ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET) Vol. 9, Issue 8, August 2022

Conditional Sentence Generation and Cross-Modal Reranking for Sign Language Translation

Mr. E.DilipKumar
Department of Mca
Dhanalakshmi Srinivasan college of Engineering and Technology

Ms.Deepika.N
Department of Mca
Dhanalakshmi Srinivasan college of Engineering and Technology

Abstract: Sign Language are a form of nonverbal communication in which visible bodily actions are used to communicate important messages, either in place of speech or together and in parallel with spoken words. Sign Language include movement of the hands, face, or other parts of the body. This project is to train a Deep Learning algorithm capable of classifying images of different sign language, such as a alphabet letter, and numeric A comparison of the proposed and current algorithms reveals that the accuracy hand gesture types classification based on CNNs is higher than other algorithms. It is predicted that the success of the obtained results will increase if the CNN method is supported by adding extra feature extraction methods and classify successfully fruits types on image. The goal is to develop a deep learning model for Sign Language classification by convolutional neural network algorithm for potentially classifying the results in the form of best accuracy by comparing the CNN architectures.

I. INTRODUCTION

They proposed for a novel framework based on word existence verification, sentence generation and cross modal re-ranking for SLT. The framework first checks the existence of words in the vocabulary by a series of binary classification to learn a cross modal similarity measurement model to rerank the candidate sentences by learning their similarity with sign videos which contain gesture transition information, After that, the appearing words are assembled and guided by a pre-trained spoken language generator to produce multiple candidate sentences in spoken language manner. Last but not least, we select the sentence most semantically similar to the input sign video as the translation result with a cross modal re-ranking model.

4.2 Drawback:

- They are not using proper technique for accurate sign language.
- They are not using CNN model.

PROPOSED SYSTEM:

The proposes the sign language presented are reasonably distinct, the images are clear and without background. Also, there is a reasonable quantity of images, which makes our model more robust. The drawback is that for different problems, we would probably need more data to stir the parameters of our model into a better direction.We proposed a deep learning (dl) based sign language classification method to prevent gestures. the deep learning method used in the study is the LeNet convolutional neural network (cnn). it is predicted that the success of the obtained results will increase if the cnn method is supported by adding extra feature extraction methods and classify successfully sign language.

Advantages:

- To identify the sign language used on artificial neural network.
- It is best model for deep learning technique to easily identify the sign language types of alphabets and numeric.

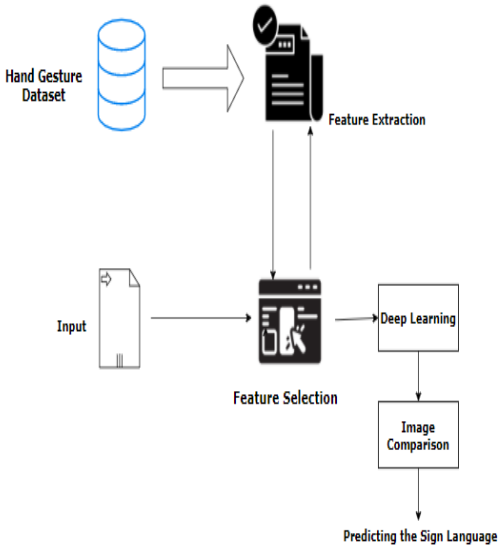
SYSTEM DESIGN:

System design is described as a process of planning a new business system or more to replace or to complement an existing system.

SYSTEM ARCHITECTURE

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022



CNN Model steps:

Conv2d:

The 2D convolution is a fairly simple operation at heart: you start with a kernel, which is simply a small matrix of weights. This kernel "slides" over the 2D input data, performing an elementwise multiplication with the part of the input it is currently on, and then summing up the results into a single output pixel.

This is all in pretty stark contrast to a fully connected layer. If this were a standard fully connected layer, you'd have a weight matrix of 25×9 = 225 parameters, with every output feature being the weighted sum of every single input feature. Convolutions allow us to do this transformation with only 9 parameters, with each output feature, instead of "looking at" every input feature, only getting to "look" at input features coming from roughly the same location. Do take note of this, as it'll be critical to our later discussion.

