DEEP LEARNING-BASED VISUAL SPEECH RECOGNITION SYSTEM USING MARATHI DIGIT

Kiran Surywanshi
Department of Computer Science, (Dr. B.A.M.U.), Ch. Sambahaji Nagar, India.
Email: suryawanshikiran@gmail.com

Krishna Shinde
Derpartment of Computer Science, (Dr. B.A.M.U), Ch. Sambahaji Nagar, India.
Email: shreekriss@gmail.com

Dr. Charnsing Kayte
Department. of Digital & Cybeer Forinsic, (Dr. B.A. M.U.), Ch. Sambahaji Nagar, India.
Email: charankayte@gmail.com

Abstract

Deep learning techniques are a potent AI tactic that has greatly aided the advancement of visual speech learning. A machine being able to perform lip-reading would have been believed inconceivable a few decades ago. But, in the last several years, machine learning has grown exponentially, making it possible for a machine to understand human speech just from visual inputs. In the absence of audio, visual speech recognition converts speech to text by extracting a person's lip features to follow the pattern that is created. In this paper, developing an automatic Marathi visual speech digit recognition system. In this study, we have used our own collected CNKR database. This collected database applied such pre-processing techniques as normalization, video-to-frame, farm-to-face detection, face-to-lip detection, and concatenation of lip movement images then used CNN, VGG16, and VGG19 deep leering techniques for Marathi Digit recognition from silent video.

Keywords: VGG, Data augmentation, Lip Extraction, Speech.

I. INTRODUCTION

Speech is a key criterion for communication since it is simple and everyone can speak without the use of any equipment, and it does not require a technical skill set. The issue with primitive interfacing devices is that some fundamental degree of skill set is required to use those interfaces. As a result, interacting with such gadgets will be challenging for those who are unaware of their technical skill set. Presently, typical technical concerns involve computer usage, such as how well computers connect and how user-friendly less conventional ways are. Knowing English literature has practically become a need for interacting with computers and using information technologies. As information technology advances, it is increasingly important for ordinary people to stay on the cutting edge of technological advancement. Aside from this limitation, the most accessible system, such as devices that can read and accept input as the speech of regional languages and respond to those regional things, will need to be created for the most user-friendly system [1,2]. Aside from this limitation, the most accessible system, such as devices that can read and accept input as the speech of regional languages and respond to those regional things, will need to be created for the most user-friendly system. Acoustic noise and disruptions in a loud setting will not affect this. Visible speech is an intriguing study topic that has mostly been employed in sectors such as improving applications in human-computer interface, security, and digital entertainment [3,4]. The aforementioned factors prompted academics to study specific VSR (visual speech recognition), as well as AVSR (audio-visual speech recognition). This is referred to as the automated lip-reading approach for visual speech recognition. Many automated speech recognition algorithms that integrate both audio and visual characteristics have been presented recently. An essential goal of visual speech recognizers in all such systems is to enhance recognition accuracy, particularly in noisy environments [5,6]. Nowadays, neural network approaches have huge outcomes on social problems, using artificial intelligence techniques to tackle numerous difficulties. Visible speech recognition is a useful approach for the deaf and dumb [7]. Subsequently, in 1980, Petejan invented a new lip contour reading technology. In the recognition model, a pixel-based technique paired with an artificial neural network (ANN) was developed in 1989 [8]. The traditional lip-reading approach is divided into two stages: feature extraction and categorization. The first phase is to extract pixel values as visual information from the mouth region of collected photos, and the second step is to identify these features using classification algorithms. Lip-reading is a way of understanding speech that involves monitoring and analysing the movement of the lips, face, and other social clues. Speech recognition is exceedingly challenging in noisy circumstances, and visual speech recognition can open the way for the development of aiding technologies [9]. The Visual speech recognition system refers to a sophisticated feature-based analysis of the lips and their surroundings. Because of the importance of the external environment and the details that play a role in prediction, it involves many elements of feature extraction [10]. Deep Convolutional Neural Networks (Deep CNNs) have reached state-of-the-art performance in most computer vision tasks because they can extract solid visual feature representations from massive amounts of data via an end-to-end learning process [11]. Hence, the introduction of deep architectures can improve the efficiency of recognition systems in the LR problem and even outperform human performance [12]. In this study, we used Deep Learning Transfer Learning to create a Marathi Digit Visual Speech Recognition System. The study is divided into four sections: introduction, literature review, proposed methodology, pre-processing, database information, and experimental analysis and outcomes.

II. LITERATURES SURVEY

Table 1 provides a full review beginning with the author, year of publishing, Methods, About Database, Types, and Language, and ending with the recognition result.

