Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)

Volume 6 –Issue 2,August 2023
Paper:74

Statistical Depth-Aware Dynamic Vision Feature Selection for hand Gesture Recognition using Xception Transfer Learning

Jivitesh R
Department of Computer Science Engineering
SRM Institute of Science and Technology
Chennai, India
jr3671@srmist.edu.in

Praveen V
Department of Computer Science Engineering
SRM Institute of Science and Technology
Chennai, India
pv5972@srmist.edu.in

Keerthana R
Department of Computer Science Engineering
SRM Institute of Science and Technology
Chennai, India
kr6910@srmist.edu.in

Ms.J.Gowthamy
Assistant Professor
Department of Computer Science Engineering
SRM Institute of Science and Technology
Chennai, India
gowthamj@srmist.edu.in

***Abstract*—Deaf individuals communicate with each other through their own country's unique sign language. Researchers have conducted numerous studies on recognizing these sign languages, and have developed Sign Language Recognition (SLR) systems that aim to convert input signs into text/speech. These systems help bridge the communication gap between the hearing-impaired and the rest of society. However, existing SLR systems are limited to recognizing isolated sign gestures and utilize Takagi Sugeno-kang (TSK) or Sugeno fuzzy inference systems. These systems extract shape features of hand gestures using elliptical Fourier descriptions and principal component analysis (PCA), and utilize a Convolution Neural Network (CNN) to extract spatial features from signed sequences for recognition. CNNs are a type of deep feed-forward neural network that require minimal preprocessing and include convolutional, pooling, fully connected, and normalization layers. By utilizing CNNs, the proposed SLR system aims to improve the accuracy and efficiency of recognizing sign language.**

Keywords— SLR, TSK, Convolution Neural network, improve accuracy.

I. INTRODUCTION

Similar to spoken languages, each country has its own unique deaf sign language that is used by deaf individuals to communicate with each other. There have been numerous research studies conducted on the recognition of deaf sign language. Sign language serves as a means of communication between people with hearing impairments and the rest of society. Researchers have developed several Sign Language Recognition (SLR) systems, but these systems are currently only capable of recognizing individual sign gestures in isolation.

The primary objective of Sign Language Recognition (SLR) is to create an assistive technology that can convert signed inputs into text or speech automatically. This technology plays a crucial role in minimizing communication barriers between people with hearing impairments and the rest of society. Currently, the most commonly used SLR system is based on the Takagi-sugeno-kang (TSK) inference system, which utilizes linear or constant output membership functions. The Sugeno fuzzy inference system involves five main steps, including fuzzification of input variables, application of fuzzy and or or operators, calculation of rule weights, determination of output level, and finally, defuzzification.

To extract shape features of hand gestures, elliptical Fourier descriptions are utilized, which significantly reduces the feature vectors for an image. Principal component analysis (PCA) is also implemented to further minimize the feature vector for a given gesture video. Additionally, the extracted features are not affected by scaling or rotation of gestures within a video, making the system more adaptable. The proposed system is built using a Convolution Neural Network (CNN).

The proposed system utilizes a Convolutional Neural Network (CNN) to extract spatial features from signed sequences, which are then processed by a modified model for recognition purposes. CNNs are a type of deep feed-forward neural network that require minimal preprocessing. These networks are known for their shared-weights architecture, which makes them shift or space invariant. CNNs typically have an input layer, an output layer, and several hidden layers, such as normalization,

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)

pooling, convolutional, and fully connected layers.

II. EXISTING SYSTEM

A hygienic and safe choice, especially during the current global pandemic, gesture recognition has grown in popularity as a mode of interaction since it allows users to avoid touching surfaces. Despite much research into gesture recognition, the traditional keyboard and mouse remain the most used forms of interaction.To recognise gestures more accurately and quickly, it is tremendously advantageous to be aware of new developments in artificial intelligence. A number of fields, including artificial intelligence, are incorporating deep learning to improve performance.

