

A Mini Project Report on

**“Classification of Kannada Character using Deep Learning”**

Submitted in partial fulfillment of the requirement for the degree of

Master of Technology In

Computer Science and Engineering

Submitted by:

#### Pratibha M Goudar [01FE20MCS010]

Under the guidance of

#### Dr. Vishwanath P Baligar

SCHOOL OF COMPUTER SCIENCE & ENGINEERING KLE TECHNOLOGICAL UNIVERSITY, HUBBALLI – 580031

2021-22

2021 - 2022



SCHOOL OF COMPUTER SCIENCE &ENGINEERING



CERTIFICATE

This is to certify that a Mini Project titled **“Classification of Kannada character using deep learning”** is a bonafide work carried out by the student comprising of **PRATIBHA M GOUDAR 01FE20MCS010** for partial fulfillment of completion of third-semester M.Tech in Computer Science and Engineering during the academic year 2021-22.

Guide P. G. Coordinator Head, SoCSE

Dr. Vishwanath. P. Baligar Dr. Vishwanath. P. Baligar Dr. Meena S. M

External Viva -Voce:

Name of the Examiners Signature with date 1.

2.

**ACKNOWLEDGEMENT**

The successful completion of any task is incomplete without complementing those who made it possible and whose guidance and encouragement made my effort successful.

I am indebted to my beloved Guide Dr. Vishwanath. P. Baligar who was constant source of enthusiasm and whose sincere guidance, valuable suggestions and benevolent direction where mainly responsible for my successful completion of the project.

I am indebted to our beloved P. G. Coordinator Dr. Vishwanath. P. Baligar who is constant source of enthusiasm and whose sincere guidance, valuable suggestions and benevolent direction where mainly responsible for my successful completion of the project.

I also take this opportunity to express my deep gratitude and sincere thanks to the HOD of School of Computer Science & Engineering Dr. Meena S. M. for her tremendous source of inspiration and valuable help in challenging my effort in right direction.

I take this opportunity to thank our Registrar Dr. N. H. Ayachit and our Vice Chancellor Dr. Ashok.S.Shettar, for providing healthy environment in the college, which helped in concentrating on the task.

Last but not the least I will take the pleasure to thank all the teaching and nonteaching staff of our college and also my student friends. I also thank K.L.E Technological University for providing computational support.

PRATIBHA M GOUDAR: [01FE20MCS010]

## ABSTRACT

The technology of recognition has advanced in recent times. There are so many models that performed brilliantly when they used to detect handwritten alphanumeric characters. But when it comes to Kannada handwritten classification these predefined models slightly underperformed. The complexity of Kannada and unavailability of a good dataset are the main reasons for the underperformance of these models. So, the problem still quite unsolved, for that reason we have created our own dataset. But in Kannada character recognition some of the models performed very well. We propose a Deep Convolutional Neural Network model to recognize for the Kannada character recognition. The dataset used in this experiment is pretty good and have trained the model and test the models and get accuracy of models.

**Keywords:** Deep Learning, Kannada, dataset

# CONTENTS

|  |  |  |
| --- | --- | --- |
|  | ACKNOWLEDGEMENT | i |
|  | ABSTRACT | ii |
|  | CONTENTS | iii |
| 1 | INTRODUCTION | 8 |
| 1.1 | Problem Statement | 9 |
| 1.2 | Objectives | 9 |
| 1.3 | Motivation | 9 |
| 1.4 | Application of deep learning | 9 |
| 2 | LITERATURESURVEY | 10 |
| 3 | SOFTWAREREQUIREMENTSPECIFICATION | 12 |
| 3.1 | Functional Requirements | 12 |
| 3.2 | Non-Functional Requirements | 12 |
| 3.3 | Hardware Requirements | 12 |
| 3.4 | Software Requirements | 12 |
| 4 | METHODOLOGY | 13 |
| 4.1 | Deep Learning Models | 13 |
| 4.2 | Evaluation metrics | 16 |
| 5 | IMPLEMENTATION | 17 |
| 5.1 | Dataset Description | 18 |
| 6 | RESULTS AND ANALYSIS | 20 |
| 7 | CONCLUSION | 22 |
| 8 | REFERENCES | 26 |