TABLE I. LITERATURES SURVEY

Sr. No.	Reference and Year of Publication	Model	Database	Type	Language	Accuracy
1	Marcheret et al. [13], 2007.	DCT+LDA+MLLT+HMM	IBMIH	Digit	English	63.00%
2	Lucey et al. [14], 2008.	DCT+PCA+HMM	IBMSR	Digit	English	66.21%
3	Wang et al. [15], 2008.	ASM+HMM and RDA	Own	Digit	English	67.32% 55.57%
4	Zhou et al. [16], 2009	LBP-TOP+SVM	OuluVS	Phrases	English	62.40%
5	Pass et al. [17], 2010.	MCPV, DCT+HMM	QuLips	Digit	English	98.00%
6	Navarathna et al. [18], 2011.	DCT+PCA+HMM	AVICAR	Digit	English	25.00%
7	Estellers et al. [19], 2011.	DCT+LDA+HMM	Own	Digit	French	71.00%
8	Huang et al. [20], 2013.	DCT+LDA+HMM, DCT+LDA+DBN	Own	Digit		35.20% 35.70%
9	Bear et al. [21], 2014.	AAM+HMM	AVLetters	Alphabets	English	35.00%

10	Stewart et al. [22], 2014.	DCT+MS-HMM	XM2VTS	Digit	English	70.00%
11	Noda et al. [23], 2015.	CNN+MS-HMM	ATR	Word	Japanese	22.50%
12	Sui et al. [24], 2015.	DBM+DCT+LDA+HMM	AusTalk	Digit	English	69.10%
13	Ninomiya et al. [25], 2015	DBN+MS-HMM	CENSREC-1-AV	Digit	Japanese	22.50%
14	Lee et al,[26], 2015.	CNN_LSTM	OuluVS2	Phrases	English	81.10%
15	Rekik et al. [27], 2016	HOG_MBH	MIRACL-VC1 CUAVE OuluVS	word Digit Phrases	English	96.00% 90.00% 93.20%
16	Saitof et al. [28], 2016.	CFI+NN	OuluVS2	Phrase	English	81.10%
17	Sui et al. [29], 2017	CHAVF+HMM	AusTalk	Digit	English	69.18%
18	Wand et al. [30], 2017.	FeedForwrd+LSTM	GRID	Phrases	English	42.00%
19	Petridis et al. [31], 2018.	Autoencodor_Bi-LSTM	AVDigit	Digit	English	68.00%

III. PROPOSED METHODOLOGY

The general procedure of the automatic Marathi visual speech digit recognition method is shown in Figure 1. Training and Testing are the two phases of the AVSR for the Marathi digit recognition technique. During the training phase, the video dataset is transformed into a frame, and the class-labeled digits images are then expanded to 224*224 pixels and used to train the networks using the proposed architecture. The trained model will be tested using the AVSR's Marathi digit recognition process. Following the application of the AVSR for the Marathi digit recognition procedure, loading the test video, converting to frame, detecting face, detecting lip, concatenating lips images, and resizing to fit the input model, a threshold is applied to discard the digit labels with confidence scores below the threshold and pass the digit labels with confidence scores above the threshold. The model's inaccurate classifications will be reduced as a result of this operation. The AVSR output of the Marathi digit recognition system, the digit label, is then shown on the screen.

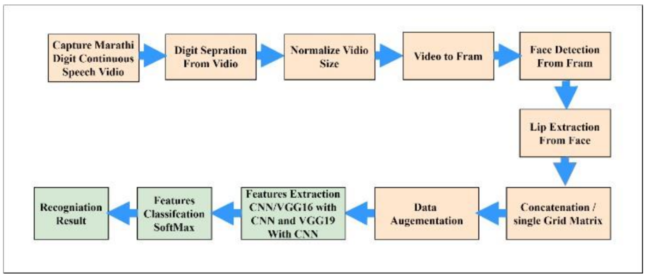

Capture Marathi Digit Continuous Speech Vidio
Digit Sepration From Vidio
Normalize Vidio Size
Video to Fram
Face Detection From Fram
Lip Extraction From Face
Concatenation / single Grid Matrix
Data Augementation
Features Extraction CNN/VGG16 with CNN and VGG19 With CNN
Features Classifcation SoftMax
Recogniation Result

Fig. 1. Automatic Marathi Visual Digit Recognition system

A. About Database

In this study, we have used our own collected database. This database was created in the Forensic Cyber Computer Lab at the Government Institute of Forensic Science Aurangabad. The collected data contains 1 to 10 Marathi-digit continuous speech videos. At the time of collection of datasets used such standees the frontal facial position, 60 frames per second video, controlled room environment, same lighting condition, 80 cm between the individual and the video recording equipment, and black backdrop dataset standards were used. A total of ten persons were given their samples in two videos with a 16:9 aspect ratio and a resolution of 1920*1080 using the standard smartphone one plus, which has a 48-megapixel camera and an IMX586 Sony sensor. According to Schaller's research, students and employees, both men and women, are between the ages of 19 and 45.

B. Pre-processing

Using the obtained CNKR database, we performed the following pre-processing steps: normalization, video-to-frame, frame-to-face detection, face cropping, lip detection, and data augmentation techniques.

1) Normalization

The database Normalization guarantees that comparisons between data-collecting techniques and texture instances are as accurate as possible. For imaging modalities that do not correlate to absolute physical quantities, pixel value normalization (intensity) is advised. Many sophisticated ways for normalizing values have been presented, and they are frequently modality-specific [32]. 1 to 10 continuous Marathi digits video dataset developed with huge video pixel size and wide backdrop, unable to extract finer spatiotemporal characteristics due to lack of normalization. By cropping, the 123apps online tool separated each digit video and dataset, in which this 1920*1080-pixel movie was reduced to 608*1080 pixels [33].