This shows that gesture recognition might someday be a practical choice for regular user engagement. Therefore, the primary goal of this essay is to significantly advance this process.In light of the aforementioned, our research has examined 571 scientific papers on artificial intelligence and gesture recognition. With the help of this study, we have acquired vital data on the output of authors, pertinent publications, and important pieces on the subject. Additionally, we have created our own model to illustrate how various gesture recognition methods and artificial intelligence strategies used in this field are interconnected. Recent years have seen a significant increase in the use of gesture recognition in a variety of scientific sectors, including education, virtual reality, cars, and sign language translation, among others.

The incorporation of AI approaches has had a substantial impact on the advancement of gesture recognition. This prompted the start of the current study, which attempts to collect relevant data on the most important trends in research on AI approaches used for gesture detection throughout the previous two decades (2000-2020). According to our data, there has been a noticeable upscaling in the number of citations and articles on this topic since 2016, which was previously ignored. The academic institutions from China have shown a larger propensity towards research on gesture-recognition and Artificial Intelligence, and have significantly contributed to this subject, which is a notable finding of our investigation. This may be related to the widespread use of deep learning techniques in several fields, including computer vision. Given the strong relationship between gesture recognition and computer vision, particularly vision-based gesture recognition, this combination has significantly advanced both computer vision and gesture recognition. Therefore, it is not surprising that research on this subject has recently increased in order to achieve positive results using AI techniques. About 571 articles from the scientific databases WoS and Scopus were evaluated as part of our bibliometric analysis.The study requirements placed a particular emphasis on the employment of AI techniques in the creation of various types of gesture recognition combined with machine-learning and deep-learning approaches. Based on this data, we have highlighted the key features of the scientific literature on this subject, beginning in part III with a description of the methodology used for this bibliometric analysis and concluding in section IV with a discussion of the findings. The major indicators that were examined are presented in Section IV of the study, including the number of articles published over time, the journals with the most articles, the authors with the most citations, and the nations with the most articles and highest h-index. According to our research, the *Applications* journal has published the most publications on this topic. The highest rates of productivity have been found in China and the United States, Chinese institutions providing the majority of publications on the topic overall. However, a study on conscious gesture-recognition techniques by J. J. Ojeda-Castelo et al. found that South Korean authors make up the majority of authors with more articles published. A snippet of each of the 10 articles with major citations has also been included, along with the keyphrases that are most helpful when looking for terms associated with this work. According to our models, face gestures have been employed in the research the least, followed by hand gestures.Recurrent networks and CNN are the most often utilized techniques in DL, with CNN being the technique most typically used for both hand and face motions. The fact that this technique has been employed for both sorts of visible motions makes it one of the rare ones.In the ML model, where it has been used the most frequently in works that include facial expressions, the SVM technique has stood out. Researchers have also praised the CNN and SVM combination highly. Despite the research, there are still a number of issues to be resolved before gesture detection with AI systems is practical. However, the improvements are encouraging, as can be seen from the summary snippets of the 10 articles that have been a part of this analysis with the majority of citations.

III. PROPOSED SYSTEM

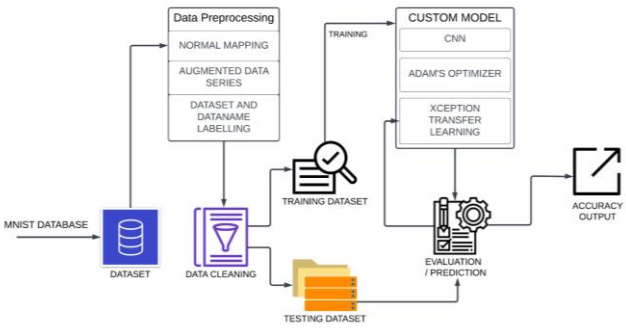

Fig 1 System Architecture

In the proposed system, Datasets include duplicate information like finger and wrist joints. Only pertinent data are kept once this superfluous information has been deleted. A few preprocessing procedures are used to regulate the diversity in hand spans, palm sizes, and movements. We have retrieved dynamic information for both hands from each signer who makes the sign gesture. The fingertip coordinates make up each feature. These traits are individually retrieved from the collected data. Since it is crucial for use as an educational tool, we tested the machine learning component in various environments as a learning mechanism to determine how it behaves in various settings. This application may easily integrate more signals with the right training. If the user may choose the alphabet range he wants to study,

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 611X)

usability will be increased. Developing a user-friendly app that takes account will enable him to pause and resume as needed. This will aid in understanding simple or complex signs based on general facts. Additionally, the model's classification accuracy threshold can be adjusted for each sign individually rather than worldwide.

