MATLAB SPEECH RECOGNITION USING DEEP LEARNING TO INSTRUCTROBOTS IN THE INDUSTRIES

A.VISVA SAI SRIDHAR K.TARUN SAI CHOWDARY B.SREE HARSHA

Guide: Dr.Malay Kumar Hota

ABSTRACT

Robots are perfect replacement for humans in the industry.But there are some areas where humans need to instruct the robots to place objects.One such application is in the diary industry and the automobile industry.This projects aimsat instructing the robots to place the objects according to the wish of the controller present in the control room.We use matlab for simulation and Artificialneural network to train the robots.

In this project we build a deep learning model using convolution neural network after taking it through the machine cycle of testing and training using the very famous data set the google command data set.Then we make it industry ready byimproving its accuracy and deploying into the controller room

Introduction

Speech recognition is one of the next generation technologies forhuman-computer interaction.Speech recognition has been researched since the late 1950s but due to its computational complexity and limited computing capabilities of the last few decades, its progress has been impeded. In laboratory settingsautomatic speech recognition systems (ASR) have achieved high levels of recognition accuracies, which tend to degrade in real world environment

The latest trend in automation comes from the fact that deep learning is the new technology adopted by many industries.This can also be used for speechrecognition.This ensures that computers/robots learn the signals before handand thus identify them when the instructor uses the same language that was pre heard by the robot.

Deep learning consists of a multiple of machine learningalgorithms fed with inputsin the form of multiple layeredmodels. These models are usually neural networks consistingof different levels of non-linear operations. The machinelearning algorithms attempt to learn from these deep neuralnetworks by extracting specificfeatures and information .Prior to 2006, searching deep architecture inputs was not apredictable straight forward task; however, the developmentof deep learningalgorithms helped resolve this issue andsimplified the process of searching the parameter space ofdeep architectures . Deep learning models can also operateas agreedy layerwise unsupervised pre-training. This meansthat it will learn hierarchy from extracted features from eachlayer at a time.

We sometimes feel as if we've got as far as we can go with technological advancement and to be honest what we've achieved in many areas is nothing shortof fantastic. And then every so often a new breakthrough is made and we're left astounded by the new doors that have been thrown open for us. One such technological advancement is voice recognition software.

These voice-activated digital assistants or virtual assistants have applications inmany sectors, such as in finance, marketing, human resources, and even in thepublic transportation. They are bringing down costs, simplifying processes, andincreasing the overall efficiency of a business.

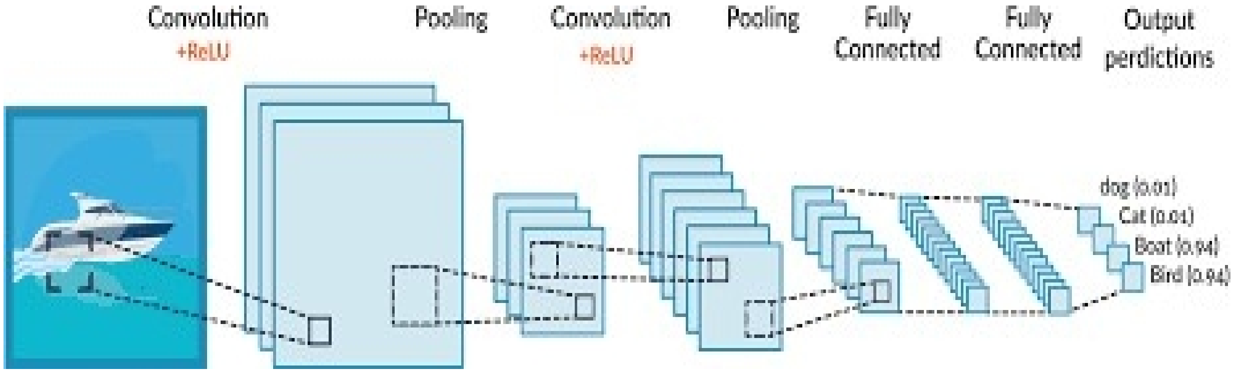
The improvement in natural language understanding and speech accuracy rates, aswell the back of AI, ML, Big Data, and Cloud processing, have led businesses to explore the scopes of bringing in a speech recognition system to their processes and procedures.