2) Video To Frame Extraction

Frame extraction is critical in breaking down a large video sequence into smaller units for subsequent processing. Effective frame extraction implies gathering key video information and sending it over the network so that network stress load is minimal, which is why video data sharing is simple and quick, and there is a growing emphasis on image and video processing technology [34]. Scikit is a Python video processing library that supports a variety of processing techniques. Skvideo.io is a video read/write a module that runs on top of FFmpeg/LiBAV. It will use the proper probe to parse the video information. The FFmepg Reader function of skvideo.io turns the video into a series of video shapes in terms of the number of frames, height, width, and channels per pixel. These frames are utilized in subsequent procedures. [35].

3) Frame-to-Face Detection

One of the initial processes in our VSR system is face detection. [36]. Face detection is a subset of object detection that includes automated face recognition. Face detection is a computer vision method that detects human frontal faces from digital photos or movies, and we use the Dlib face detection module. The user ID and other unnecessary information from the image frame are ignored by VSR systems, which are primarily concerned with the mouth regions [36,37 The proposed VSR system employs Dlib, a cross-platform library written in C++ that integrates a variety of machine learning techniques. The following are the steps for identifying faces using open cv Dlib: Use the function dlib.get frontal face detector to detect faces. When there is no face in the frame, it is removed, and the frame with a single face and a suitable proportion of face is retained [35,38].

4) Face cropping

Cropping generates a new picture for the specified area. The picture has been cropped to eliminate the extraneous region and backdrop [39]. Face cropping is the technique of teaching the neural network only the characteristics of facial pictures. Face cropping has the benefit of cutting off the face with varied resolutions dependent on the distance of the face. [37]. proposed system based on open cv Dlib: Use the dlib function. get a frontal face detector and crop the picture of the face.

5) Lip Extraction

The suggested method's goal is to locate automated face and facial 68 landmark points detection. The Dlib library was used to detect 68 key points on the face, and the lip region was removed and suitably enlarged from the marked key points. The identified frames show that the range of lips is 48 to 68 points. and lip area extraction from a cropped facial picture using Python's Dlib module [35,38].

6) Lip Frame Concatenation

The term "concatenate image" refers to the merging of two or more pictures. Concatenated frame image (CFI) is a novel sequence picture encoding approach, as is the data augmentation method for CFI.CFI is straightforward, yet it provides spatial-temporal information for a whole image sequence. The proposed approach was tested using a database that was built by the author. This is a visual speech dataset from ten participants. [40]. The term "concatenate image" refers to the merging of two or more pictures. This may combine any image, regardless of its pixels or image type, such as 'jpeg,' 'png,' 'gif,' 'tiff,' and so on. Using the Python image package Pillow, you may connect two or more pictures. The suggested VSR system deals with the number of frames generated by videos. Since frames are made for each moment of speech, all of these frames are connected in a grid matrix using the pillow python package. It is used to extract features [40,38].

7) Data Augmentation

Deep learning uses an image pre-processing technology called data augmentation to overcome the overfitting problem. Essentially, these tactics have been used to artificially enhance the size of tiny datasets. Conventional data augmentation alterations include image, color, rotation, reflection, and scale adjustments. Moreover, geometrical artifacts such as brightness, histogram equalization, white balancing, sharpening, and blurring are employed. The Keras library allows you to improve data to better suit the model. These approaches, which include data augmentation operations like rotation, sharing range, zoom, brightness, scale, and horizontal Filip [35,40,41], have been utilized on the CNKR Marathi Digit Concatenated pictures databases.

C. Features Extraction

1) CNN

The convolutional neural network is an ANN model that needs to function in many layers. This strategy is commonly used in deep learning. This model is inspired by the biological visual cortex and demonstrates a wide range of chrematistic reactions to different brain neurons. This method has been used to solve a variety of image classification problems, including pedestrian detection, action recognition, digit recognition, face recognition, object detection, and image captioning [42,43]. The CNN model has the following Layers.

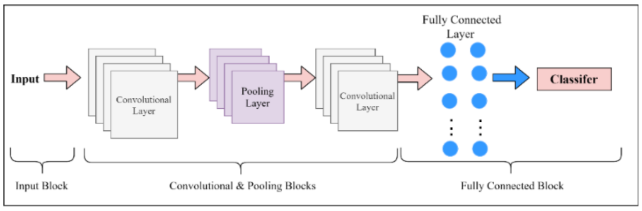


Fig. 2. CNN Model Block Diggram

Convolution Layer: Convolutional layers are used to extract picture characteristics. Each convolutional "filters" or "kernels" are a set of weighted matrices that travel through the input pictures to pick out certain information. CNN's early layers of filters are used to recognize colors and basic patterns.

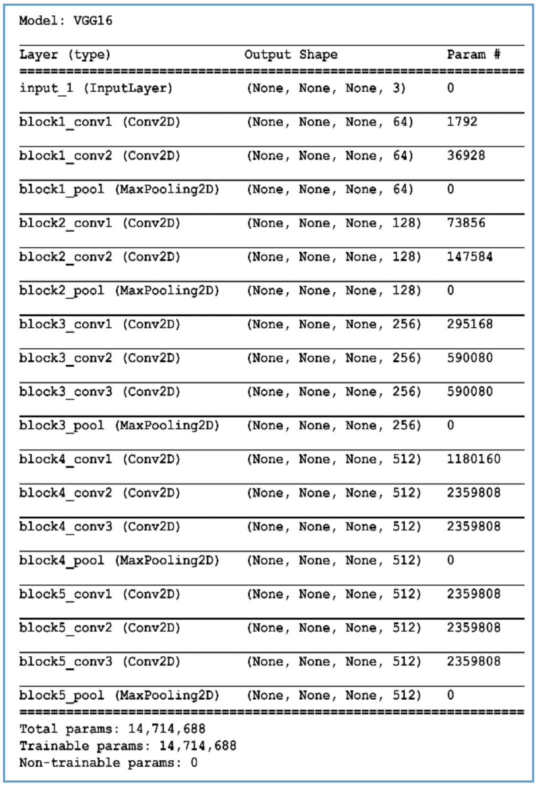
ReLU Layer: The rectified linear activation function is used to turn the negative value into zero in the ReLU Layer.

Pooling Layer: The kernel map's pooling layer assembles the neighboring outputs. With the exception of depth, this layer minimizes the size of the activation maps and transmits the information to the following layer. The computation's complexity is lowered by reducing the spatial dimension and the number of parameters in the image. No parameters are introduced since this layer applies a fixed function to the input. Weight is not changed while backpropagating. Due to the translational invariance given by this layer, the picture can still be identified even if its position changes somewhat.

Fully Connected Layer: The last pooling layer's flattened output serves as the input for a fully connected layer. This layer, which functions in the same way as a standard neural network layer, connects every neuron from the layer above to the one below. The fully connected layers use the classifier to calculate the classification conclusion after integrating the data into a one-dimensional vector. This layer has more parameters than the convolution layer. This fully connected layer is linked to the output layer, which is often a classifier that provides a probability score for the number of classes.

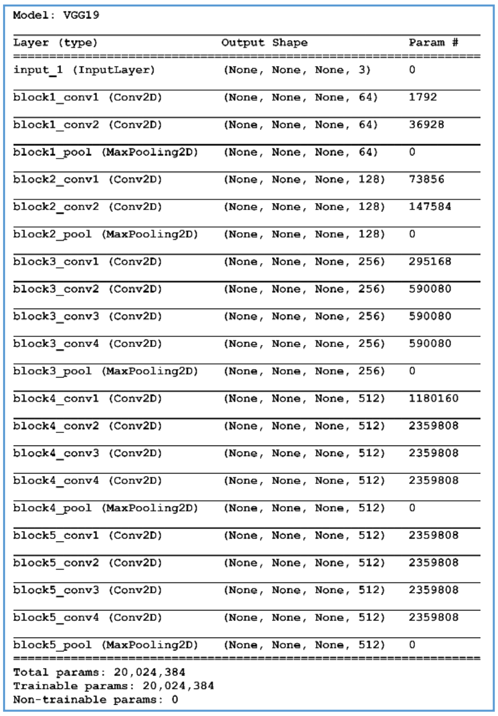
2) VGG16 and VGG19

The VGG deep CNN models were developed by Oxford University's Visual Geometry Group. The well-known models VGG16 & VGG19 each have 16 or 19 layers. The ImageNet database has been used extensively to train the VGG16 model. Around 20000 categories and approximately 14 million photographs make up this vast collection. The VGG16 model has five convolutional blocks. This includes a max-pooling layer and two convolutional layers (size: 3X3) in each convolution block (size: 2X2). Both the prediction and classification tasks are carried out by the fully connected (FC) layers [44, 45]. Figure 3 depicts the summary of the VGG16, and 4 shows the summary of VGG19.



```
Model: VGG16
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, None, None, 3)     0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, None, None, 64)    1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, None, None, 64)    36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, None, None, 64)    0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, None, None, 128)   73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, None, None, 128)   147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, None, None, 128)   0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, None, None, 256)   295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, None, None, 256)   0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, None, None, 512)   1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, None, None, 512)   0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, None, None, 512)   0
=================================================================
Total params: 14,714,688
Trainable params: 14,714,688
Non-trainable params: 0
```

Fig. 3. VGG16 Model Summary



```
Model: VGG19
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, None, None, 3)     0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, None, None, 64)    1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, None, None, 64)    36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, None, None, 64)    0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, None, None, 128)   73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, None, None, 128)   147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, None, None, 128)   0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, None, None, 256)   295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_conv4 (Conv2D)        (None, None, None, 256)   590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, None, None, 256)   0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, None, None, 512)   1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_conv4 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, None, None, 512)   0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_conv4 (Conv2D)        (None, None, None, 512)   2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, None, None, 512)   0
=================================================================
Total params: 20,024,384
Trainable params: 20,024,384
Non-trainable params: 0
```

Fig. 4. VGG19 Model Summary

D. Features Classification and Matching

SoftMax: A very crucial task is the object's classification. In contrast to SVM-based classifiers, which predict class labels based on calculated probabilities, SoftMax classifiers predict class labels based on classification scores based on hyperplanes that divide the data into two groups. We calculated the probabilities in the preceding layer and obtained the value for label calculation before using the fully linked layer for the final score calculation. The activation function for multi-class classification at the output layer that is most frequently used is the SoftMax function. The SoftMax function computes the probability distribution for real numbers and returns a value between 0 and 1, with the sum of the probabilities equating to 1.

IV. EXPERIMENTAL SETUPS

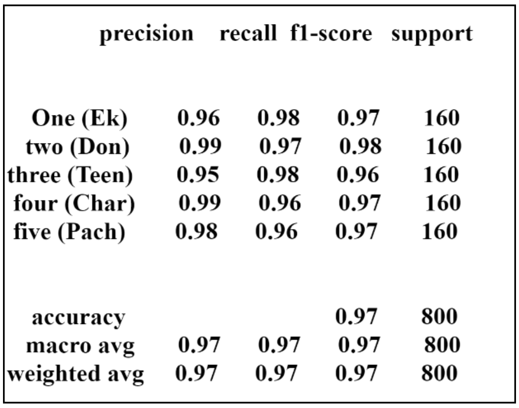
The suggested approach uses Python 3.6, together with other open-source library tools like Keres, TensorFlow, CUDA, and image processing libraries like OpenCV, matplotlib, and scikit-learn, among others, to build deep learning models. The training model was created using the Jupyter notebook IDE, and the pre-processing was carried out using the spider IDE. The approach operates on a laptop running Windows 11 with an Intel Core-i5 CPU, NVidia 2GB of memory, and 8GB of RAM. For this work, we made use of databases on Marathi Digit 10 that we had amassed on our own, and we proposed a pre-train transfer learning method using silent video and the VGG16 with CNN Marathi digit Identification System. In our study, we use VGG16 with the convolutional neural network to extract hidden characteristics from video farms pictures. The batch size is 64, the epochs are 7 and 8, the loss function employs categorical cross-entropy, Adam is the optimizer, and the SoftMax classifier is used for the activation function. In our initial experiments, we divided the database into different proportions, such as (80:20) and (60:40), then evaluated the system's performance using these proportions and personally tested all of the photographs. Several data percentages were used for the studies, and we obtained good recognition accuracy in (80:20) As a result, we divided the dataset into two portions: 20% for model validation and 80% for training.

V. PERFORMANCE ANALYSIS

In this study, we have to Calculate the various performance analysis techniques used such as Classification matrix, Confusion matrix, Modal Training accuracy, and modal training loss, etc.

A. Classification and Confusion Matrix

The CNKR VSR Marathi digit database was used to test the face and fingerprint score level fusion multimodal biometric identification system, and the results were assessed using several evaluation metrics, including Confusion matrix, Precision, Recall, Support, Micro and weighted avg, and F1 Score. Precision depicts the model's positive predictive value, whereas recall depicts its sensitivity and true positive rate. We utilized micro-averages to integrate the findings across the thirty categories to get overall accuracy and recall. Fig 5 shows the classification report of precision (P) and recall (R) rate of overall classes and fig 6 shows the confusion matrix of the Automatic Marathi Visual Speech Digit Recognition system score of 10 classes for Marathi digit identification result.



	precision	recall	f1-score	support
One (Ek)	0.96	0.98	0.97	160
two (Don)	0.99	0.97	0.98	160
three (Teen)	0.95	0.98	0.96	160
four (Char)	0.99	0.96	0.97	160
five (Pach)	0.98	0.96	0.97	160
accuracy			0.97	800
macro avg	0.97	0.97	0.97	800
weighted avg	0.97	0.97	0.97	800

Fig. 5. Classificatioin Matrix of VGG19 with CNN (Split Data 80:20)

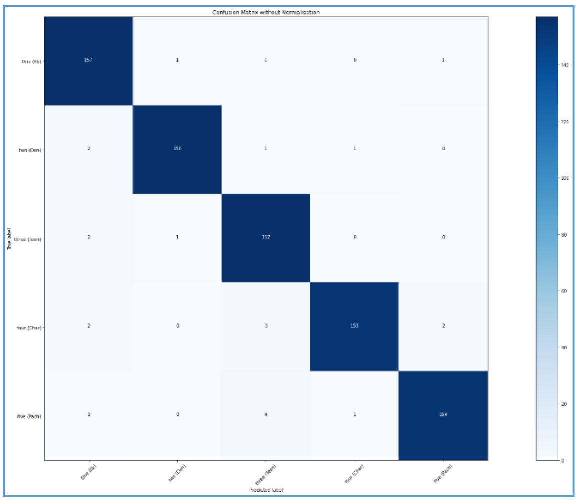


Fig. 6. Confusion Matrix Without Normalize

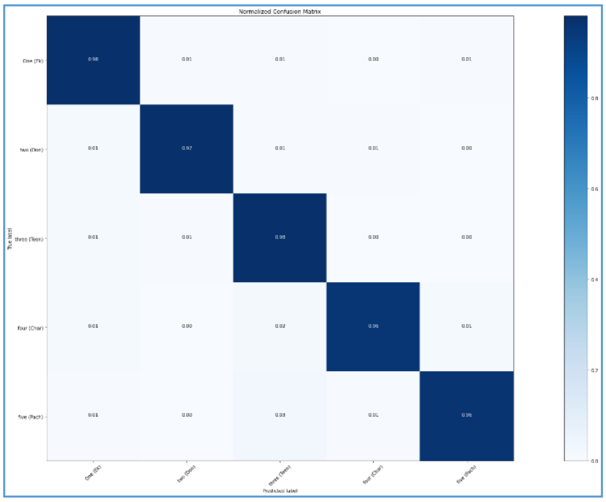
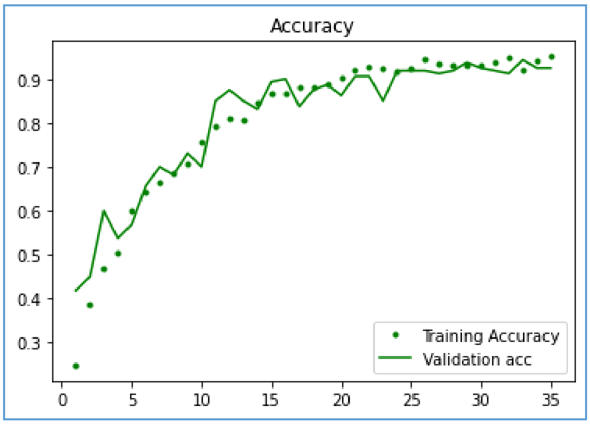