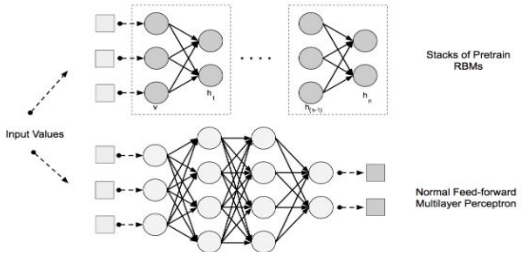
MaxPooling2D layer

Down samples the input along its spatial dimensions (height and width) by taking the maximum value over an input window for each channel of the input. The resulting output, when using the "valid" padding option, has a spatial shape (number of rows or columns) of: output_shape = math.floor((input_shape - pool_size) / strides) + 1 (when input_shape >= pool_size)The resulting output shape when using the "same" padding option is: output_shape = math.floor((input_shape - 1) / strides) + 1

ARTIFICIAL NEURAL NETWORK:

Artificial Neural Networks (ANN) are multi-layer fully-connected neural nets that look like the figure below. They consist of an input layer, multiple hidden layers, and an output layer. Every node in one layer is connected to every other node in the next layer. We make the network deeper by increasing the number of hidden layers.

Key Points related to the architecture:



The network architecture has an input layer, hidden layer (there can be more than 1) and the output layer. It is0 also called MLP (Multi Layer Perceptron) because of the multiple layers. The hidden layer can be seen as a "distillation layer" that distills some of the important patterns from the inputs and passes it onto the next layer to see. It makes the network faster and efficient.

. The activation function serves two notable purposes:

- It captures non-linear relationship between the inputs
- It helps convert the input into a more useful output.

Key advantages of neural Networks:

ANNs have some key advantages that make them most suitable for certain problems and situations:

1. ANNs have the abil1ity to learn and model non-linear and complex relationships, which is really important because in real-life, many of the relationships between inputs and outputs are non-linear as well as complex.ANNs can generalize — After learning from the initial inputs and their relationships, it can infer unseen relationships on unseen data as well, thus making the model generalize and predict on unseen data.

ARCHITECTURE OF CNN

CONVOLUTIONAL NEURAL NETWORK:

A Convolutional neural network (CNN) is one type of Artificial Neural Network. A Convolutional neural network (CNN) is a neural network that has one or more convolutional layers and are used mainly for image processing, classification, segmentation and also for other auto correlated data.

TYPES OF CNN:

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022

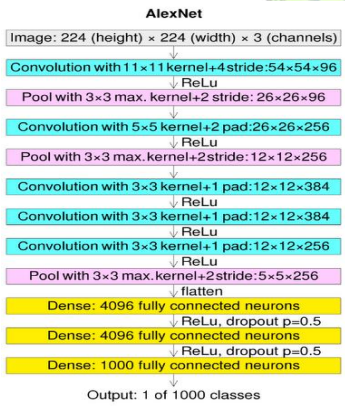
- AlexNet
- LeNet

ALEXNET:

AlexNet is the name of a convolutional neural network which has had a large impact on the field of machine learning, specifically in the application of deep learning to machine vision. AlexNet was the first convolutional network which used GPU to boost performance.

AlexNet architecture consists of 5 convolutional layers, 3 max-pooling layers, 2 normalization layers, 2 fully connected layers, and 1 softmax layer. Each convolutional layer consists of convolutional filters and a nonlinear activation function ReLU. The pooling layers are used to perform max pooling

Architecture of AlexNet:



are the a user are

Convolutional layers:

Convolutional layers the layers where filters are applied to original image, or to other feature maps in deep CNN. This is where most of the specified parameters in the network. The most important parameters are the number of kernels and the size of the kernels.

Pooling layers:

Pooling layers are similar to convolutional layers, but they perform a specific function such as max pooling, which takes the maximum value in a certain filter region, or average pooling, which takes the average value in a filter region. These are typically used to reduce the dimensionality of the network.

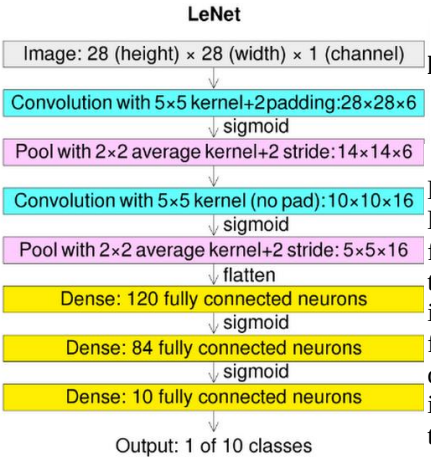
Dense or Fully connected layers: Fully connected layers are placed before the classification output of a CNN and are used to flatten the results before classification. This is similar to the output layer of an MLP.