The objective of feature engineering in the proposed system is to eliminate unnecessary dimensions and reduce the number of feature space dimensions without sacrificing the essential information so that we can represent each piece of data simply and reduce the computational load. Because each dimension discovered by feature engineering is independent and distinct, there is no meaningful relationship between any given dimension and any given component. The curse of dimensionality can be avoided by reducing the dimension of the feature space. Thus, a small cluster of training data is required during the training phase, which can shorten the training period and enable real-time training. This group includes CNNs, or deep feed-forward artificial neural networks. With minimal preprocessing objectives, CNN uses a multilayer perceptron version.

These networks also go by the names shift or space invariant because of their shared-weights architecture. Normalisation, convolutional pooling, and fully connected layers are just a few of the hidden layers that make up a CNN. It also often has an input layer, an output layer, and other layers. In this study, a one-dimensional CNN (D CNN) was employed for feature extraction.

IV. System Overview

The MNIST database we used consists of 37 datasets with 1500 images each which contains numbers, alphabets and sign gestures. Each image is generated into a 5x5 matrix and is augmented by reshaping,angle changing or rotating. The data and data name is stored in a series which is shuffled for training and testing. After training and testing, the prediction is done by reverse mapping the new shuffled series and the dataset series. The comparison has arrived at an accuracy score of 87.6%.

The usage of CNN Model and Adams Optimizer together reduces the loss and increases the accuracy. To achieve this, the whole process of training and testing is repeated but instead of starting from scratch, Adams optimizer uses the knowledge of previously trained models and gets trained from them. By this way, the accuracy score of 87.6% can be increased to 93%.

From this we can conclude that by using Xception transfer learning protocols and developing a custom model, the accuracy can be increased.

V. Module Implementation

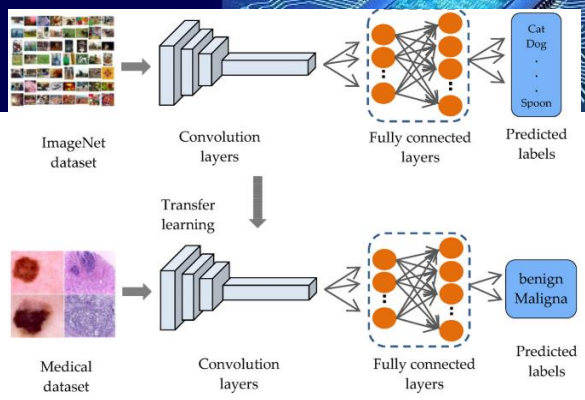

Fig 3 Xception transfer Learning

A variation of the Inception architecture called Xception transfer learning makes use of depth wise separable convolutions to achieve excellent accuracy with fewer parameters than previous models.Using a model that has already been trained on a big dataset, such as ImageNet, and then fine-tuning it on a smaller dataset for a particular task is known as transfer learning. Because it has achieved great accuracy on ImageNet and can be tailored for a range of tasks, Xception is a well-liked architecture for transfer learning.
These steps can be used to do transfer learning with Xception:

Pre-trained weights from Keras applications are loaded into the pre-trained Xception model. The model's pre-trained layers should be frozen to prevent updating while being trained.

Over the previously trained layers, add a fresh, completely connected layer.

Using your own dataset, train the model by starting with a low learning rate and then gradually raising it.

Make any necessary changes to the model and training settings after evaluating the model's performance on a validation set.