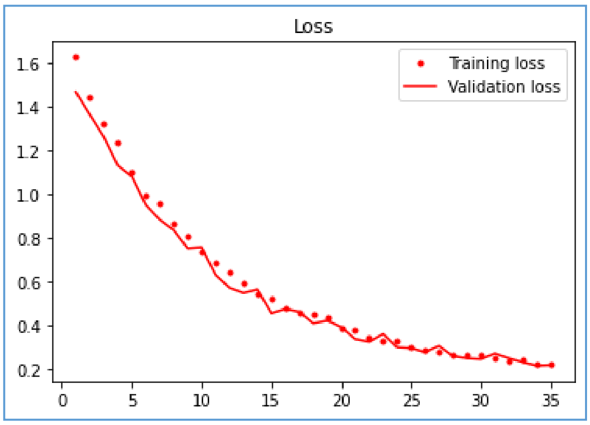
Fig. 7. Confusion Matrix Normaize

VI. Result and Discussion

Here, is the accuracy of the proposed CNN, VGG16 with CNN, and VGG19 with CNN model. In this study, we have used the CNKR Marathi Digits video database. Table 2 shows the CNN, VGG16 with CNN, and VGG19 with CNN model automatic Marathi visual speech Digit recognition system result. In the experimental work, CNN model database splitting was made at 80:20 and got 87.63% recognition and database splitting was 60:40 got 76.38%, In VGG16 with CNN database splitting was made at 80:20 got 91.87% and splitting was 60:40 got 80.75% accuracy



Graph 1 Training and validation Accuracy



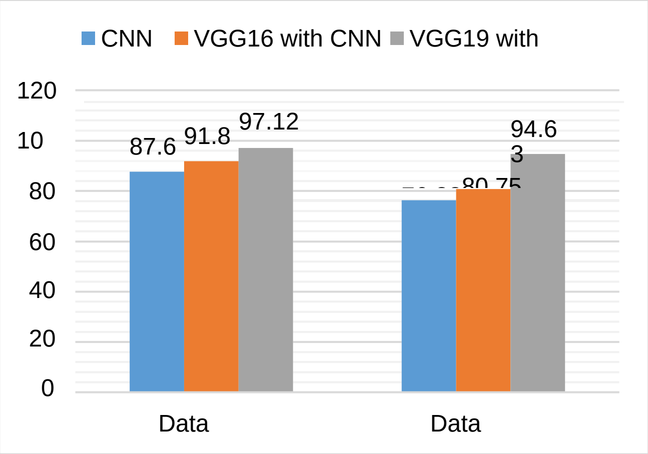
Graph 2 Training and Validation Loss

and lastly VGG16 with CNN database splitting was 80:20 got 97.12% and splitting was 60:40 got 94.63% accuracy. In this study, we have got AVSR Marathi digits recognition system got a good recognition accuracy in using database splitting 80:20 using VGG19 with CNN model. Graph 1 and 2 shows the modal training and validation loss and accuracy of VGG19 with the CNN model modal.

Graph 3 shows the comparative analysis of CNN, VGG16 with CNN, and VGG19 with CNN. We have got in VGG19 with CNN model split was made (80:20%) database has 97.12% recognition accuracy than CNN and VGG16 with CNN. Graph 3 shows the Comparative Analysis of CNN, VGG16 with CNN, and VGG19 with CNN.

TABLE II. COMPARRATIVE ANALYSIS RESULT

Data	TR: VA	Model	Loss	Time	Accuracy
CNKRG	80:20	CNN	0.4864350152015686	0:00:23.955156	87.63%
CNKRG	60:40	CNN	0.7071377825737	0:00:22.157533	76.38%
CNKRG	80:20	VGG16 with CNN	0.5962432909011841	0:03:41.701994	91.87%
CNKRG	60:40	VGG16 with CNN	0.7574836349487305	0:03:34.254496	80.75%
CNKRG	80:20	VGG19 with CNN	0.1528753137588501	0:06:05.995623	**97.12%**
CNKRG	60:40	VGG19 with CNN	0.29596620440483096	0:04:56.374521	94.63%



Graph 3 Comparative Analysis of the Automatic Marathi Visual Speech Digit Recognition System

VII. CONCLUSION

In this study, we have used deep learning techniques for the Automatic Marathi visual speech Digit Recognition System. We have used our own collected CNKR Marathi digit video database. First, we applied Normalization techniques on digit-collected video for separation of each digit from the video, then used Video to frame, Farm Face detection, face detection to Lip extraction, concatenation techniques for concatenating all extracted lip movement to images, and lastly applied data augmentation for artificially expand the size of the database. In this study, we have used ten subject databases and proposed a system using CNN data split made the first time 80:20 got 87.63% accuracy and the second time 60:40% got 76.38% accuracy. VGG16 with CNN data split was made 80:20% got 91.87% accuracy and 60:40% got 80.75% accuracy. In the VGG19 data split 80:20 got 97.12% accuracy and 60:40% accuracy got 94.63%. The comparative analysis of CNN, VGG16 with CNN, and VGG19 with CNN got in good recognition Accuracy in the VGG19 with CNN data splitting wad 80:20 got 97.12%.