LENET:

LeNet was one among the earliest convolutional neural networks which promoted the event of deep learning. After innumerous years of analysis and plenty of compelling iterations, the end result was named LeNet.

Architecture of LeNet-5:

LeNet-5 CNN architecture is made up of 7 layers. The layer composition consists of 3 convolutional layers, 2 subsampling layers and 2 fully connected layers.



Convolutional layers:

Convolutional layers are the layers where filters are applied to the original image, or to other feature maps in a deep CNN. This is where most of the user-specified parameters are in the network. The most important parameters are the number of kernels and the size of the kernels.

Pooling layers:

Pooling layers are similar to convolutional layers, but they perform a specific function such as max pooling, which takes the maximum value in a certain filter region, or average pooling, which takes the average value in a filter region. These are typically used to reduce the dimensionality of the network.

Dense or Fully connected layers:

Fully connected layers are placed before the classification output of a CNN and are used to flatten the results before classification. This is similar to the output layer of an MLP.

7.5 LIST OF MODULES

1. Manual Net
2. AlexNet
3. LeNet
4. Deploy

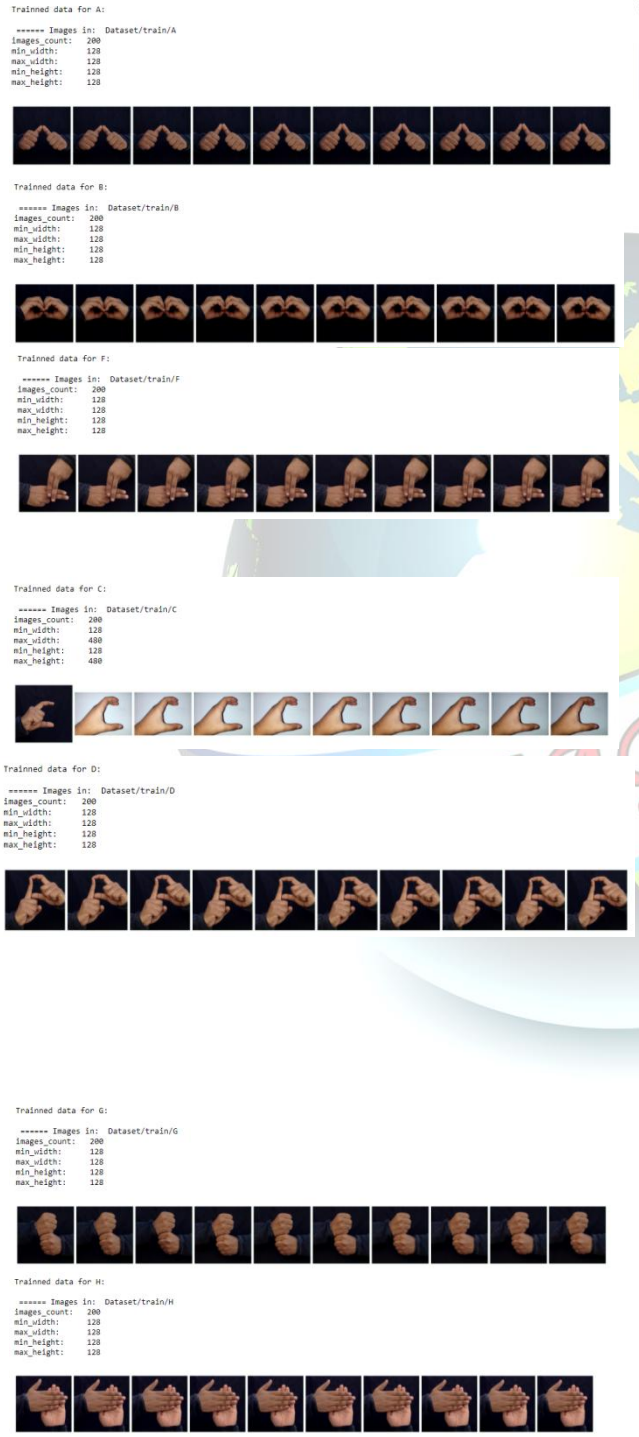
MODULE DESCRIPTION

IMPORT THE GIVEN IMAGE FROM DATASET:

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

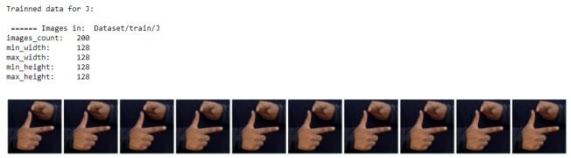
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022

We have to import our data set using keras preprocessing image data generator function also we create size, rescale, range, zoom range, horizontal flip. Here we set train, test, and validation also we set target size, batch size and class-mode from this function we have to train using our own created network by adding layers of CNN.