## LIST OF FIGURE

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.No** | **Figure** | **Figure Name** | **Page No** |
| **1** | **1.1** | **Workflow of system** | **8** |
| **2** | **4.1** | **Convolutional layer architecture** | **13** |
| **3** | **4.2** | **Sequential Model** | **15** |
| **4** | **5.1** | **Applying CNN to dataset** | **17** |
| **5** | **5.2** | **Sample Dataset images** | **18** |
| **6** | **6.2** | **Model accuracy** | **20** |
| **7** | **6.3** | **Model loss** | **21** |
| **8** | **6.4** | **Confusion Matrix with Heat Map** | **21** |

**Chapter1**

**INTRODUCTION**

Neural networks are a series of algorithms that mimic the operations of a human brain to recognize the relationship between vast amounts of data. Kannada is the official language of the south Indian state of Karnataka, The focus of this work is to recognize the Kannada character image. It consists of many stages where the first stage is pre-processing of data, which includes removing unwanted spaces and noise from the images, and resizing. The data set is stored as training and testing data and are fed as input to the neural network as input. Convolutional Neural Network (CNN) model is trained using training data and validation is performed with testing data. Accuracy is improved using trained models as input and the output obtained is used for prediction.

Kannada, is the official language of the south Indian state of Karnataka, The focus of this work is to recognize the Kannada character image .It consists of many stages where the first stage is pre-processing of data, which includes removing unwanted spaces and noise from the images, and resizing.

The data set is stored as training and testing data and are fed as input to the neural network as input. Convolutional Neural Network (CNN) model is trained using training data and validation is performed with testing data. Accuracy is improved using trained models as input and the output obtained is used for prediction.

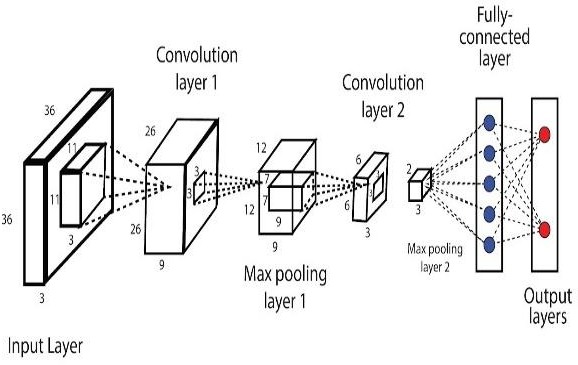


Figure 1.1: CNN model

**Problem Statement**

Designing and developing a model for the classification of Kannada chartered images, and the prediction of those characters based on deep neural network architectures and techniques.

**1.2 Objectives**

* Create our own Kannada Dataset required to implement the model.
* Augment the dataset required to implement the model.
* Select the Appropriate model which can recognize the characters in the given image correctly.[Sequential Model ]
  1. **Motivation**

Artificial Neural networks can be trained over several billion images and can be used to detect and recognize handwritten characters with relative ease and flexibility in an instant. This concept is used in the implementation of this real time handwritten recognition

### 1.4 Application of Deep learning

* + - Some of the applications deep learning are
    - Adding Sounds to Silent Movies.
    - Automatic Machine Translation.
    - Object Classification in Photographs.
    - Automatic Handwriting Generation.
    - Character Text Generation.
    - Image Caption Generation.
    - Automatic Game Playing.
    - Decoding Facial Recognition.

**Chapter 2**

**LITERATURE SURVEY**

Neural networks are a series of algorithms that mimic the operations of a human brain to recognize the relationship between vast amounts of data. Kannada is the official language of the south Indian state of Karnataka, The focus of this work is to recognize the Kannada character image. It consists of many stages where the first stage is pre-processing of data, which includes removing unwanted spaces and noise from the images, and resizing. The data set is stored as training and testing data and are fed as input to the neural network as input. Convolutional Neural Network (CNN) model is trained using training data and validation is performed with testing data. Accuracy is improved using trained models as input and the output obtained is used for prediction.

Researcher Sandhya.N, Krishnan.R carried out work on Broken Kannada Character Recognition a Neural Network based approach [1], they propose a novel approach of rebuilding the broken characters and then using neural network. The broken characters are therefore rebuilt using end point algorithm to remove brokenness and a single layer neural network is used for classification. A recognition accuracy of 98.9% was achieved for broken characters on synthetically generated data sets. The future work is to enhance the rebuilding technique and use the method for non-synthetic datasets and present a comparative study with other promising techniques.

Again C. Naveen, V.N. Manjunath worked on Handwritten Character Segmentation for Kannada Scripts [2] in this paper, a novel Kannada character segmentation algorithm is presented. The proposed method is based on the thinning, branch points and mixture models. The thinning is the most commonly adopted technique to skeletonize an input image and is used to find the branch points present in an image. The proposed method is tested on handwritten Kannada words and has shown encouraging results.

Further Chaithra D, Dr. K Indira, Handwritten Online Character Recognition for Single Stroke Kannada Characters [3] in this paper OHKC character recognition system is discussed. The samples are collected from variety of writers belonging to various age groups, in order to achieve a variety of writing styles of characters. The character recognition system for online handwritten Kannada characters is designed, implemented, and tested. The performance of the system with KNN. Along with the normalization based features has a recognition rate.

Roshan Fernandes, Anisha P carried out work on Kannada Handwritten Script Recognition using Machine Learning Techniques [4] proposed two methods to recognize the handwritten Kannada characters. The first techniques are by Tesseract tool, and second are by using Convolution Neural Network (CNN). With Tesseract tool we have achieved 86% accuracy and through Convolution Neural Network we achieved 87% accuracy although it might improve with the data set chosen and further enhanced image processing.

Leena Ragha, M Sasikumar Adapting Moments for Handwritten Kannada Kagunita Recognition [5] In this paper, investigate the use of moments features on Kannada Kagunita. To recognize a Kagunita, we need to identify the vowel and the consonant present in the image. The features are trained and tested for both vowel and Kagunita recognition on Multi-Layer Perceptron with Back Propagation Neural Network. The recognition results for vowels is average 85% and consonants is 59% when tested on separate test data with moments features from directional images and cut images.

Niranjan S.K, Vijaya Kumar carried out work on FLD based Unconstrained Handwritten Kannada Character Recognition [6]**.** In this paper, propose unconstrained handwritten Kannada character recognition based on Fisher Linear Discriminant Analysis (FLD). The proposed system extracts features from well known FLD, Two-dimensional FLD (2D-FLD), and Diagonal FLD. For classification purpose, explored different distance measure techniques and tested there superiority on unconstrained handwritten Kannada characters.

Shwetha D. Mrs. Ramya S. Comparison of Smoothing Techniques and Recognition Methods for Online Kannada Character Recognition System [7].This paper aimed at working on Online Recognition of Handwritten Kannada Characters. The recognition was done for the Top, Middle and Bottom strokes of Kannada characters. Recognition was carried out by KNN and SVM pattern recognition methods and a second level of verification rules was incorporated, yielding a maximum recognition rate of 92.5% for KNN and 94.35% for SVM.

Further Aravinda C.V , Dr.H.N.Prakash carried out work on Template Matching Method For Kannada Handwritten Recognition Based On Correlation Analysis[8] In this paper we adopted a Correlation Technique for recognition of Kannada Handwritten Characters. The formation of Kannada Characters into its compound form, also called as Kagunita makes its recognition more complex. The segmented individual character is correlated with the stored templates. The template with maximum correlation value is displayed in editable format.

Mr. Sridharamurthy S K, Dr. H.R.Sudarshana ReddyPCA based feature vector for handwritten Kannada characters recognition [9]. An approach for selection of features using principal component analysis technique to classify segmented (isolated) Kannada characters is presented in this paper. Artificial neural network is used as classifier. Handwritten characters are scan converted to binary images and normalized to a size of 50 x 50 pixels. the implementation of this approach on a comprehensive database, higher degree of accuracy in results has been obtained.

Madhavaraj A, A G Ramakrishnan.Improved recognition of aged Kannada documents by effective segmentation of merged characters [10] In this paper, propose the first algorithm to segment merged Kannada characters by using a hypothesis to select the positions to be cut. The proposed segmentation algorithm works well for different font styles, shapes, and sizes better than the existing vertical projection profile based segmentation. A method to optimally select the sequence of segments from all possible sequences has been proposed and implemented. The results show that the segmentation algorithm has achieved a segmentation accuracy of 89.6%.

## Chapter 3

**SOFTWARE REQUIREMENT SPECIFICATION**

Software Requirement Specification talks about the necessary hardware and software components need for the smooth working of the system, some of the software and hardware requirements are listed.

### 3.1 Functional Requirements

* The deep learning model should predict more accuracy

### 3.2 Non-Functional Requirements

* Non-Functional requirements for the application are as follows:
* Compatibility: The application should work on any machine, which required configurations.
* Availability: The application should be available all the time.
* Performance: The application must provide high performance.
* Efficiency: The application must have good final test accuracy after the completion of training the model

### 3.3 Hardware Requirements

* RAM 4GB and above
* Windows OS
* Processor:3.2 GHz processor

### Software Requirements

* Programming Language: Python
* Packages:
  + - Numpy
    - Matplotlib
    - pandas
  + Framework:
    - OpenCV
    - Tensor Flow

## Chapter 4

**METHODOLOGY**

**4.1 Deep Learning Models**

In deep learning, a computer model learns to perform classification tasks directly from images, text, or sound. Deep learning models can achieve accuracy, sometimes exceeding human-level performance. In deep learning models, the models are trained by using a large set of labeled data and neural network architectures that contain many layers. The term deep usually refers to the number of hidden layers in the neural network. Traditional neural networks only contain 2-3 hidden layers, while deep networks can have as many as 150. Deep learning models are trained by using large sets of labeled data and neural network architectures that learn features directly from the data without the need for manual feature extraction.

There are many Deep Learning models some of them are VGG-16, ResNet, VGG-19, Inception, Alex Net, Face-net, YoLo-v3, deep-face etc. We have considered sequential (baseline) model for our study.

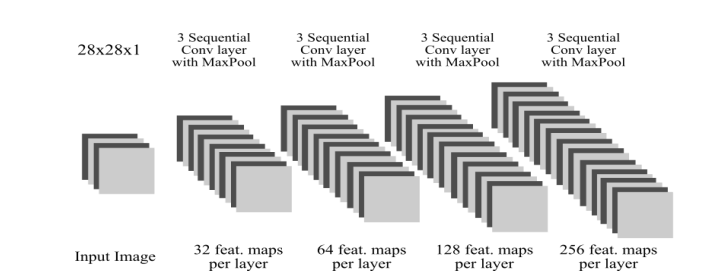


Figure 4.1: Convolutional layer architecture

CNN was initially applied to digit recognition task CNN and its variants are gradually adopted to various applications. CNN is designed to imitate human visual processing, and it has highly optimized structures to process 2D images. Furthermore, CNN can effectively learn the extraction and abstraction of 2D features. In detail, the max-pooling layer of CNN is very effective in absorbing shape variations.

Moreover, sparse connection with tied weights makes CNN involve with fewer parameters than a fully connected network with similar size. Most importantly, CNN is trainable with the gradient-based learning algorithm and suffers less from the diminishing gradient problem. Given that the gradient-based algorithm trains the whole network to minimize an error criterion directly, CNN can produce highly optimized weights and good generalization performance.

The overall architecture of a CNN, as shown in Figure, consists of two main parts: feature extractor and classifier. In the feature extraction unit, each layer of the network receives the output from its immediate previous layer as inputs and passes current output as inputs to the immediate next layer, whereas classification part generates the predicted outputs associated with the input data.

The two basic layers in CNN architecture are convolution and pooling layers. In convolution layer, each node extracts the features from the input images by convolution operation on the input nodes. The max-pooling layer abstracts the feature through average or maximum operation on input nodes.

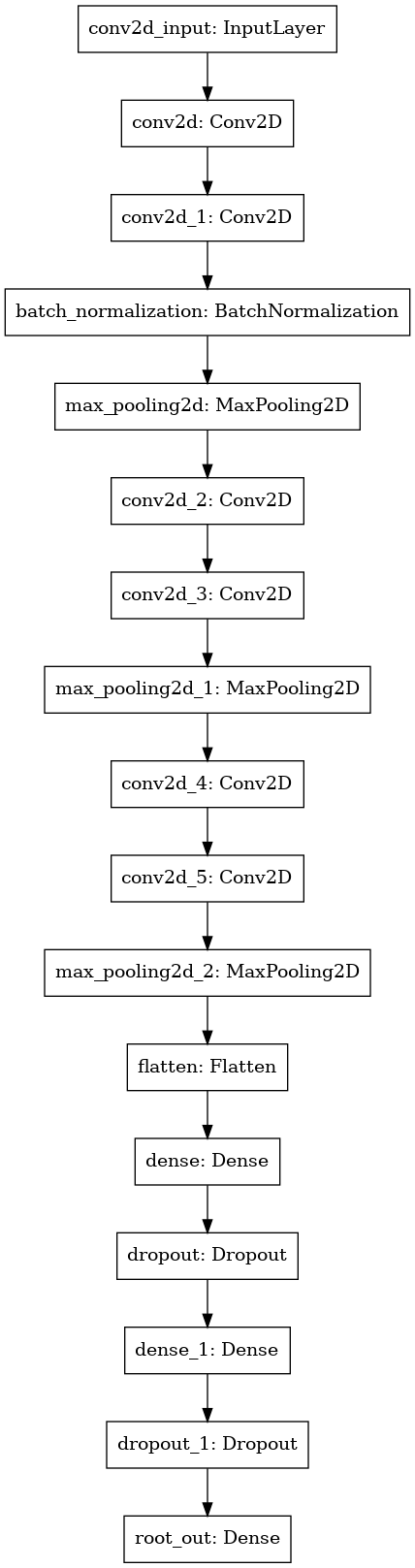
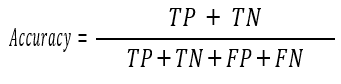


Figure 4.2 : Sequential Model

**4.2 Evaluation metrics:**

The paper includes evaluation metrics that calculates the performance of the model. Categorical cross entropy loss is used to calculate the training loss of the models (CNN and VGG16). To calculate the accuracy of the models the below equation is used:



Where TP= True Positive, TN=True Negative, FP=False Positive and FN=False Negative

**Chapter 5**

**IMPLEMENTATION**

In conventional CNN, convolution is done on an image with a given filter to construct a correlation statistic, layer-by-layer and then clustering some neurons that are highly correlated as an output. Important point to note is the correlation is local to the image patch and the highest correlation exist in the earlier layers of the network and hence large filter size and early pooling would reduce the important information hidden in the image patch.

