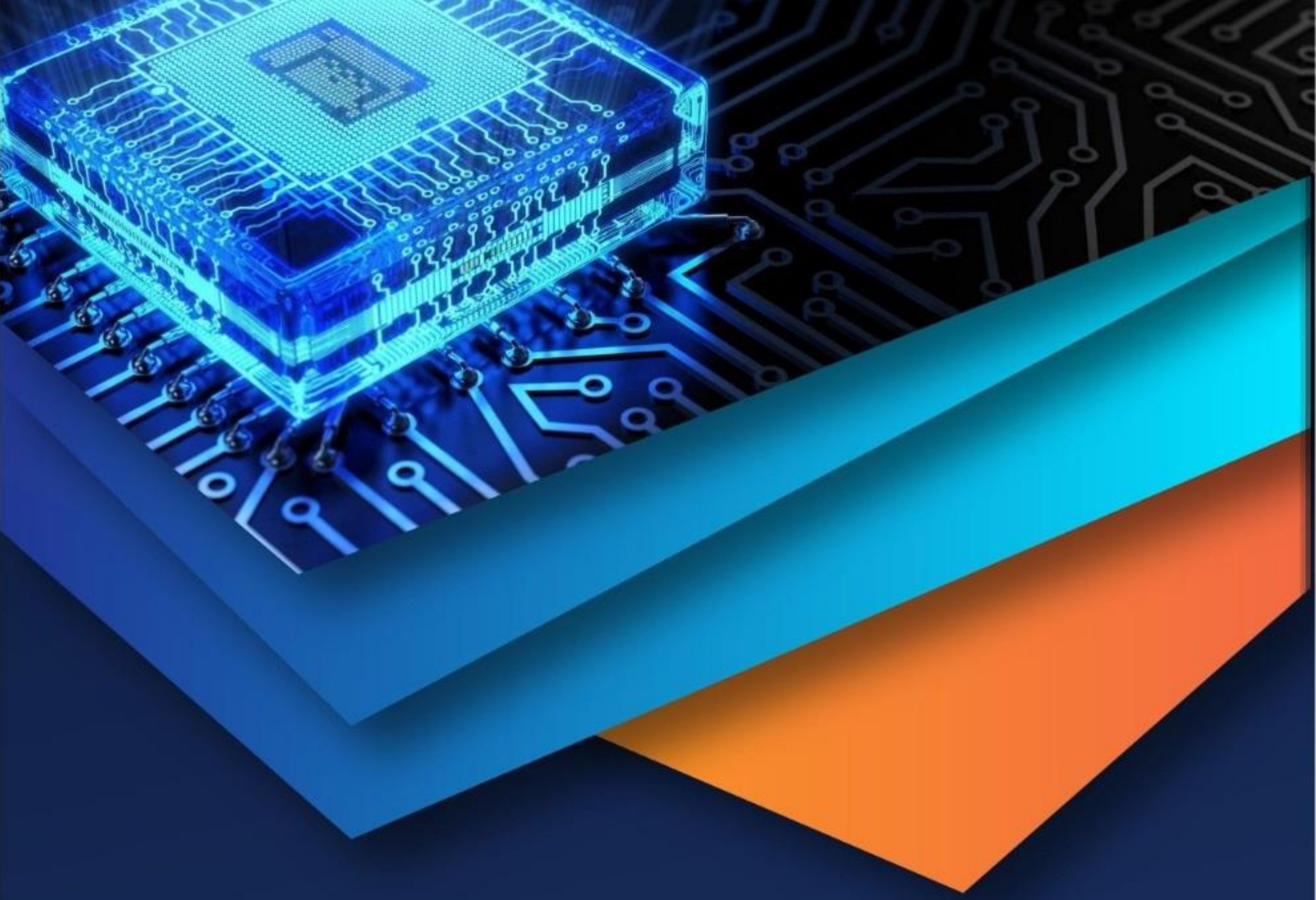
[4] Togootogtokh, E., Shih, T.K., Kumara, W.G.C.W. et al. 3D finger tracking and recognition image processing for real-time music playing with depth sensors. Multimed Tools Appl 77, 9233–9248 (2018). doi: 10.1007/s11042-017-4784-9.

[5] Saggio, Giovanni & Riillo, Francesco & Sbernini, Laura & Quitadamo, Lucia. (2016). Resistive flex sensors: A survey. Smart Materials and Structures. 25. 013001. 10.1088/0964-1726/25/1/013001.

[6] Dahiya, Menal. (2017). Fingertip Infotainment: A Way to Secure Information. International Journal of Advanced Research in Computer Science. 8. 1068-1071.

[7] H. Yigit, "A weighting approach for KNN classifier," 2013 International Conference on Electronics, Computer and Computation (ICECCO), 2013, pp. 228-231, doi: 10.1109/ICECCO.2013.6718270.

[8] Y. Pititeeraphab and M. Sangworasil, "Design and construction of system to control the movement of the robot arm," 2015 8th Biomedical Engineering International Conference (BMEiCON), 2015, pp. 1-4, doi: 10.1109/BMEiCON.2015.7399564.



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY