SIGN LANGUAGE RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS

Ms. KOLUKULA KALYANI[1], Mr. ATCHUTANNA SAI NIRMAL[2], Ms. SIRAPARAPU ROHINI [3], Mr. MARASU BALA VIHAR[4], Mr. Dr T.V.S. SRIRAM[5]

1. BTECH, NADIMPALLI SATYANARAYANA RAJU INSTITUTE OF TECHNOLOGY, SONTYAM, VISAKHAPATNAM, ANDHRA PRADESH, INDIA - 531173
2. BTECH, NADIMPALLI SATYANARAYANA RAJU INSTITUTE OF TECHNOLOGY, SONTYAM, VISAKHAPATNAM, ANDHRA PRADESH, INDIA - 531173
3. BTECH, NADIMPALLI SATYANARAYANA RAJU INSTITUTE OF TECHNOLOGY, SONTYAM, VISAKHAPATNAM, ANDHRA PRADESH, INDIA - 531173
4. BTECH, NADIMPALLI SATYANARAYANA RAJU INSTITUTE OF TECHNOLOGY, SONTYAM, VISAKHAPATNAM, ANDHRA PRADESH, INDIA – 531173
5.Professor COMPUTER SCIENCE AND ENGINEERING, NADIMPALLI SATYANARAYANA RAJUINSTITUTE OF TECHNOLOGY, SONTYAM, VISAKHAPATNAM, ANDHRA PRADESH, INDIA-531173

ABSTRACT

Sign Language Recognition (SLR) seeks to convert sign language into text or speech in order to enhance communication between deaf-mute people and hearing people.. While having a significant social influence, this task is nonetheless highly difficult due to its intricacy and wide range of hand gestures. Current SLR techniques create classification models based on hand-crafted characteristics to describe sign language motion. Nonetheless, it is challenging to build trustworthy features that can adjust to the wide range of hand motions. To solve this issue, we suggest a novel convolutional neural network (CNN) that automatically and automatically extracts discriminative spatial-temporal features from the raw video stream, reducing the need to build features. To improve performance, multiple video streams with colour and depth information are used. Input to the CNN that integrates colour, depth, and trajectory data includes clue and body joint locations. We test the proposed model on a real dataset obtained using Microsoft Kinect, and we show that it outperforms conventional methods based on manually created features.

Keywords: CNN,SLR

1 INTRODUCTION

One of the most popular ways for people who are hard of hearing to communicate is through sign language, which is represented through different hand and body gestures as well as facial expressions. Sign language recognition is still a very difficult task since it is challenging to cooperatively exploit the information from hand forms and body movement trajectory. In order to assist hearing-impaired people who use sign language to communicate with hearing people, this research suggests an efficient recognition model for translating sign language into text or speech. Strictly speaking, creating descriptors to express hand forms and motion trajectory represents the biggest obstacle to sign language detection. Tracking hand areas in video streams, separating hand-shape images from complicated backgrounds in each frame, and motion detection issues are specifically involved in hand-shape description. trajectory of motion is also connected to matching curves and tracking the critical locations. Despite the fact that there has been a lot of research on these two topics up to this point, it is still challenging to achieve satisfactory results for SLR because of the variance and occlusion of hands and body joints. Moreover, integrating the hand-shape features and trajectory features is a difficult task. To overcome these challenges, we create CNNs that innately incorporate hand forms, action trajectory, and facial expression. We take colour images, depth images, and body skeleton images simultaneously from Microsoft Kinect in place of using standard colour images as input to networks like [1, 2]. A motion sensor with depth and colour information is called Kinect. stream. The body joint locations can be acquired in real-time using the open-source Windows SDK. As a result, we decided to record the dataset of sign words using the Kinect. To distinguish between various sign behaviours, the depth and colour variations at the pixel level provide useful information. Moreover, changes in body joints throughout time can show the progression of sign activities. Using a variety of visual sources as input encourages CNNs to pay attention to changes in depth and trajectory in addition to colour. It is important to note that because CNNs have the ability to learn features automatically from raw data without any human intervention, we can avoid the challenges of tracking hands, segmenting hands from background, and defining descriptors for hands. prior information [3].

Recently, CNNs have been used to classify video streams. CNNs' possible worry is time consumption. The expense of training a CNN with a million-scale in a million videos is several weeks or months. Fortunately, using CUDA for parallel processing still makes it possible to operate efficiently in real-time. We suggest using CNNs to extract spatial and temporal information from a video stream for the purpose of recognising sign language (SLR). Current methods for SLR construct classification models based on hand-crafted characteristics that describe sign language motion. CNNs, on the other hand, can automatically detect motion from unprocessed video data, removing the need to develop features. We create a CNNs using a variety of data formats as input. This design incorporates depth, colour, and trajectory. by applying convolution and subsampling on adjacent video frames to extract information. On several sign words that we have recorded, experimental results show that 3D CNNs can significantly outperform Gaussian mixture models with Hidden Markov models (GMM-HMM) baselines.

2. LITERATURE SURVEY AND RELATED WORK

Python is currently the most well-liked high-level, versatile programming language.
Both procedural and object-oriented programming paradigms are supported by Python. Programmes written in Python are often smaller than those written in other programming languages like Java.
Programmers type much less than other types of writers, and the indentation rule of the language ensures that their work is always readable.
The Python programming language is used by almost all tech-giant companies, such as Google, Amazon, Facebook, Instagram, Dropbox, Uber, and others.
Python's largest asset is its extensive standard library, which may be used for the following things:
• Computer learning
• Apps for GUI (like Kivy, Tkinter, PyQt etc.)
• Image processing (like OpenCV, Pillow); • Web scraping (like Scrapy, BeautifulSoup, Selenium); • Test frameworks, such as those used by YouTube, Instagram, and Dropbox;
• Multimedia