2. 1-Dimensional CNN

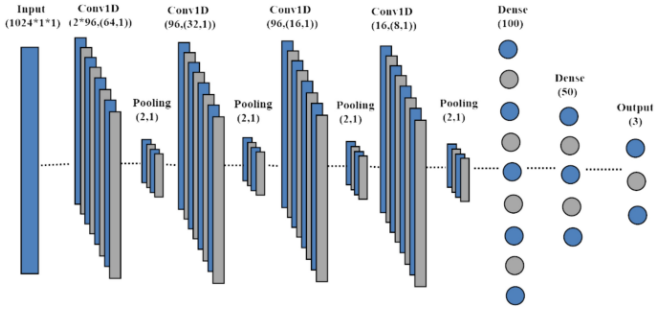

Fig 4 1-Dimensional CNN

A dimensional CNN (Convolutional Neural Network) is a type that is specifically designed to work with multidimensional data, such as images or videos. It is also known as a 3D CNN or a spatiotemporal CNN.

While a traditional 2D CNN is designed to work with two-dimensional data, such as images, a dimensional CNN extends the idea to work with three-dimensional data, such as

Journal of Current Research in Engineering and Science
JCRES
Bi-Annual Online Journal (ISSN : 2581 - 611X)

video frames or 3D images. It uses the concept of 3D convolution to extract spatiotemporal features from the input data.

The additional dimension in a 3D convolutional layer is that the latter considers the input data as a three-dimensional volume, where each slice of the volume corresponds to a frame in a video or a 2D image at a certain point in time. The 3D convolution operation takes into account both the spatial and temporal information, enabling the network to learn spatiotemporal features. Convolutional, pooling, and fully linked layers are only a few of the many layers that make up a dimensional CNN. In order to siphon features from the input data, the convolutional layers conduct the 3D convolution operation. The feature maps' spatial dimensions are reduced while their temporal details are preserved by the pooling layers. The ultimate output for classification or regression comes from the fully linked layers. In applications like action identification in movies, medical image analysis, and autonomous driving, where both spatial and temporal information is essential for effective prediction, dimensional CNNs have been frequently used. However, they are computationally intensive and require large amounts of data for training.

3. Adams Optimizer

Adams optimizer is a partnership of AdaGrad (Adaptive Gradient) and Root-Mean-Square-Propagation (RMSprop) algorithms. It helps to compute the adaptive learning rates for each parameter based on the moving average of both the 1st and 2nd moments of the gradients, which results in better convergence and faster optimization.

```
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])

no_of_epochs = 3

history=model.fit(datagen.flow(trainx,trainy,batch_size=32),validation_data=(testx,testy),epochs=no_of_epochs)

Epoch 1/3
1041/1041 [==============================] - 359s 334ms/step - loss: 1.0379 - accuracy: 0.6720 - val_loss: 0.4244 - val_accurac
y: 0.8625
Epoch 2/3
1041/1041 [==============================] - 346s 333ms/step - loss: 0.5187 - accuracy: 0.8257 - val_loss: 0.3766 - val_accurac
y: 0.8769
Epoch 3/3
1041/1041 [==============================] - 345s 332ms/step - loss: 0.4284 - accuracy: 0.8556 - val_loss: 0.3253 - val_accurac
y: 0.8843
```

Fig 5 Adam's Optimizer

$$
\begin{aligned}
v_t &= \beta_1 * v_{t-1} + (1-\beta_1) * g_t \\
s_t &= \beta_2 * s_{t-1} + (1-\beta_2) * g_t^2 \\
v_t^{corrected} &= v_t / (1-\beta_1^t) \\
s_t^{corrected} &= s_t / (1-\beta_2^t) \\
w_t &= w_{t-1} - \alpha * v_t^{corrected} / (sqrt(s_t^{corrected}) + \epsilon)
\end{aligned}
$$

Fig 6 Adam's Optimizer Formula

The optimization algorithm utilized is "Adam," which is well-known for its quick convergence rate and strong generalization capabilities. "Accuracy," a common statistic for classification issues, is the metric used to evaluate the model's effectiveness. We can improve the training process and make sure the model performs effectively on the given job by building the model with the proper parameters.