Electronic documents replaced paper based record keeping practices a decade agoor two. Now, the technology is going a step further by replacing manually going through files on a computer with voice command. The technology of voice recognition has evolved to a level where you just don't need to type out a correspondence. You can dictate it to a computer having voice assistance which cangenerate a flawless doc, instead. Offices will have more secured environment if swiping cards are replaced with the voice recognition. Voice commands control lighting, temperature and can optimize your comfort based on predetermined indicators. So, the speech recognition technology can be used in various office processes.

Voice recognition and AI can turn into a sophisticated blend to go ahead and devise a smartest solution that doesn't require manual inputting in performing operations.
Google has already integrated its voice assistant services to almost all of its appslike Gmail, Maps, Youtube, Play Music, Play Movie, and Google Play. Amazon's offerings via Alex are quite impressive.

METHODOLOGY USED

TRAINING A CONVOLUTION NEURAL NETWORK USING MATLAB



A **Convolutional Neural Network (ConvNet/CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases)to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand- engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern ofNeurons in the Human Brain and was inspired by the organization of the VisualCortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visualarea.

The commands that were to be identified by the robot are given to the convolutionneural network and we Use a weighted cross entropy classification loss. weightedClassificationLayer(classWeights) creates a custom classification layer that calculates the cross entropy loss with observations weighted by classWeights. Specify the class weights in the same order as the classes appear incategories(YTrain). To give each class equal total weight in the loss, use class weights that are inversely proportional to the number of training examples in each class. When using the Adam optimizer to train the network, the training algorithm is independent of the overall normalization of the class weights.

Convolutional Neural Networks are very similar to ordinary Neural Networks from the previous chapter: they are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. The whole network still expresses asingle differentiable score function: from the raw image pixels on one end to class scores at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully-connected) layer and all the tips/tricks we developed for learning regular Neural Networks still apply.

Regular Neural Nets don't scale well to full images. In CIFAR-10, images are only of size 32x32x3 (32 wide, 32 high, 3 color channels), so a single fully-connected neuron in a first hidden layer of a regular Neural Network would have 32*32*3 =3072 weights. This amount still seems manageable, but clearly this fully- connected structure does not scale to larger images. For example, an image of more respectable size, e.g. 200x200x3, would lead to neurons that have

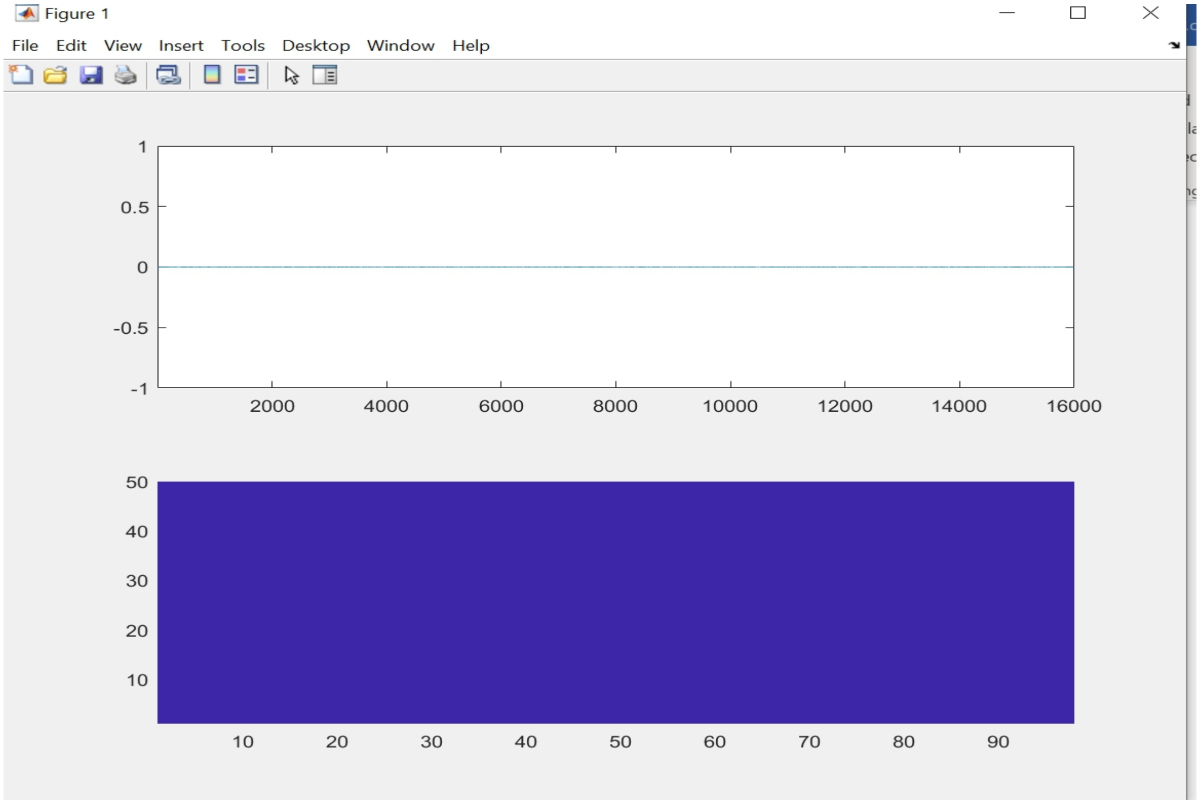
200*200*3 = 120,000 weights. Moreover, we would almost certainly want to have several such neurons, so the parameters would add up quickly! Clearly, thisfull connectivity is wasteful and the huge number of parameters would quickly lead to overfitting.