Python's benefits are as follows: Let's look at how Python outperforms different tongues.
1. Generous Libraries
Regular expressions, documentation generation, unit testing, web browsers, threading, databases, CGI, email, image processing, and other features are all included in the large library that Python downloads. So, we don't need to manually write the entire code for it.
2. Flexible
Python can be extended to other languages, as we have already seen. Some of your code can be written in C++ or C, for example. This is useful, particularly for projects.
3. Embedded
Python also has embeddability, which is complementary to extensibility. Python code can be inserted into the source code of another language, such as C++. This enables us to add scripting functionality to our other language code.
Enhanced Productivity
The plainness of the language and Programmers are more productive when using comprehensive libraries than when using languages like Java and C++. Also, you should write less and accomplish more things.
5. IOT Possibilities
Python believes that the Internet of Things has a bright future since it serves as the foundation for cutting-edge platforms like Raspberry Pi. This helps to link the language to the outside world.
6. Easy and Simple
You might need to construct a class in Java to print "Hello World." But all you need with Python is a print statement. Moreover, it is quite simple to learn, comprehend, and code. Because of this, when people learn Python, they find it difficult to switch to other, verboser languages like Java.
7. Readable Considering Reading Python is very similar to reading English because it is not a very verbose language. This explains why learning, comprehending, and coding are so simple. Block definitions can be made without curly braces, and indentation is required. This improves the code's readability much further.
8. Object-orientation
Both the procedural and object-oriented programming paradigms are supported by this language. Classes and objects allow us to simulate the real world, whereas functions assist us with code reuse. A class enables the consolidation of functions and data into one unit.
9. Open-Source Software
Python is available for free, as we have mentioned. Python can be downloaded for free, but you can also have access to its source code, modify it, and even share it. it. To assist you with your tasks, a large number of libraries are downloaded with it.
10. Convenient
If you wish to run your project on another platform after you develop it in a language like C++, you might need to make some changes to it. With Python, though, it's not the same. Here, you simply need to write the code once, and it will work everywhere. It's known as Write Once, Run Anywhere (WORA). You must take care not to include any system-dependent features, though.
12. Interpreted
We shall conclude by stating that it is an interpreted language. Compared to compiled languages, debugging is simpler since statements are performed one at a time.
Do you still have any reservations about Python's benefits? In the remark, mention section. Python's Benefits Over Other Languages
1. Reduced Coding
When compared to other languages, almost all jobs done in Python require less coding. To complete your task, you don't need to look for any third-party libraries because Python offers a fantastic standard library support. This is the rationale behind the widespread recommendation that novices learn Python.
2. Economic
Because Python is open source, anyone can use the free resources to create apps, whether they are individuals, small businesses, or large corporations. Python offers you superior community support because it is well-known and often used.

Python has surpassed Java in the category of the most popular programming languages, according to the 2019 Github annual survey.

3. Everyone Can Use Python Python code can operate on any device, including Linux, Mac, and Windows. For each position, a programmer needs to master a new language, but Python allows you to construct web apps, automate tasks, perform machine learning and data analysis, create games, and create amazing visualisations. It is a programming language with several uses.

The drawbacks of Python

We've already seen several reasons why Python is a wise choice for your project. But if you decide to go that route, you should also be mindful of the outcomes. Now let's look at Python's disadvantages compared to other languages.

(1) Speed restrictions

Python code is run line by line, as we have seen. However because Python is an interpreted language, its performance is frequently sluggish. Yet this isn't a problem. Unless the project's main goal is speed. In other words, Python's advantages are sufficient to make up for its speed constraints unless high speed is absolutely necessary.

2. Lackluster Browsers and Mobile Computing

Python makes a great server-side language, but it's far less common on the client-side. In addition, it is seldom ever utilised to implement applications for smartphones. The Carbonnelle app is one such example.

Despite Brython's presence, it is less well-known since it lacks sufficient security.

3. Limitations on Design

As you are aware, Python uses dynamic typing. As a result, you are not need to define the type of a variable as you write the code. It types with a duck. However, what's that? Well, it just translates to "it must be a duck if it looks like a duck." While this makes coding easier for the programmers, run-time mistakes may result.

4. Insufficient Database Access Layers

Python's database access layers are a little immature in comparison to more popular technologies like JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity). As a result, it is used less frequently in large businesses.

5. Basic

Seriously, we aren't joking. The simplicity of Python can in fact be a drawback. Consider what I did. Python interests me more than Java does. I find that its syntax is so straightforward that the Java code's verbosity looks superfluous.

All of this was about the benefits and drawbacks of the Python programming language.

3 PROPOSED WORK AND ALGORITHM

The filtering of the photographs in this article is significant. Even in dimly lit environments, the accuracy of symbol identification is improved. Prior to saturation and grey scaling, the image is submitted to a filtering system, which looks for the sign depicted in the hands. After the symbol is found, the image is then processed further to produce the word. the process of taking visual pictures, like those of a real-world situation. Preprocessing: Preprocessing is done to prepare the incoming image data for further processing by reducing unneeded noise or enhancing important image attributes. Feature Learning: To be descriptive and help learning, feature learning develops its derived features from the initial assessed data. following education. An worldwide publication that focuses on current and innovative trends in computing and communication is called Recent and Innovative Trends in Computing and Communication. steps[2], and in some instances, this has led to more distinct personal interpretations. Categorization, the process of separating concepts and objects from one another, is a process that is connected to classification. Finding patterns and regularities in findings is what recognition is concerned with. Convolution is an advanced process that extracts several properties from the data. In the initial stage, it extracts basic elements like edges and corners. After that, higher-level layers extract functionality. For the 3D convolution process in CNNs. The input is convolved and has dimensions N x N x D. using the H kernels, each of which measures k x k x D. One output feature is created when one input and one kernel are convolutioned, and H features are created when H kernels are individually convolutioned. Starting in the top-left corner of the input, each kernel is moved from left to right. A kernel is relocated one element downward and then moved one element at a time from left to right if it enters the top-right corner. Up till the kernel reaches the bottom-right corner of the screen, the process is repeated[6]. Convolution is a mathematical operation that requires two inputs, such as an image matrix and a filter or kernel. A digital representation of picture pixels is called an image matrix. Moreover, the filter is a different matrix that processes the picture matrix. We can process any aspect of the image because the kernel is significantly smaller than the image. The picture matrix in this layer should be filtered. Convolution is carried out by combining this filter matrix with the image matrix shown in Figure. Depending on the features to be eliminated, any number of convolution layers can be added. Four arguments are required for the convolution function: the first is the number of filters, the second is the structure of each filter, the third is the shape of the input, and the fourth is the resolution and form of the image. The fourth argument is the triggering function that has to be used. Which neuron can activate depends on the mechanism next fire.

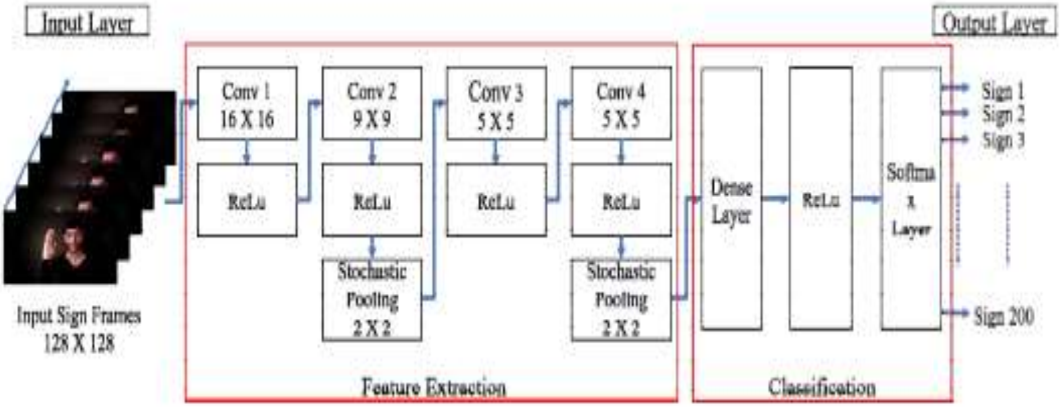


Fig: ARCHITECTURE

4 METHODOLOGIES

Tensorflow
For dataflow and differentiable programming across a variety of tasks, TensorFlow is a free and open-source software library. It is a library for symbolic maths, and neural networks and other machine learning techniques use it. At Google, it is utilised for both research and production.

For usage within Google, the Google Brain team created TensorFlow. On November 9, 2015, it was made available under an open-source Apache 2.0 licence.
Numpy

A versatile package for handling arrays is called Numpy. In addition to tools for working with these arrays, it offers a high-performance multidimensional array object.
It serves as Python's foundational package for scientific computing. It has a number of characteristics, significant ones among them being:
Strong N-dimensional array object
Advanced (broadcasting) features
C/C++ and Fortran integration tools Helpful Fourier transform, random number, and linear algebra capabilities
In addition to its apparent scientific applications, Numpy is a powerful multi-dimensional data container. Numpy's ability to establish any data-types makes it possible for Numpy to quickly and easily interact with a wide range of databases.
Pandas
With its potent data structures, Pandas, an open-source Python library, offers high-performance data manipulation and analysis tools. Python was mostly utilised for data preprocessing and munging. It did not make much of an impact on data analysis. Pandas figured out the solution. Regardless of the source of the data input, we may complete the five standard processes of data processing and analysis using Pandas: prepare, modify, model, and analyse. Python combined with Pandas is utilised in several many academic and professional disciplines, such as finance, economics, statistics, analytics, etc.
Matplotlib
A Python 2D plotting toolkit called Matplotlib creates publication-quality graphics in a range of physical formats and in cross-platform interactive settings. Four graphical user interface toolkits, the Python and IPython shells, the Jupyter Notebook, web application servers, and Python scripts can all make use of Matplotlib. Matplotlib aims to make difficult things feasible and simple things easy. With just a few lines of code, you can create plots, histograms, power spectra, bar charts, error charts, scatter plots, and more. See the sample plots and thumbnail galleries for examples.
Particularly when used in conjunction with IPython, the pyplot package offers a MATLAB-like interface for basic plotting. You have all control if you're a power user. line styles, font attributes, axis attributes, etc. via an object-oriented interface or with a set of MATLAB-friendly functions.
The Scikit-Learn
With a standardised Python interface, Scikit-learn offers a variety of supervised and unsupervised learning techniques. It is released under several Linux distributions and is available under a liberal simplified BSD licence, which promotes both academic and commercial use. Python
Python is a high-level, interpreted general-purpose programming language. Python, which was developed by Guido van Rossum and originally made available in 1991, stresses code readability and makes extensive use of whitespace.

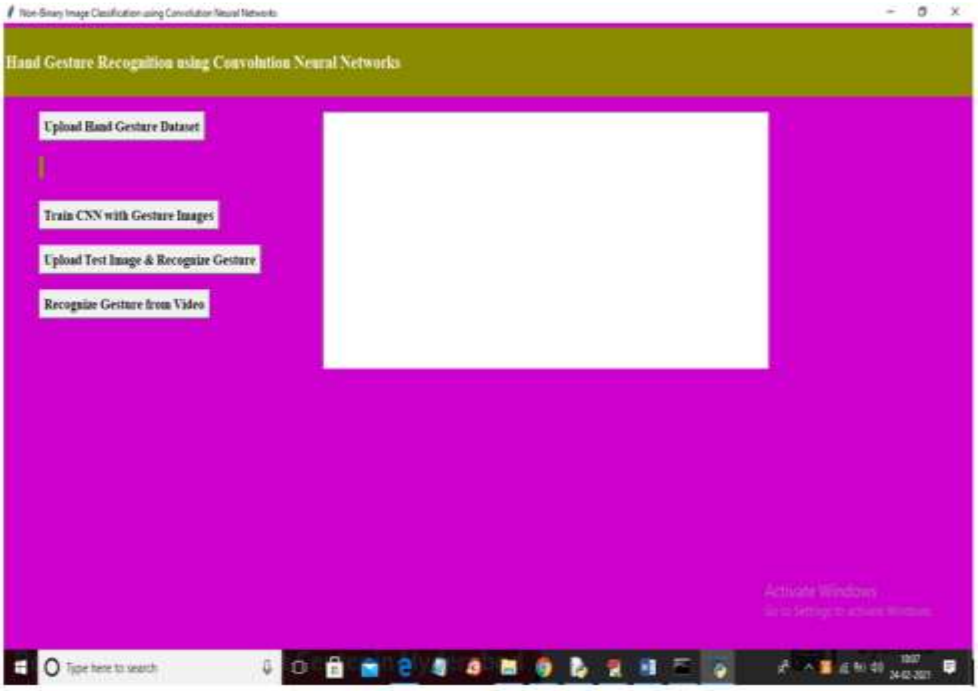
Python has an autonomous memory management system and a dynamic type system. It includes a sizable and thorough standard library, supports a variety of programming paradigms, including imperative, functional, procedural, and object-oriented. • Python is Interpreted The interpreter processes Python at runtime. Your software does not need to be compiled before running. This is comparable to PHP and PERL.

• Python is Interactive; when writing programmes, you can actually sit at a Python prompt and communicate with the interpreter immediately.