4. AdaGrad algorithm

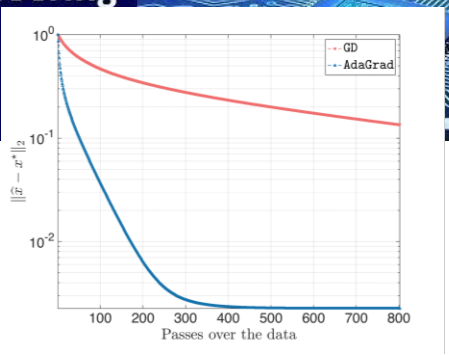

Fig 7 AdaGrad

The minima of an objective function are found using the optimisation approach of gradient descent by following the negative gradient of the function.

The fact that gradient descent employs the same step size (learning rate) for all input variables is a drawback. On objective functions with varying degrees of curvature in different dimensions, this can be a problem and may necessitate a different size step to a new point.

The adaptive gradients optimisation algorithm, also known as AdaGrad, is an extension of the gradient-descent optimisation algorithm that enables the step-size of each dimension to be automatically adjusted with respect to the gradients seen for the varying partial derivatives observed throughout the search process.

It aims to either compound the optimisation process by, for example, depleting the function evaluations required to reach the optima, or to improve the functionality of the optimisation method by, for example, resulting in a better result.

5. Root Mean Square Propagation

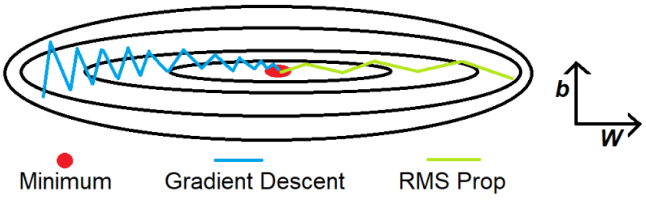

Fig 8 Root Mean Square Propagation

For training Artificial Neural Networks (ANN), the Root-Mean-Square propagation (RMSProp) optimization method was created. Adaptive learning rate approaches, which have become more and more popular recently, include RMSProp. It is a development of the Adam algorithm's Adam algorithm, momentum technique, and stochastic gradient descent (SGD) algorithm. The stochastic method for mini-batch gradient descent is one use for RMSProp.

6. DenseNet201 Module

```
def create_densenet_model():
    densenet_model = DenseNet201(input_shape=(50,50,3),include_top=False,weights='imagenet',pooling='avg')
    densenet_model.trainable = False
    inputs3 = densenet_model.input
    x = Dense(128, activation='relu')(densenet_model.output)
    outputs3 = Dense(len(folders), activation='softmax')(x)
    model = Model(inputs=inputs3, outputs=outputs3)
    return model
```

Fig 9 DenseNet201 Module

DenseNet-201 consists of 201-layers of convolutional neural networks. A network that has been pre-trained by utilizing upwards of a million images available in the ImageNet database. The pretrained network is able to cluster

images into different object categories. Because of this, the network now offers full feature-representations for a number of images.

7. ImageDataGenerator Module

```
def imageAugmentor():
    data_generator = ImageDataGenerator(rotation_range=180)
    plot(data_generator)

    data_generator = ImageDataGenerator(featurewise_center=False,
                                        width_shift_range=0.65)
    plot(data_generator)

    data_generator = ImageDataGenerator(featurewise_center=False,
                                        width_shift_range=0.65)
    plot(data_generator)

    data_generator = ImageDataGenerator(vertical_flip=True,
                                        zoom_range=[0.2, 0.9],
                                        width_shift_range=0.2)
    plot(data_generator)

    data_generator = ImageDataGenerator(horizontal_flip=True,
                                        zoom_range=[1, 1.5],
                                        width_shift_range=0.2)
    plot(data_generator)

    data_generator = ImageDataGenerator(width_shift_range=[0.1, 0.5])
    plot(data_generator)

    data_generator = ImageDataGenerator(zoom_range=[1, 2], rotation_range=260)
    plot(data_generator)
```

Fig 10 ImageDataGenerator Module

The Keras image data generator is used in the field of real-time data augmentation to create batches that contain the data from tensor pictures. We can loop through the data in batches when using the picture data generator provided by Keras. The picture data generator class has a number of methods and arguments that aid in defining the data generation's behavior.