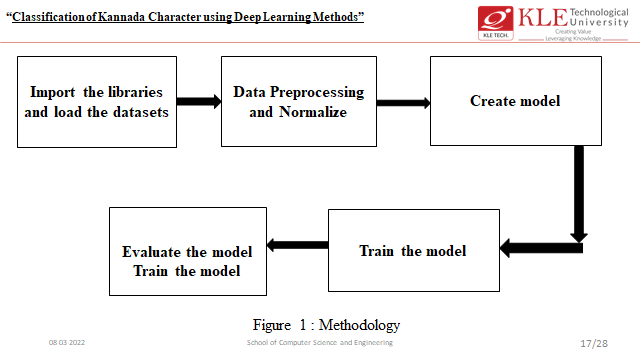


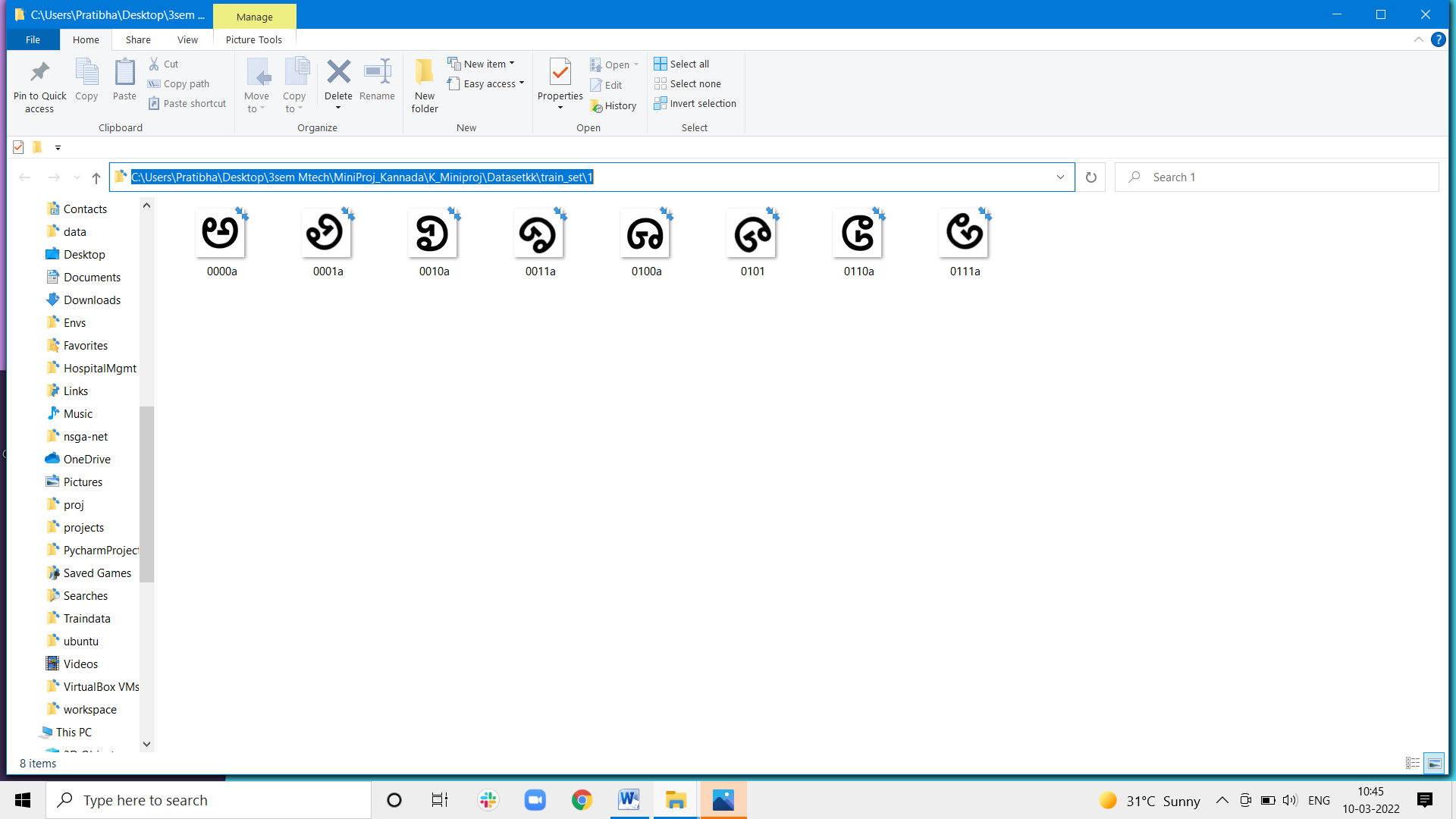
Figure5.1: Applying CNN to dataset

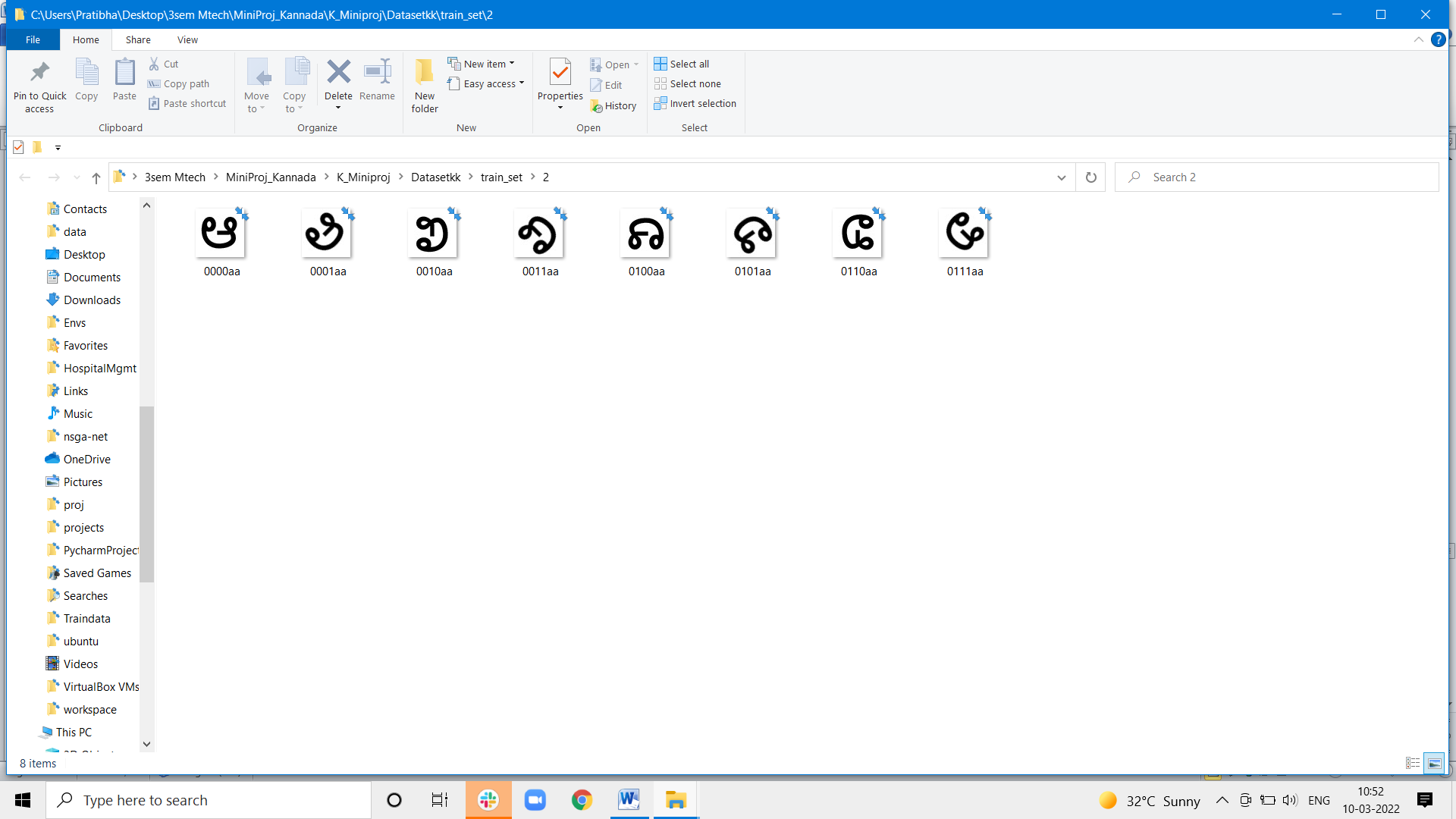
At the first step, importing of necessary libraries is carried out to fit, train and test the model. Then, the CNN model with six convolution layers and six max-pooling layers is applied. It is then flattened and fully connected. Then the model is fitted with supplied image and the training accuracy is obtained.