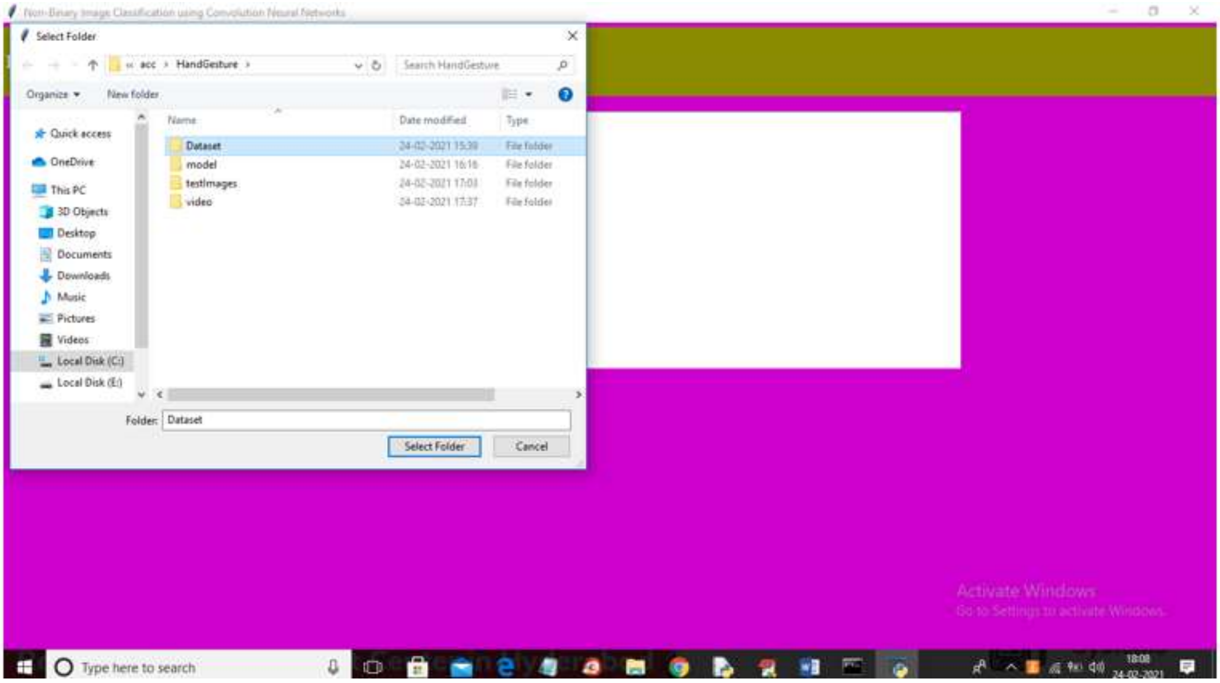
Python also acknowledges the significance of development pace. This includes having access to strong constructs that prevent laborious code repetition as well as readable and concise code. Maintainability is related to this and may be an almost meaningless measure, but it does provide information about how much code you must scan, read, and/or comprehend in order to fix issues or modify behaviour. Due to the rapid advancement and simplicity of programming, The vast standard library is essential to another area where Python shines, and other languages can learn the fundamentals of Python. All of its tools were simple to use, saved a tonne of time, and several of them could later be fixed and upgraded by non-Python experts without hurting anything.

5.RESULTS AND DISCUSSION SCREENSHOTS

To run project double click on run.bat file to get below screen


Non-Binary Image Classification using Convolution Neural Networks
Hand Gesture Recognition using Convolution Neural Networks
Upload Hand Gesture Dataset
Train CNN with Gesture Images
Upload Test Image & Recognize Gesture
Recognize Gesture from Video
Type here to search

In above screen click on 'Upload Hand Gesture Dataset' button to upload dataset and to get below screen



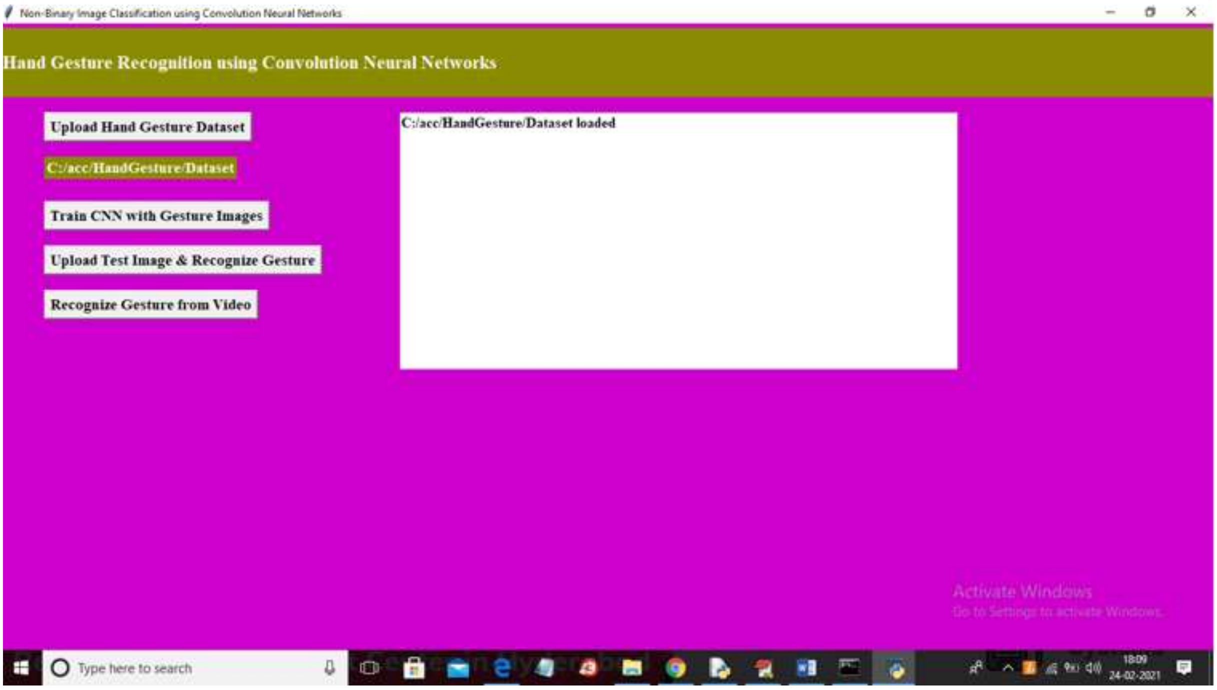
Non-Binary Image Classification using Convolution Neural Networks

Select Folder

Name	Date modified	Type
Dataset	24-02-2021 15:38	File folder
model	24-02-2021 16:16	File folder
testImages	24-02-2021 17:03	File folder
video	24-02-2021 17:37	File folder

Folder: Dataset

In above screen selecting and uploading 'Dataset' folder and then click on 'Select Folder' button to load dataset and to get below screen



Non-Binary Image Classification using Convolution Neural Networks

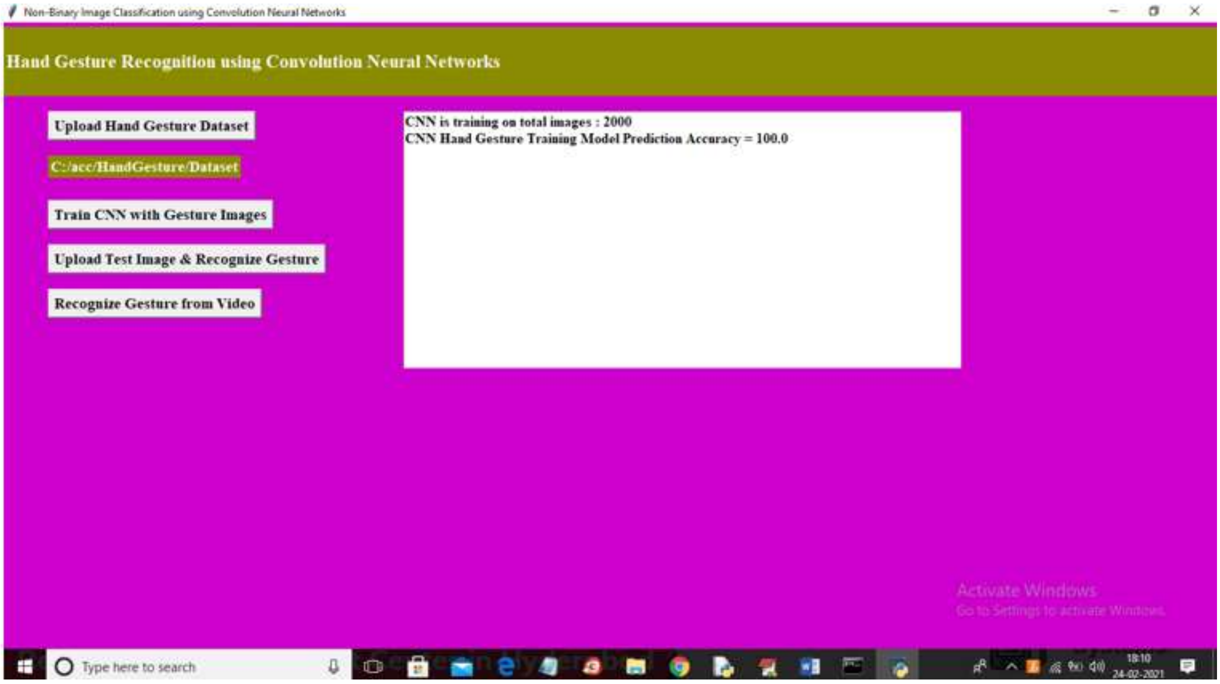
Hand Gesture Recognition using Convolution Neural Networks

C:/acc/HandGesture/Dataset

C:/acc/HandGesture/Dataset loaded

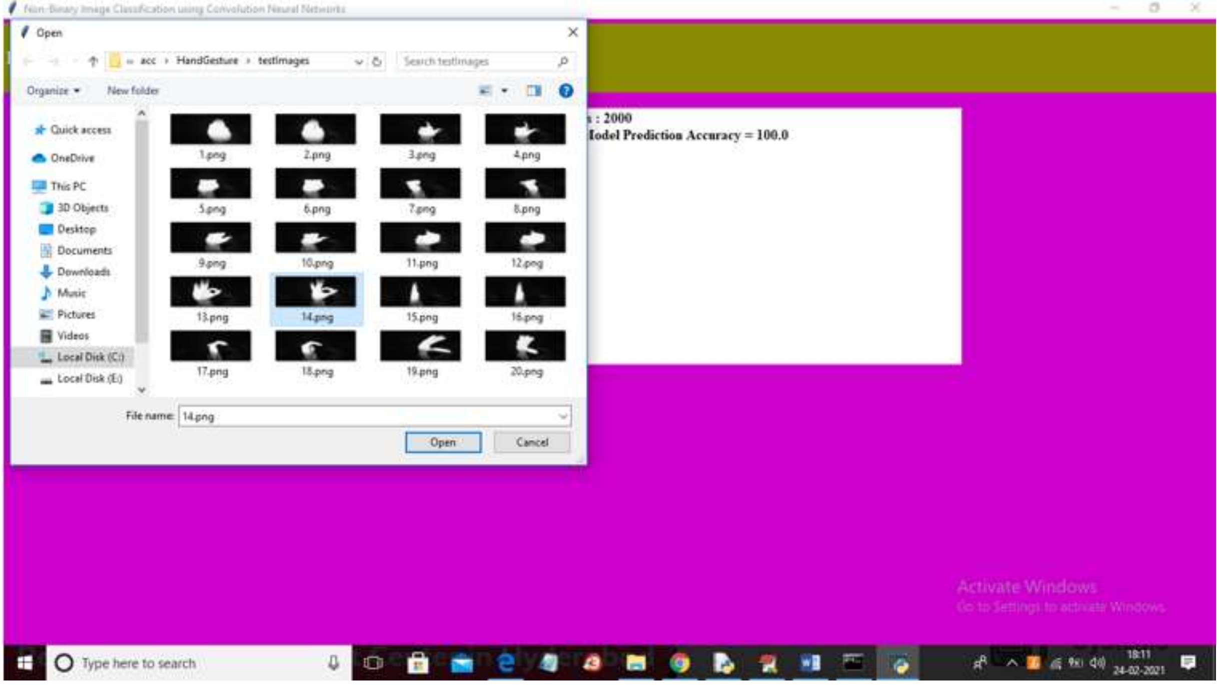
Upload Hand Gesture Dataset
Train CNN with Gesture Images
Upload Test Image & Recognize Gesture
Recognize Gesture from Video
Activate Windows
Go to Settings to activate Windows.
Type here to search
18:09
24-02-2021

In above screen dataset loaded and now click on 'Train CNN with Gesture Images' button to trained CNN model and to get below screen