Image data enhancement is a technique that transforms old photos into new ones. You can achieve this by making a few minor adjustments to them, such as changing the image's brightness, rotating it, or moving the subject horizontally or vertically.

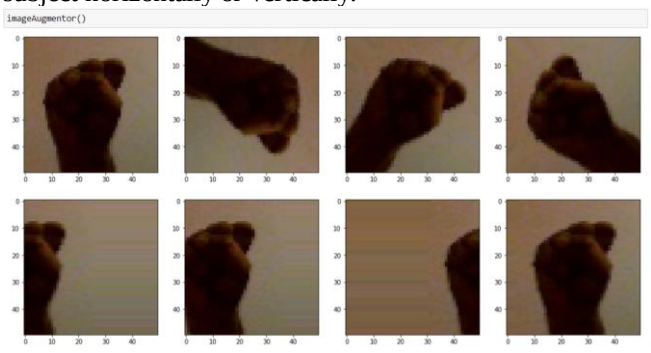

Fig 11 ImageAugmentor

Using image augmentation techniques, you may artificially expand the size of your training set and give your model considerably more data to work with. By making your model better able to recognise novel variations of your training data, you can increase the accuracy of your model.

Vertical shift, Horizontal shift, Vertical flip, Horizontal flip, Rotation, Brightness modification, and

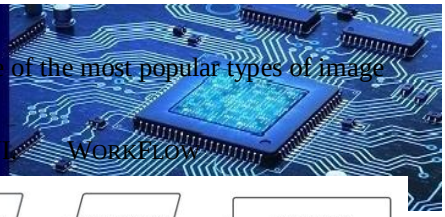

Fig 12 One example of the most popular types of image augmentation.

VI. WORKFLOW

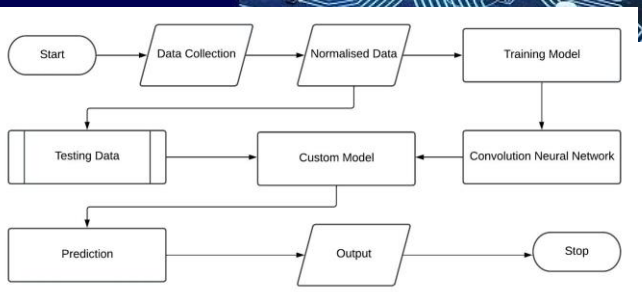

Fig.Workflow diagram

1. *Data Collection*

The data collection module of the hand-gesture recognition project is responsible for acquiring and pre-processing the data required for the training and testing of the machine learning model. The data is an amalgamation of 37 subsets of gestures varying from 0-9 and A-Z, each subset consists of 1500 images which correlate to that specific gesture.

The collected data is then pre-processed to remove any noise or irrelevant information that may affect the accuracy of the machine learning model. The pre-processing steps may include resizing the images, normalizing the pixel values, and augmenting the data to increase the diversity of the dataset. The data collection module is critical to the success of the project as the accuracy of the machine learning model depends on the quality and diversity of the dataset. A well-designed data collection module ensures that the model is trained on a diverse and representative dataset that reflects the real-world scenarios.

2. *Data Normalisation / Preprocessing*

The data is then processed, normalised and the process of normal mapping takes place. Here the data is stored in a dictionary for further processing. Image is generated in a 5x5 matrix and image augmentation factors like reshaping, angle, rotation etc are equipped. The cluster of augmented images are clustered together as data and data name in a series.

An important phase of every machine learning research is data preprocessing. The loaded data in this project is preprocessed to make it appropriate for the model's training. The data is then divided into features and labels. The names reflect the respective classes, whereas the features represent the input pictures. The whole series is copied into a new test file and shuffled for testing and training.

3. *Distortion Management*

To make sure that the system correctly records and decodes the user's hand motions, distortion management is an essential phase in the sign language converter process. Using gloves with sensors that precisely record the hand's movement in real time can help control distortion. Instead of depending just on visual signals, the gloves can offer a more precise picture of the hand's location and movement.Many sensors, including accelerometers and flex sensors, which

Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN : 2581 - 5113)

may monitor how the fingers are moving and where they are. Sensors may be included in the gloves. The relevant sign language motions can subsequently be identified by processing and analysing the data. Additionally, this method can help in lessening the effects of outside influences like changes in the environment or unintended hand gestures.