Dataset used for training:Google voice data set 2017.

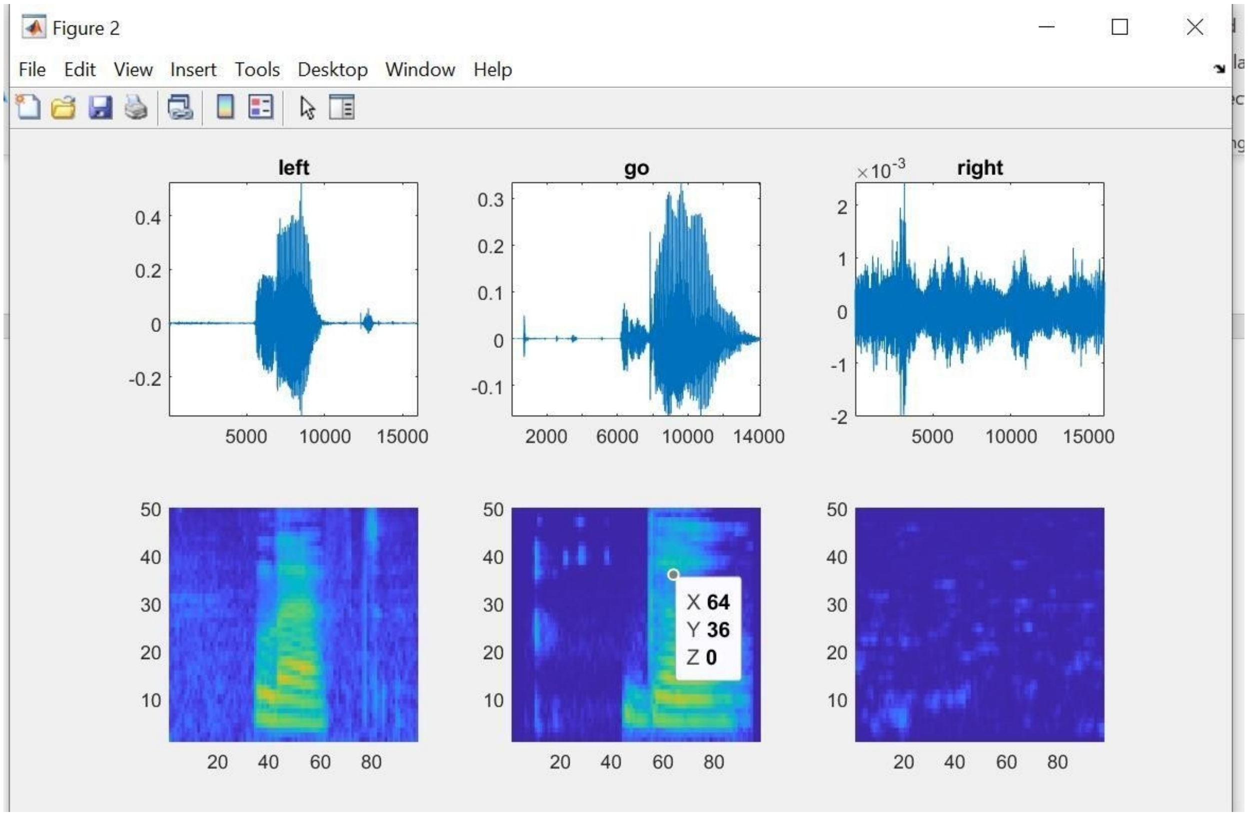
Matlab code for the project:
https://drive.google.com/file/d/1RD6-ZcGl6tkXI4kwKKS3T4yBLUhiBzQv/view?usp=sharing