ACKNOWLEDGMENT

The author acknowledges and thanks CSMNRF, Pune for financial support “SARTHI” an Autonomous Institute Government of Maharashtra.

REFERENCES

[1] Gerasimos Potamianos, Chalapathy Neti, Guillaume Gravier, Ashutosh Garg and Andrew W. Senior “Recent Advances in the Automatic Recognition of Audio-Visual Speech” IEEE, Vol. 12, 2013.

[2] Lewis, T. W. and D. M. W. Powers. “Audio-Visual Speech Recognition using Red Exclusion and Neural Networks” Journal of Research and Practice in Information Technology, Vol. 35, Issue 1, 2003.

[3] Piotr Dalka and Andrzej Czyzewski “human-computer interface based on visual lip movement and gesture recognition” International Journal of Computer Science and Applications, Vol. 7 No. 3, pp. 124 – 139, 2010.

[4] Rajitha Navarathna, Patrick Lucey, David Dean, Clinton Fookes, and Sridha Sridharan. “Lip Detection for Audio-Visual Speech Recognition In-Car Environment” International Conference on Information Science, Signal Processing and their Applications, 2010.

[5] Siew Wen Chin, Li-Minn Ang, and Kah Phooi Seng “Lips Detection for Audio-Visual Speech Recognition System” International Symposium on Intelligent Signal Processing and Communication Systems, 2008.

[6] Yong-Ki Kim, Jong Gwan Lim, and Mi-Hye Kim “Comparison of Lip Image Feature Extraction Methods for Improvement of Isolated Word Recognition Rate” Advanced Science and Technology Letters Vol. 107, pp. 57-61, 2015.

[7] Sooraj V, Hardhik M and Nishanth S Murthy “Lip reading technique - A Review” International Journal of Scientific & Technology .2020.

[8] Yuanyao Lu and Hongbo Li "Automatic lip-reading system based on Deep convolutional neural network and attention based Long short-term memory" International Journal of Applied Science. 2019.
[9] Priti Yadav, Priyanka Yadav and Vishal Sharma "Lip reading using neural networks" International Journal of Computer Applications and mobile computing. 2014.
[10] Dhairya Desai, Priyansh Parikh, Priyesh Agrawal and Mr. Piyush Kumar Soni "Visual Speech Recognition" International Journal of Engineering Research & Technology (IJERT), Vol. 9 Issue 04, April-2020.
[11] K. He, X. Zhang, S. Ren and J. Sun "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
[12] J. S. Chung and A. Zisserma "Lip Reading in the Wild" in Asian Conference on Computer Vision, 2016.
[13] Etienne, Marcheret and Vit Libal, Gerasimos Potamianos "dynamic stream weight modeling for audio-visual speech recognition" IEEE International Conference on Acoustics, Speech, and Signal Processing - ICASSP 2007, IV-945–IV-948. doi:10.1109/icassp.2007.367227
[14] P. J. Lucey, G. Potamianos and S. Sridharan "Patch-based analysis of visual speech from multiple views" in Proc. International Conference on Auditory-Visual Speech Processing 2008. http://www.isca-speech.org/archive_open/avsp08.
[15] S.-L. Wang, A. W.-C. Liew, W. H. Lau and S. H. Leung "An automatic lipreading system for spoken digits with limited training data, Circuits, and Systems for Video Technology" 18 (12) (2008) 1760–1765.
[16] Guoying Zhao, Barnard, M., Pietikainen M. "Lipreading with Local Spatiotemporal Descriptors" IEEE Transactions on Multimedia,11(7),1254–1265. doi:10.1109/tmm.2009.2030637. 2009.
[17] Adrian Pass, Jianguo. Zhang, Darryl Stewart "An investigation into features for multi-view lipreading" in Proc.17th IEEE International Conference on Image Processing, pp. 2417–2420. doi:10.1109/ICIP.2010.5650963. 2010
[18] R. Navarathna, T. Kleinschmidt, D. B. Dean, S. Sridharan, P. J. Lucey, "Can audio-visual speech recognition outperform acoustically enhanced speech recognition in the automotive environment?" in Proceedings of Interspeech, 2011, pp. 2241– 2244.2011.
[19] Virginia. Estellers, J.-P. Thiran, "Multi-pose lipreading, and audiovisual speech recognition" EURASIP Journal on Advances in Signal Processing. doi:10.1186/1687-6180-2012-51. 2012.
[20] J. Huang, B. Kingsbury "Audio-visual deep learning for noise robust speech recognition" in Proc. IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP) 7596–7599. doi:10.1109/icassp.2013.6639140. 2013.
[21] Bear H. L., Harvey R. W., Theobald B.-J, Lan Y "Which phoneme-to-viseme maps best improve visual-only computer lip-reading".Lecture Notes in Computer Science, 230-239. doi:10.1007/978-3-319-14364-4_22. 2014.
[22] Darryl. Stewart, R. Seymour, A. Pass, J. Ming, "Robust audio-visual speech recognition under noisy audio-video conditions" IEEE Transactions on Cybernetics 44 (2) 175–184 doi:10.1109/TCYB.2013.2250954.2014.
[23] Kuniaki. Noda, Yuki. Yamaguchi, Kazuhiro. Nakadai, H. G. Okuno, Tetsuya. Ogata, "Audio-visual speech recognition using deep learning. Applied Intelligence". doi:10.1007/s10489-014-0629-7.2015.
[24] Chao. Sui, M. Bennamoun, R. Togneri, "listening with your eyes: Towards a practical visual speech recognition system using deep Boltzmann machines" IEEE International Conference on Computer Vision (ICCV) – Santiago 154–162. doi:10.1109/ICCV.2015.26. 2015.
[25] H. Ninomiya, N. Kitaoka, S. Tamura, Y. Iribe, K. Takeda, "Integration of deep bottleneck features for audio-visual speech recognition" in Proceedings of Inter speech, 2015.
[26] D. Lee, J. Lee, K.-E. Kim "Multi-view automatic lip-reading using neural network" in Proc. Asian Conference on Computer Vision, https://doi.org/10.1007/978-3-319-54427-4_22. 2017.
[27] Rekik, Ahmed; Ben-Hamadou, Achraf; Mahdi, Walid "An adaptive approach for lip-reading using image and depth data. Multimedia Tools and Applications". doi:10.1007/s11042-015-2774-3. 2016.
[28] T. Saitoh, Z. Zhou, G. Zhao, M. Pietik¨ainen, "Concatenated frame image-based CNN for visual speech recognition" in Proc. Asian Conference on Computer Vision, DOI: 10.1007/978-3-319-54427-