VII. RESULT

We have employed the transfer learning algorithm and the convolutional neural network in our hand-gesture recognition project to attain the maximum accuracy and precision (CNN). A pre-trained neural network is utilized as a commencing point for a new issue in the transfer learning approach. The goal is to apply the information gained from fixing one problem to another that is unrelated but still present. Results after training for 3 epochs consisting of 37 subsets show how well Xception transfer learning and CNNs perform and also emphasize the value of adopting data augmentation approaches to enlarge the dataset and avoid overfitting.

The usage of CNN Model and Adams Optimizer together reduces the loss and increases the accuracy. To achieve this, the whole process of training and testing is repeated but instead of starting from scratch, Adams optimizer uses the knowledge of previously trained models and gets trained from them. By this way, the accuracy score of 87.6% can be increased to 93%.

VIII. CONCLUSION

From this we can conclude that by using Xception transfer learning protocols and developing a custom model, the accuracy can be increased.

In conclusion, this module provides a potent method for precisely identifying hand gestures, which might be helpful in a variety of applications, such as human-computer interaction, sign language identification, and virtual reality.

IX. REFERENCES

[1] D. Kelly, J. McDonald, and C. Markham, A person independent system for recognition of hand postures used in sign language, Pattern Recognit.
[2] A. Tang, K. Lu, Y. Wang, J. Huang, and H. Li, A real-time hand posture recognition system using deep neural networks, ACM Trans. Intell. Syst.
[3] A. zaslan, M. Y. can, . zaslan, H. Tucu, and S. Ko, Estimation Mar. 2003.
[4] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240180.
[5] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. C. Kot, "Skeleton- Jun. 2020.
[6] J. Liu, X. Sheng, D. Zhang, J. He, and X. Zhu, "Reduced daily recalibra- tion of myoelectric prosthesis classifiers based on domain adaptation," 166176, 2016.
[7] C. Castellini and G. Passig, "Ultrasound image features of the wrist are linearly related to finger positions," in Intelligent Robots and Systems 2011 IEEE
[8] C. Castellini and D. S. Gonzalez, "Ultrasound imaging as a human," 2013.
[9] D. G. Lowe, "Object recognition from local scale-invariant features," Computer Vision, 2002. p. 1150.
[10] T. Furuya and R. Ohbuchi, "Dense sampling and fast encoding for 3d model retrieval using bag-of-visual features," in ACM International Conference on Image and Video Retrieval, 2009, p. 26. 1986.
[11] M. A. Oskoei, H. Hu. "Myoelectric control systemsA survey." Biomed. X. Liu, X. Zhai, W. Lu and C. Wu, "QoS-guarantee resource allocation for multibeam satellite industrial internet of things DOI: 10.1109/TII.2019.2951728, Nov. 2019.
[12] Q. Xiao, Y. Zhao and W. Huan, "Multi-sensor data fusion for sign language recognition based o n dynamic Bayesian network and convolutional neural network," Multimedia Nov. 2019.
[13] W. Jian and J. Roozbeh, "Orientation independent activity/gesture recognition using wearable motion sensors," Jul. 2018.
[14] J. B. Wen, Y. S. Xiong and S. L. Wang, "A novel two-stage weak classifier selection approach for adaptive boosting for 122-135, Sept. 2013.
[15] Z. Zhou, Y. Dai and W. Li, "Gesture recognition based on global template DTW for Chinese sign language," Journal of Aug. 2018.
[16] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems (pp. 91-99).
[17] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).
[18] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).
[19] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
[20] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
[21] Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6848-6856).
[22] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
[23] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015).

Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3), 211-252.

[24] Liu, X., He, P., Chen, W., & Gao, J. (2019). Multi-task deep neural networks for natural language understanding. arXiv preprint arXiv:1901.11504.

[25] Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10), 1499-1503.