RESULTS

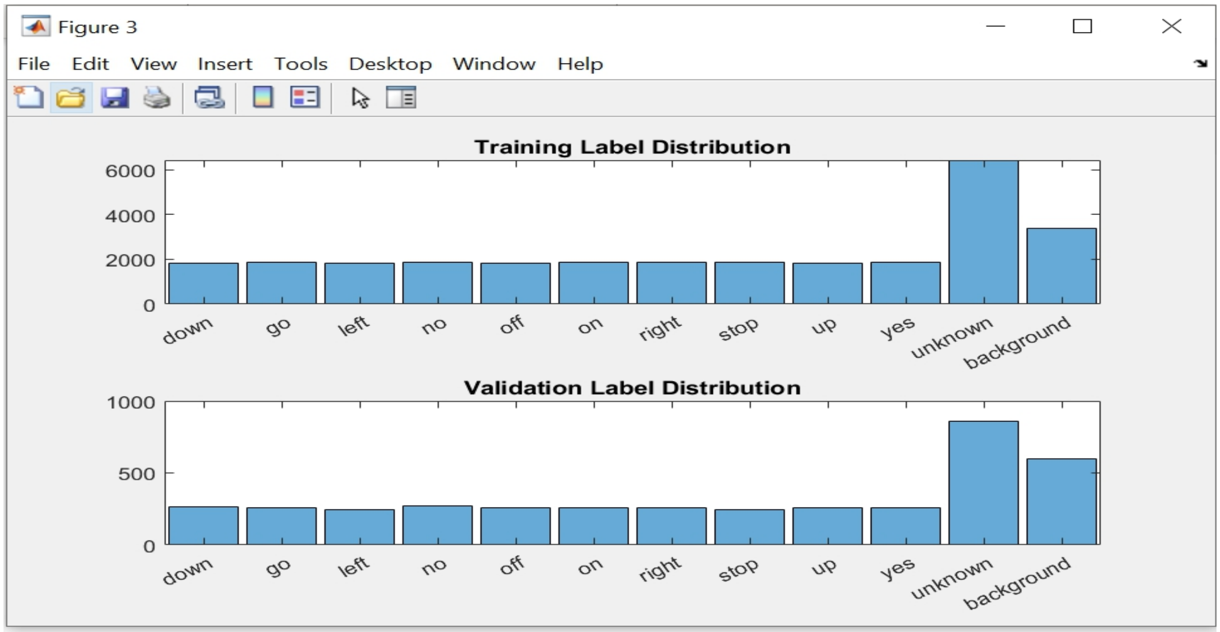
RECOGNITION OF THE SOUND FROM MICROPHONE



RECOGNIZING THE COMMANDS THAT WERE GIVEN



THE DISTRIBUTION OF COMMANDS FOR THE INPUT VOICE

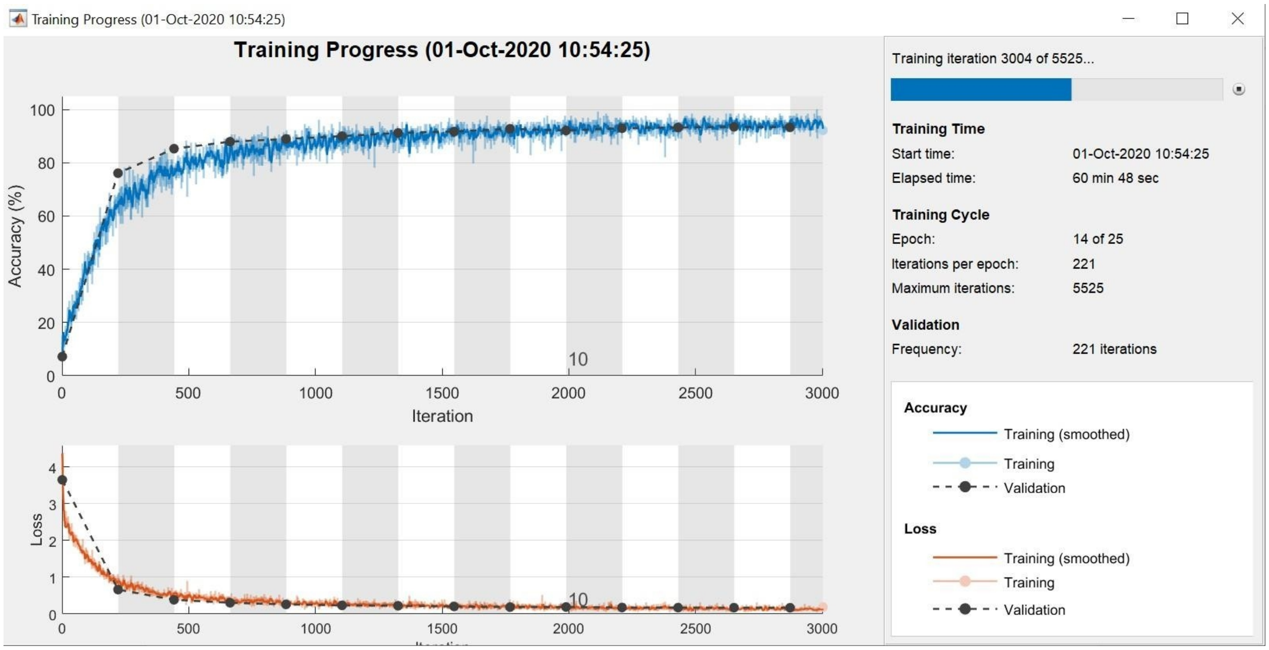


TRAINING PROGRESS

NO OF EPOCHES:25

NO OF ITERATION:5522

THE BELOW IS THE PROGRESS AT THE 14TH EPOCH AND 3004TH ITERATION



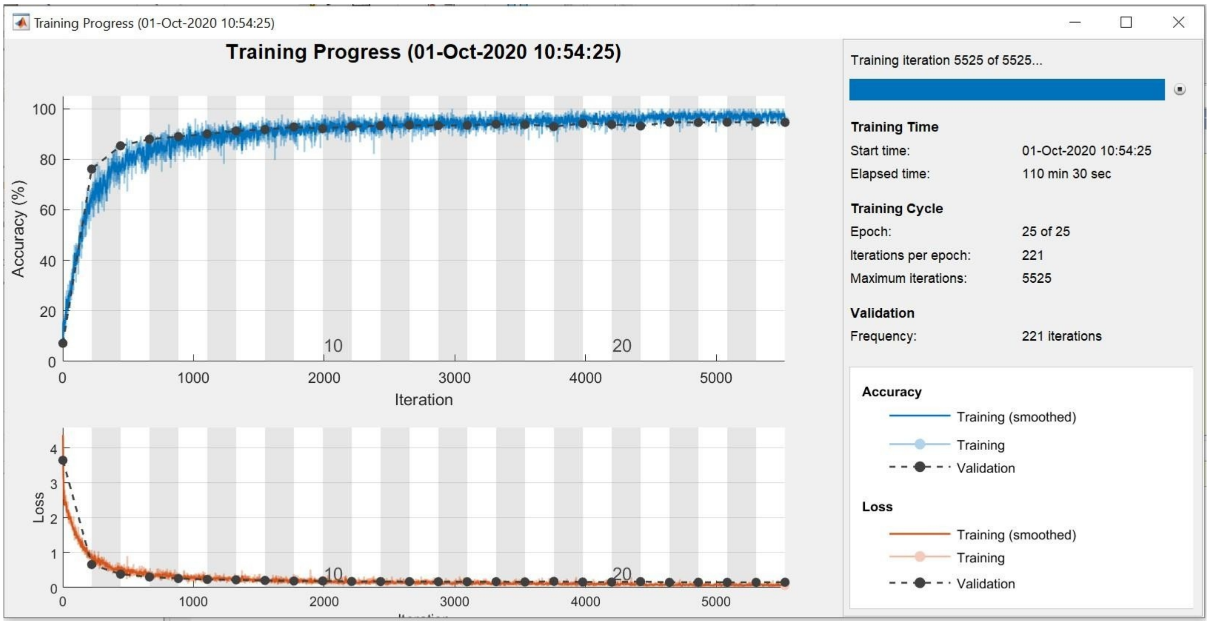
Training iteration 3004 of 5525...