Non-Binary Image Classification using Convolution Neural Networks
Hand Gesture Recognition using Convolution Neural Networks
Upload Hand Gesture Dataset
C:/acc/HandGesture/Dataset
Train CNN with Gesture Images
Upload Test Image & Recognize Gesture
Recognize Gesture from Video
Type here to search

```
CNN is training on total images : 2000
CNN Hand Gesture Training Model Prediction Accuracy = 100.0
```

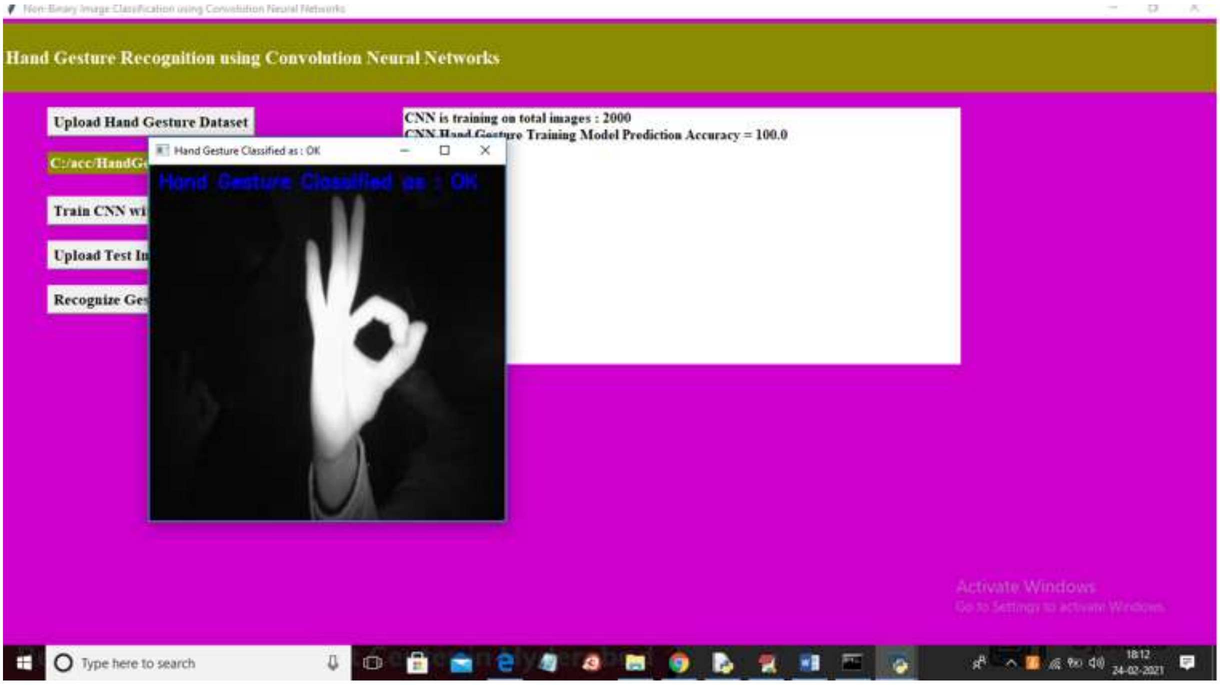
In above screen CNN model trained on 2000 images and its prediction accuracy we got as 100% and now model is ready and now click on ‘Upload Test Image & Recognize Gesture’ button to upload image and to gesture recognition



File name: 14.png

```
s : 2000
lodel Prediction Accuracy = 100.0
```

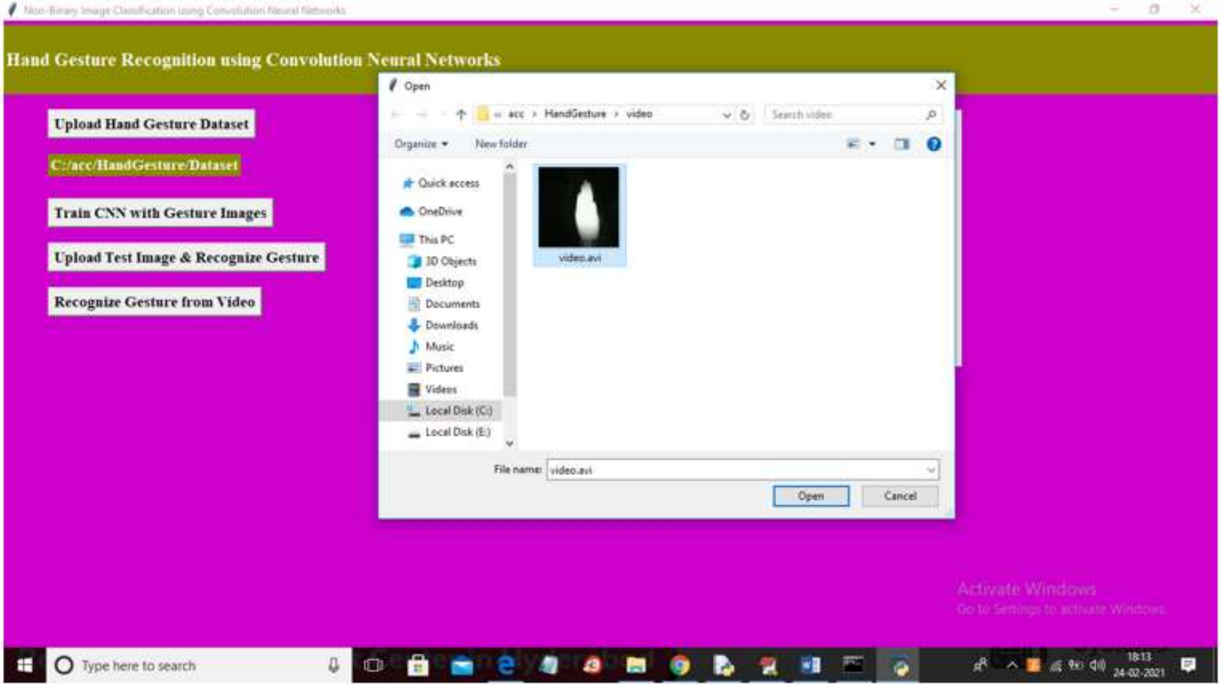
In above screen selecting and uploading ’14.png’ file and then click Open button to get below result