4 21. 2016.

[29] Chao. Sui, R. Togneri, M. Bennamoun, “A cascade gray-stereo visual feature extraction method for visual and audio-visual speech recognition” Speech Communication doi:10.1016/j.specom.2017.01.005.2017.

[30] M. Wand, J. Schmidhuber, “Improving speaker-independent lipreading with domain-adversarial training, in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2017, pp. 3662–3666. https://doi.org/10.48550/arXiv.1708.01565

[31] S. Petridis, J. Shen, D. Cetin, M. Pantic, “Visual-only recognition of normal, whispered and silent speech”. in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (in press), doi:10.1109/ICASSP.2018.8461596. 2018.

[32] Depeursinge, Adrien “Fundamentals of Texture Processing for Biomedical Image Analysis.” doi:10.1016/B978-0-12-812133-7.00001-6 .2017.

[33] Zhou, Ziheng; Zhao, Guoying; Pietikainen, “Towards a practical lipreading system” IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2011.5995345.2011.

[34] C. Mithlesh, D Shukla, M. Sharma. “ Video to Image Conversion Techniques Video Frame Extraction.” International Journal of Emerging Science and Engineering (IJESE) ISSN: 2319–6378, Volume-4 Issue-3, February 2016.

[35] Jin Ting, Chai Song, Hongyang Huang, Taoling Tian. “A Comprehensive Dataset for Machine-Learning-based Lip-Reading Algorithm” The 8th International Conference on Information Technology and Quantitative Management. Procedia Computer Science 199 (2022) 1444–1449. ww.sciencedirect.com. 2022.

[36] Huang, Tingwen; Zeng, Zhigang; Li, Chuandong; Leung, Chi Sing “.Integration of Face Detection and User Identification with Visual Speech Recognition.” doi:10.1007/978-3-642-34500-5_57.2012.

[37] Sharma, S.; Shanmugasundaram, Karthikeyan; Ramasamy, Sathees Kumar. “ FAREC — CNN based efficient face recognition technique using Dlib”. International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)192–195. doi:10.1109/icaccct.2016.7831628. 2016.

[38] N. H. Alsulami, A. T. Jamal and L. A. Elrefaei, " Deep learning-based approach for Arabic visual speech recognition," Computers, Materials & Continua, vol. 71, no.1, pp. 85–108, 2022. https://doi.org/10.32604/cmc.2022.019450.2022.

[39] Hlaing Htake Khaung Tin. “Age Dependent Face Recognition using Eigenface.” I.J. Modern Education and Computer Science, 2013, 9, 38-44. DOI: 10.5815/ijmecs.2013.09.06.2013.

[40] Chen, Chu-Song; Lu, Jiwen; Ma, Kai-Kuang “ Concatenated Frame Image Based CNN for Visual Speech Recognition.”.doi:10.1007/978-3-319-54427-4_21. 2017.

[41] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. “ ImageNet Classification with Deep Convolutional Neural Networks.” Advances in Neural Information Processing Systems 25 (NIPS 2012) ISBN: 9781627480031.2012.

[42] M. Arif Wani, Farooq Ahmad Bhat, Saduf Afzal, and Asif Iqbal Khan “Advances in Deep Learning” Springer, 2020.

[43] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S. Lew “Deep learning for visual understanding: A review” Elsevier, Neurocomputing, Volume 187, Pages 27-48, 26 April 2016.

[44] Nada A. and Heyam H. Al-Baity “A multimodal biometric system for personal verification based on the different level fusion of iris and face traits” Biosci. Biotech. Res. Comm. 12(3): 767-778, 2019.

[45] Karen S. and Andrew Zisserman “VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION” was Published as a conference paper at ICLR, in 2015.