Training Time
Start time: 01-Oct-2020 10:54:25
Elapsed time: 60 min 48 sec

Training Cycle
Epoch: 14 of 25
Iterations per epoch: 221
Maximum iterations: 5525

Validation
Frequency: 221 iterations

END RESULT AFTER TRAINING



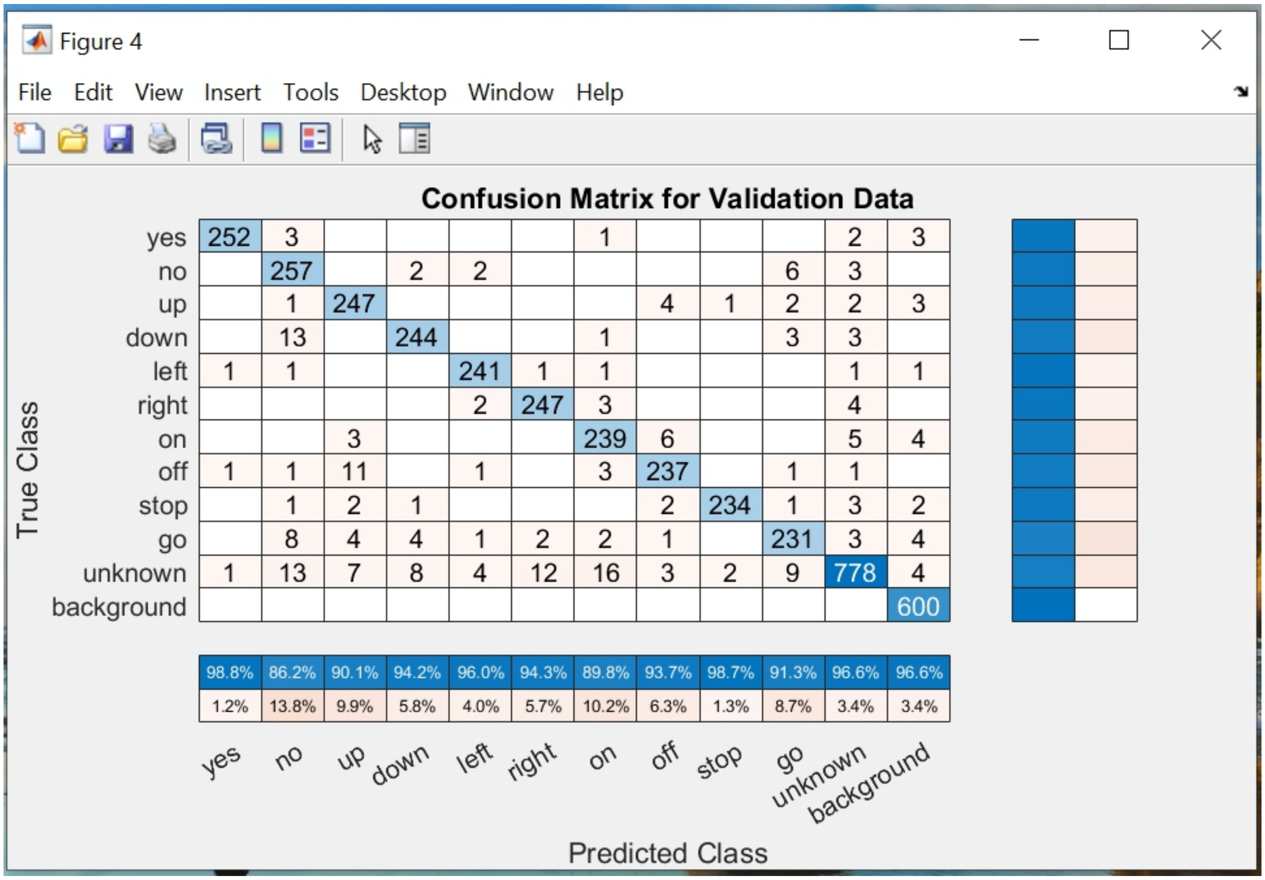
Training iteration 5525 of 5525...

Training Time
Start time: 01-Oct-2020 10:54:25
Elapsed time: 110 min 30 sec

Training Cycle
Epoch: 25 of 25
Iterations per epoch: 221
Maximum iterations: 5525