Non-Binary Image Classification using Convolution Neural Networks
Hand Gesture Recognition using Convolution Neural Networks
Upload Hand Gesture Dataset
C:/acc/HandG
Train CNN wi
Upload Test In
Recognize Ges
Hand Gesture Classified as : OK
Type here to search
Activate Windows
Go to Settings to activate Windows.
18:12
24-02-2021

CNN is training on total images : 2000
CNN Hand Gesture Training Model Prediction Accuracy = 100.0

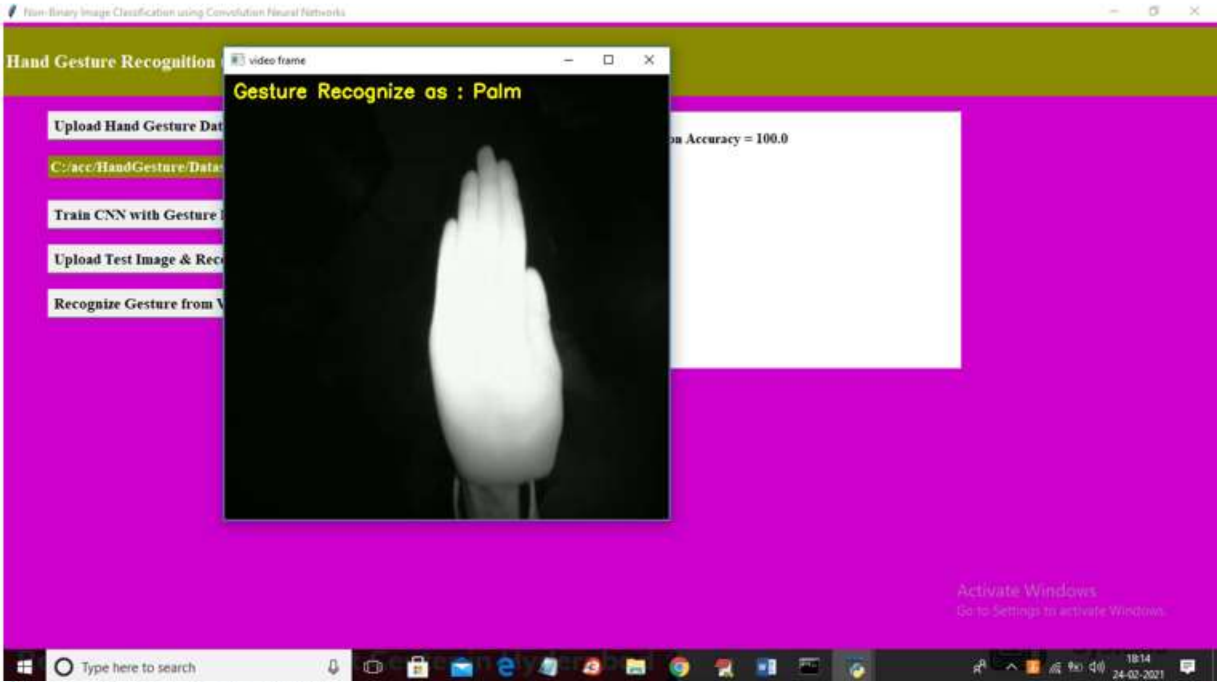
Hand Gesture Classified as : OK

In above screen gesture recognize as OK and similarly you can upload any image and get result and now click on ‘Recognize Gesture from Video’ button to upload video and get result


Non-Binary Image Classification using Convolution Neural Networks
Hand Gesture Recognition using Convolution Neural Networks
Upload Hand Gesture Dataset
C:/acc/HandGesture/Dataset
Train CNN with Gesture Images
Upload Test Image & Recognize Gesture
Recognize Gesture from Video
Open
« acc › HandGesture › video
Search video
Organize ▾
New folder
Quick access
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
video.avi
Open
Cancel
Type here to search
Activate Windows
Go to Settings to activate Windows.
18:13
24-02-2021

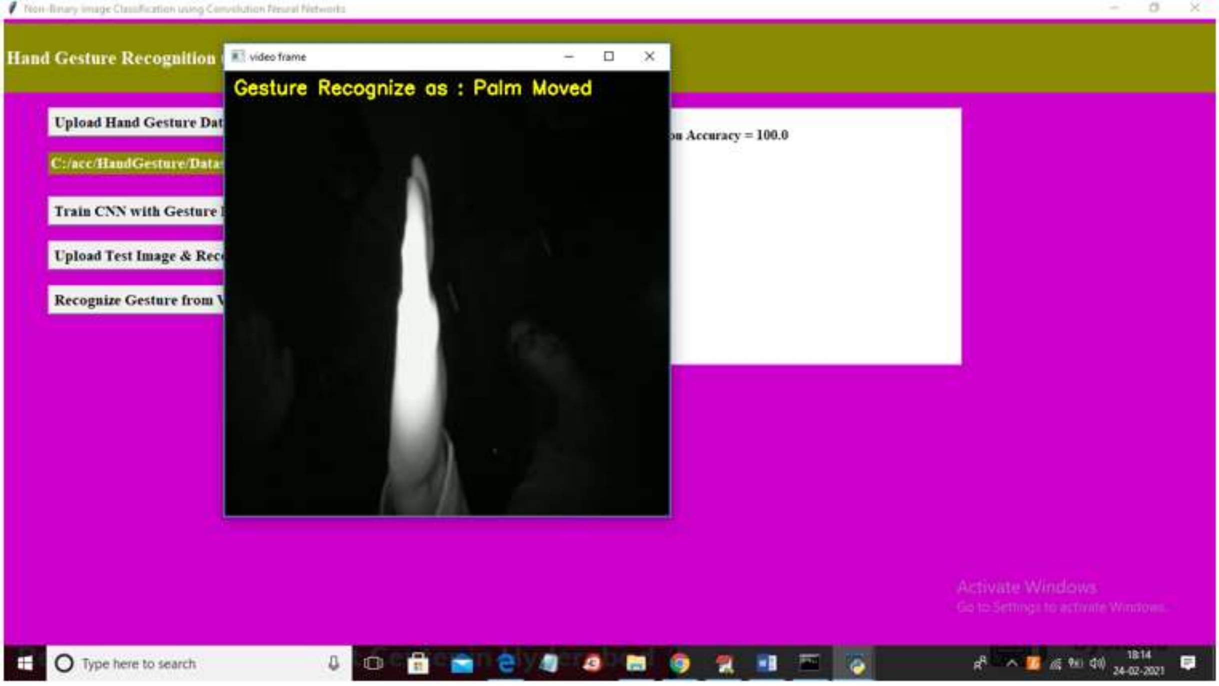
File name: video.avi

In above screen selecting and uploading ‘video.avi’ file and then click on ‘Open’ button to get below result