WORKING PROCESS OF LAYERS IN CNN

MODEL:



A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance to various aspects/objects in the image and be able to differentiate one from the other. The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Their network consists of four layers with 1,024 input units, 256 units in the first hidden layer, eight units in the second hidden layer, and two output units.

Convo Layer:

Convo layer is sometimes called feature extractor layer because features of the image are get extracted within this layer. First of all, a part of image is connected to Convo layer to perform convolution operation as we saw earlier and calculating the dot product between receptive field. Result of the operation is single integer of the output volume. It will repeat the same process again and again until it goes through the whole image. The output will be the input for the next layer.

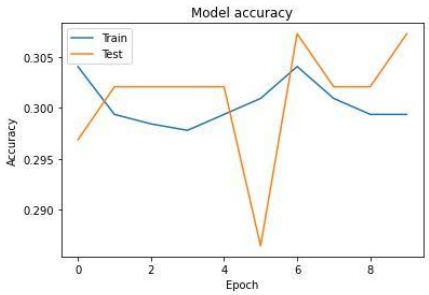


Fig 8.1: CNN model trained dataset accuracy

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022

Pooling Layer:

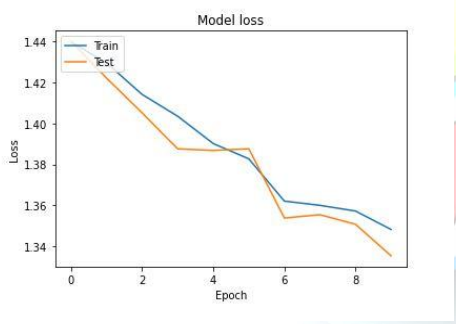
Pooling layer is used to reduce the spatial volume of input image after convolution. It is used between two convolution layers. So, the max pooling is only way to reduce the spatial volume of input image. It has applied max pooling in single depth slice with Stride of 2. It can observe the 4 x 4 dimension input is reducing to 2 x 2 dimensions.

Fully Connected Layer (FC):

Fully connected layer involves weights, biases, and neurons. It connects neurons in one layer to neurons in another layer. It is used to classify images between different categories by training.

Softmax / Logistic Layer:

Softmax or Logistic layer is the last layer of CNN. It resides at the end of FC layer. Logistic is used for binary classification and softmax is for multi-classification.



Output Layer:

Output layer contains the label which is in the form of one-hot encoded. Now you have a good understanding of CNN.

SIGN LANGUAGE CLASSIFICATION IDENTIFICATION:

We give input image using keras preprocessing package. That input Image converted into array value using pillow and image to array

Libraries Required:

TensorFlow:

TensorFlow is a Python library for fast numerical computing created and released by Google. It is a foundation library that can be used to create Deep Learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow.

TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner.

Let us now consider the following important features of TensorFlow −

It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors.

- It includes a programming support of deep neural networks and machine learning techniques.

Matplotlib:

Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. Matplotlib is written in Python and makes use of NumPy, the numerical mathematics extension of Python. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.

The OS module in Python comes with various functions that enables developers to interact with the Operating system that they are currently working on. In this article we'll be learning mainly to create and delete a directory/folder, rename a directory and even basics of file handling.

The OS comes under Python's standard utility modules. This module offers a portable way of using operating system dependent functionality.

DEPLOY

Deploying the model in Django Framework and predicting output

In this module the trained deep learning model is converted into hierarchical data format file (.h5 file) which is then deployed in our django framework for providing better user interface and predicting the output whether the given image is A/B/C/D/E/F/G/H/I/J.

Django

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022

focus on writing your app without needing to reinvent the wheel.Django helps you write software that is:

Versatile
Django can be (and has been) used to build almost any type of website .It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc).

Secure
Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically.

Scalable
Django uses a component-based "shared-nothing" architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed).

Maintainable
Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don't Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code.