Validation
Frequency: 221 iterations

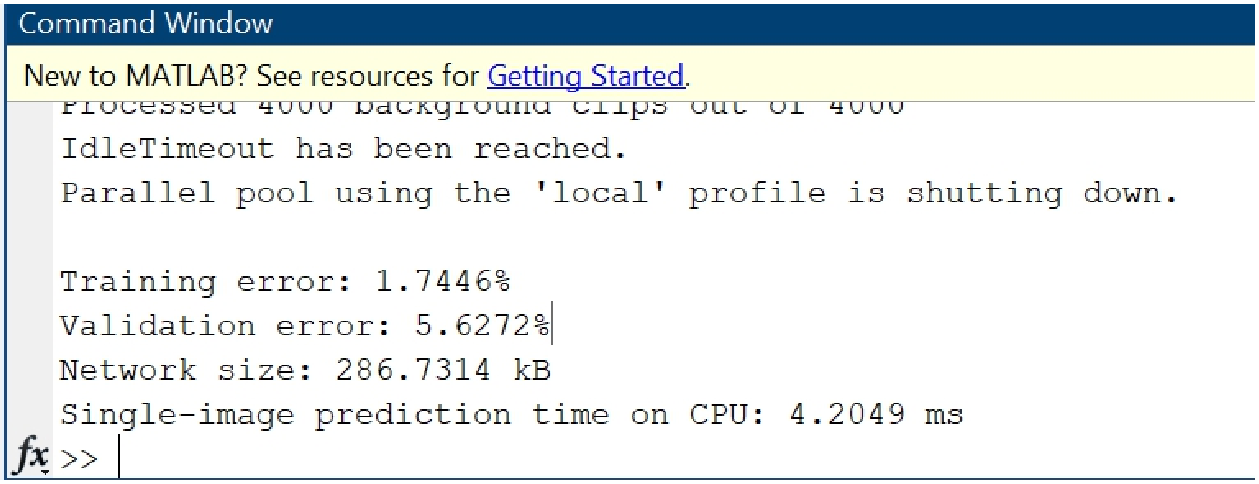
CONFUSION MATRIX TO TEST THE ACCURACY



Confusion Matrix for Validation Data

	yes	no	up	down	left	right	on	off	stop	go	unknown	background
yes	252	3					1				2	3
no		257		2	2					6	3	
up		1	247					4	1	2	2	3
down		13		244			1			3	3	
left	1	1			241	1	1				1	1
right					2	247	3				4	
on			3				239	6			5	4
off	1	1	11		1		3	237		1	1	
stop		1	2	1				2	234	1	3	2
go		8	4	4	1	2	2	1		231	3	4
unknown	1	13	7	8	4	12	16	3	2	9	778	4
background												600
	98.8%	86.2%	90.1%	94.2%	96.0%	94.3%	89.8%	93.7%	98.7%	91.3%	96.6%	96.6%
	1.2%	13.8%	9.9%	5.8%	4.0%	5.7%	10.2%	6.3%	1.3%	8.7%	3.4%	3.4%

OUTPUT IN THE COMMAND WINDOW



```
Processed 4000 background clips out of 4000
IdleTimeout has been reached.
Parallel pool using the 'local' profile is shutting down.

Training error: 1.7446%
Validation error: 5.6272%
Network size: 286.7314 kB
Single-image prediction time on CPU: 4.2049 ms
>>
```

For the entire coomad window code refer the link:

https://drive.google.com/file/d/10OZwf-E3_RsDL_jKTPeWn0sqEzIb2zAQ/view?usp=sharing

DISCUSSION

We were able to differentiate and classify the words like “go”, “stop”, “off”, “on”, “left”, etc which can be useful for industries in instructing a robot for getting worksdone which are highly repetitive, complex, and this system can help in bringing more efficiency.

Loaded the dataset and categorized them into known words and unknowns which would help in noise cancellation and also cancellations of unnecessary words whichweren’t required for getting job done by robots. Then we evolved into process splitting the given data i.e. training, validation and testing.

Training a model with around 64 recordings(2400 app recordings for each word spoke by different types of people) of 1 second each is achieved. Before directly feeding the signals into the neural network (CNN-convolution neural network), formore clear classification and processing, Data processing is done.

The suitable way and best data processing can be done by deploying Time- Frequency transformation which led to image representation of the signals. Hence,it eas possible for us to realize the spectogram plotting with time(sample) and freq(bins#) on x and y axes respectively for better processing and validation. This helps limit the complexity of the network itself and keeps its training process computationally clean.

We now established how to use our data, we are ready to feed it into the neural network. The google actually referenced an old paper(CNN paper), which is suitablefor their dataset. We can express the data in MATLAB as a vector of layers. As the data is fed into the neural network, we can perform convolution, ReLu, and pooling for some number of time required. As we have the network ready, we can proceed in training. Back propagation is used to update the weights according to the error ofthe obtained output from the target output.