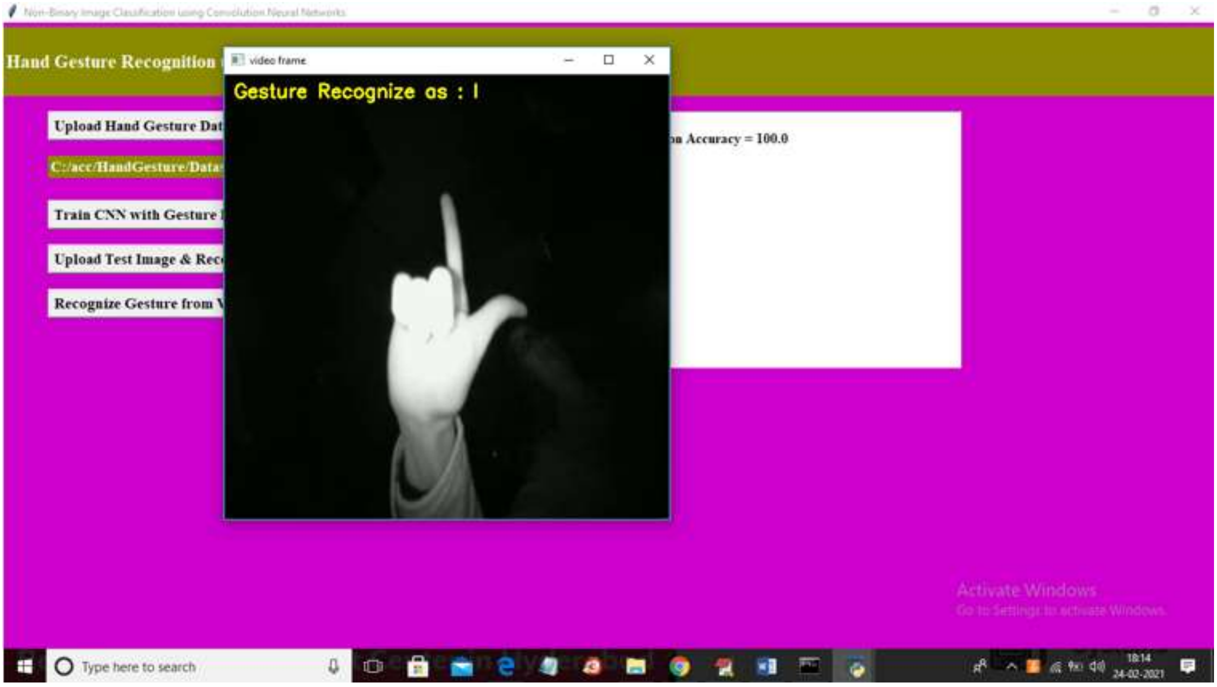
Gesture Recognize as : Palm

...n Accuracy = 100.0



Gesture Recognize as : Palm Moved

...n Accuracy = 100.0



C:/acc/HandGesture/Data

Gesture Recognize as : I

n Accuracy = 100.0

In above screen as video play then will get recognition result

6.CONCLUSION

For the purpose of recognising sign language, we created a CNN model. By using 3D convolutions, our model learns and extracts both spatial and temporal characteristics. The created deep architecture separates convolution from subsampling after extracting various sorts of information from nearby input frames. Information from each channel is combined to create the final feature representation. These feature representations are classified using a multilayer perceptron classifier. For the same dataset, we compare and contrast CNN and GMM-HMM. The outcomes of the experiments show how effective the suggested approach is.

7.REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 2012, pp. 1097–1105.

[2] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, “Large-scale video classification with convolutional neural networks,” in CVPR, 2014. [3] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick ´ Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[4] Hueihan Jhuang, Thomas Serre, Lior Wolf, and Tomaso Poggio, “A biologically inspired system for action recognition,” in Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on. Ieee, 2007, pp. 1–8. [5] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, “3D convolutional neural networks for human action recognition,” IEEE TPAMI, vol. 35, no. 1, pp. 221–231, 2013.

[6] Kirsti Grobel and Marcell Assan, “Isolated sign language recognition using hidden markov models,” in Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on. IEEE, 1997, vol. 1, pp. 162–167.

[7] Thad Starner, Joshua Weaver, and Alex Pentland, “Realtime american sign language recognition using desk and wearable computer based video,” IEEE TPAMI, vol. 20, no. 12, pp. 1371–1375, 1998.

[8] Christian Vogler and Dimitris Metaxas, “Parallel hidden markov models for american sign language recognition,” in

Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on. IEEE, 1999, vol. 1, pp. 116–122.

[9] Kouichi Murakami and Hitomi Taguchi, "Gesture recognition using recurrent neural networks," in Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, 1991, pp. 237–242.

[10] Chung-Lin Huang and Wen-Yi Huang, "Sign language recognition using model-based tracking and a 3D hopfield neural network," Machine vision and applications, vol. 10, no. 5-6, pp. 292–307, 1998.

[11] Jong-Sung Kim, Won Jang, and Zeungnam Bien, "A dynamic gesture recognition system for the korean sign language (ksl)," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 26, no. 2, pp. 354–359, 1996.

[12] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," arXiv preprint arXiv:1311.2524, 2013.

[13] Ronan Collobert and Jason Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in ICML. ACM, 2008, pp. 160–167.

[14] Clement Farabet, Camille Couprie, Laurent Najman, ´ and Yann LeCun, "Learning hierarchical features for scene labeling," IEEE TPAMI, vol. 35, no. 8, pp. 1915– 1929, 2013. [15] Srinivas C Turaga, Joseph F Murray, Viren Jain, Fabian Roth, Moritz Helmstaedter, Kevin Briggman, Winfried Denk, and H Sebastian Seung, "Convolutional networks can learn to generate affinity graphs for image segmentation," Neural Computation, vol. 22, no. 2, pp. 511– 538, 2010.

[16] Ao Tang, Ke Lu, Yufei Wang, Jie Huang, and Houqiang Li, "A real-time hand posture recognition system using deep neural networks," ACM Transactions on Intelligent Systems and Technology, 2014.

[17] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt, "Sequential deep learning for human action recognition," in Human Behavior Understanding, pp. 29–39. Springer, 2011.