SYSTEM TESTING

Testing
System Analysis and Design process including Requirement Analysis, Business Solution Options, Feasibility Study, Architectural Design was discussed in previous chapter. Generally, Software bugs will almost always exist in any software module. But it is not because of the carelessness or irresponsibility of programmer but because of the complexity. Humans have only limited ability to manage complexity. This chapter discusses about the testing of the solution and implementation methodologies.

10.2 Unit Testing
Software Testing is the process of executing a program or system with the intent of finding errors. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions. Testing stages of the project can be explained as below and system was tested for all these stages.

Component or unit testing
- Individual components are tested independently;
- Components may be functions or objects or coherent groupings of these entities.

● **System testing**
- Testing of the system as a whole. Testing of emergent properties is particularly important.

10.3 Acceptance testing
- Testing with customer data to check that the system meets the customer's needs.

Testing Methods and Comparison

10.4 Black Box Testing
Black Box Testing is testing without the knowledge of the internal workings of the item being tested. When black box testing is applied to software engineering, the tester selects valid and invalid input and what the expected outputs should be, but not how the program actually arrives at those outputs. Black box testing methods include equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing. This method of test design is applicable to all levels of software testing: unit, integration, functional testing, system and acceptance.

10.5 White Box Testing
White box testing (glass box testing) strategy deals with the internal data structures and algorithms. The tests written based on the white box testing strategy incorporate coverage of the code written, branches, paths, statements and internal logic of the code etc. These testers require programming skills to identify all paths through the software. Types of white box testing includes code coverage (creating tests to satisfy some criteria of code coverage.) mutation testing methods, fault injection methods, static testing.

PERFORMANCE AND LIMITATION

11.1 Conclusion:
It focused how image from given dataset (trained dataset) in field and past data set used predict the pattern of different hand gestures using CNN model. This brings some of the following different gestures prediction. We had applied different type of CNN compared the accuracy and saw that LeNet makes better classification and the .h5 file is taken from there and that is deployed in Django framework for better user interface.

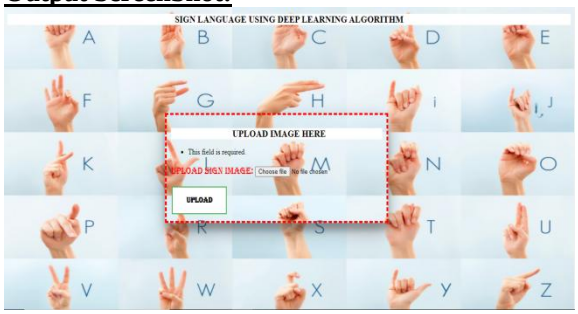
11.2 Future Work:
- Sign Language prediction to connect with AI model.
- To automate this process by show the prediction result in web application or desktop application.

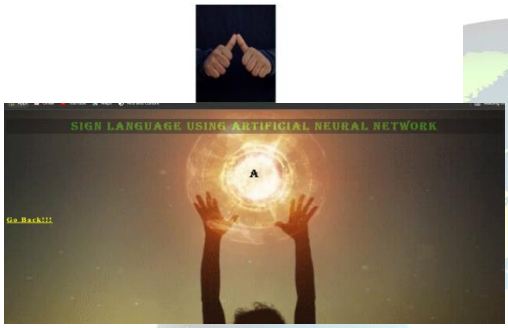
ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 9, Issue 8, August 2022

To optimize the work to implement in Artificial Intelligence environment

Output ScreenShot:





REFERENCES

- R. Cui, H. Liu, and C. Zhang, "A deep neural framework for continuous sign language recognition by iterative training," IEEE Transactions on Multimedia (TMM), vol. 21, no. 7, pp. 1880–1891, 2019.

- J. Pu, W. Zhou, and H. Li, "Iterative alignment network for continuous sign language recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

- Z. Zhang, J. Pu, L. Zhuang, W. Zhou, and H. Li, "Continuous sign language recognition via reinforcement learning," in International Conference on Image Processing (ICIP), 2019.

- H. Wang, X. Chai, and X. Chen, "A novel sign language recognition framework using hierarchical grassmann covariance matrix," IEEE Transactions on Multimedia (TMM), vol. 21, no. 11, pp. 2806–2814, 2019.

- J. Pu, W. Zhou, and H. Li, "Dilated convolutional network with iterative optimization for continuous sign language recognition." in International Joint Conference on Artificial Intelligence (IJCAI), 2018.