The input signal is fed forward and the error signal is fed backward, hence the process gets done. Deep Convolution neural networks include repeating patterns offew different types of layers. This feature made it suitable for command recognition.

Testing the network also means evaluating the network. The best ways to do it include accuracy, confusion matrix etc. The validation accuracy of our data piledupto 95.41%.for 25 epochs each having 220 iterations, made a total of 5500 iterations for validation. Test error is 5.1%, training error is 3.3%.

On evaluating the performance of the network on test data, we've be more analyticand used a confusion matrix as the one shown in results. Taking an example, when supplied an 262 recordings on "left" commands, the network got 246/262 right,
instead there are only 256 predicted "left" to the same 246 recordings, it shows 95.7% in predicted class. This table suggested us the things we improved, which ledto less error in our output.

How can it be used in real world?

1. Working offices

This can be installed work places and can be involved in simpler tasks and can increase efficiency and can do some recursive tasks decreasing the human efforts. These can be used for crating statistics, printing documents on demand, can search files on computer, dictate the data involved in the file, scheduling and recording conferences in offices, and also can replace travelagencies for planning and managing travels.

2. Iot

Iot includes, iot in smart homes, smart lights etc. the major field instance where command recognition is growing more now is in automobile cars. This could enhance way of driving in the intention of reducing distractionscaused to a driver. This can include controlling radio, and can evolve to
bring "command and navigate" process.

3. Retailing and banking purposes

In banking system, this can reduce the difficulties faced by the customers, by reducing the use of human-customer, and can lower workers cost. This can be used for performing transactions, and can be used for enquiring data of balance, transactions etc. In marketing and retailing, this can be useful for companies to get ones slang, vocabulary and can be used for classification among the customers according to their belonging places, and hence can be used for proposing new strategies.

4. Industries

We can deploy command recognition in industries like dairy industries for manufacturing purposes to instruct robots in a very efficient way, so thatthey can reduce the human pressure and effort, and hence can play important role in efficiency factor

CONCLUSION

Since the neural network is now trained.This can be deployed into a robotworking in the industry to efficiently follow the instructor at the workstation.This not only ensures the accuracy of the robot but also helpsthe industry to use humans for more productive work.

The accuracy in the project is satisfactory and there were enough no of epochs performed to train the network ensuring that there wont be any fluctuation in thefuture.

After 5525 iteration and 25 epochs the accuracy is maximum and the loss isminimum hence the model is satisfactory

References

1. [1] Warden P. "Speech Commands: A public dataset for single-wordspeech recognition", 2017. Available from https://storage.googleapis.com/download.tensorflow.org/data/speech_co m mands_v0.01.tar.gz. Copyright Google 2017. The Speech Commands Dataset is licensed under the Creative Commons Attribution 4.0 license, available here: https://creativecommons.org/licenses/by/4.0/legalcode.
2. https://www.researchgate.net/publication/330815113_Speech_Recognitio n_Using_Dee p_Neural_Networks_A_Systematic_Review
3. https://www.researchgate.net/publication/335106017_Speech_Recognitio n_Using_MA TLAB_and_Cross-Correlation_Technique

]

4. D. P. Mital and G. W. Leng, "A voice-activated robot with artificial intelligence,"*Robotics and Autonomous Systems*, vol. 4, no. 4, pp. 339- 344, 1989.
5.] S. Davis and P. Mermelstein, "Comparison of parametric representationsfor monosyllabic word recognition in continuously spoken sentences,"*In IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no.4, pp. 357-366, 1980.
6.] D. Marquardt, "An algorithm for least-squares estimation of nonlinearparameters,"*SIAM Journal on Applied Mathematics*, vol. 11, no. 2, pp. 431441, 1963.
7.] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, "Discrete-Time SignalProcessing. 2nd Ed,"*Upper Saddle River, NJ: Prentice Hall*, 1999.
8. Y.-H. Tu, J. Du, L. Sun, F. Ma, H.-K. Wang, J.-D. Chen, and C.-H. Lee, "An iterative mask estimation approach to deep learning based multichannelspeech recognition," Speech Communication, vol. 106, pp. 31– 43, 2019.
9.) Lakkhanawannakun, P., & Noyunsan, C. (2019). Speech Recognition usingDeep Learning. 2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC). doi:10.1109/itc-cscc.2019.8793338
10.] Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Cheng, Q., Chen, G. and Chen, J., 2016, June. Deep speech 2: End-to-end speech recognition in english and mandarin. In International Conference on Machine Learning (pp. 173-182).