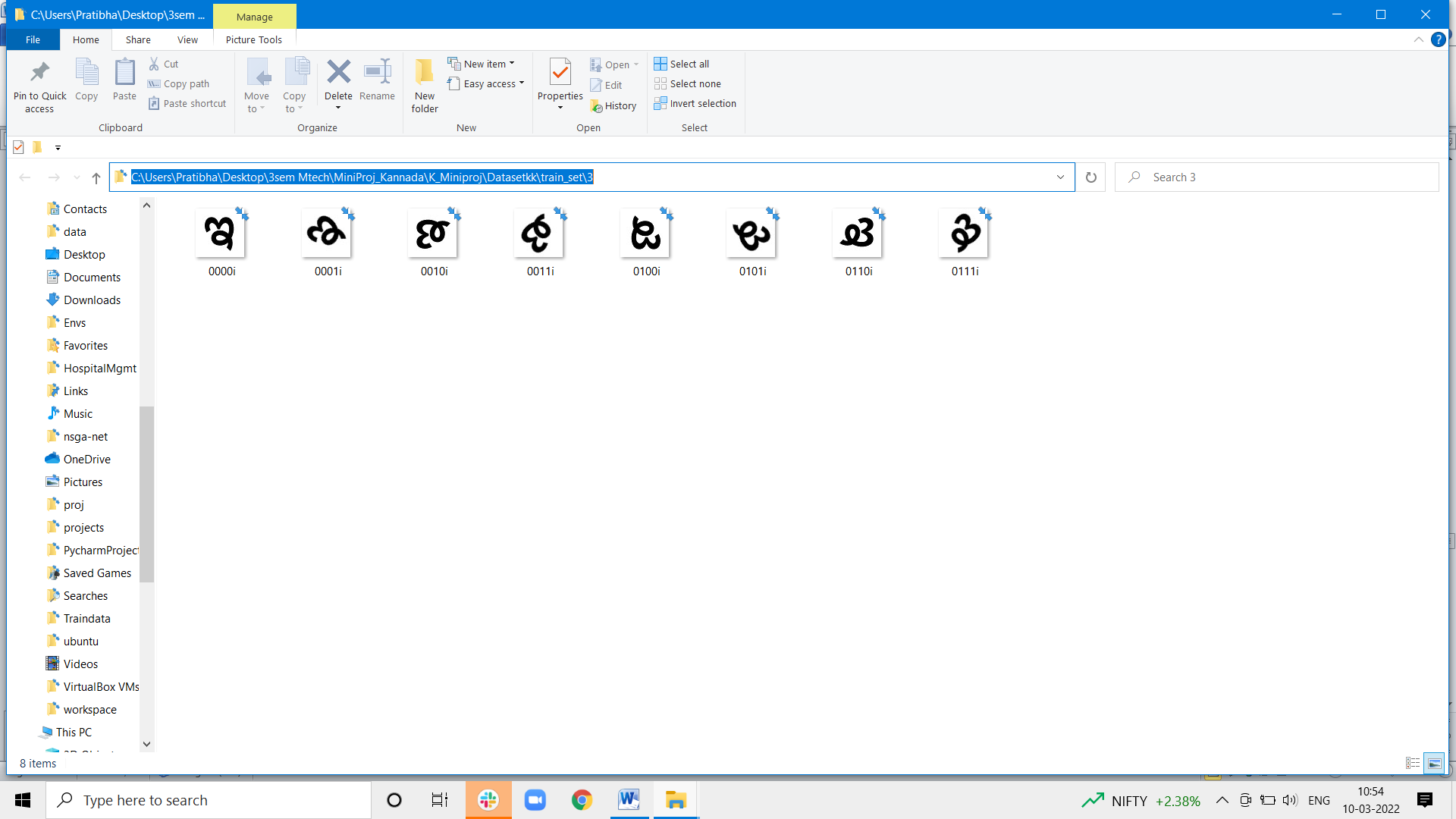
The model is tested using testing data and the testing accuracy is obtained. The classification of brain tumor images is plotted using a heat map confusion matrix. The fundamental advantage of CNN over its predecessors is that it automatically recognizes relevant elements without the need for human intervention.

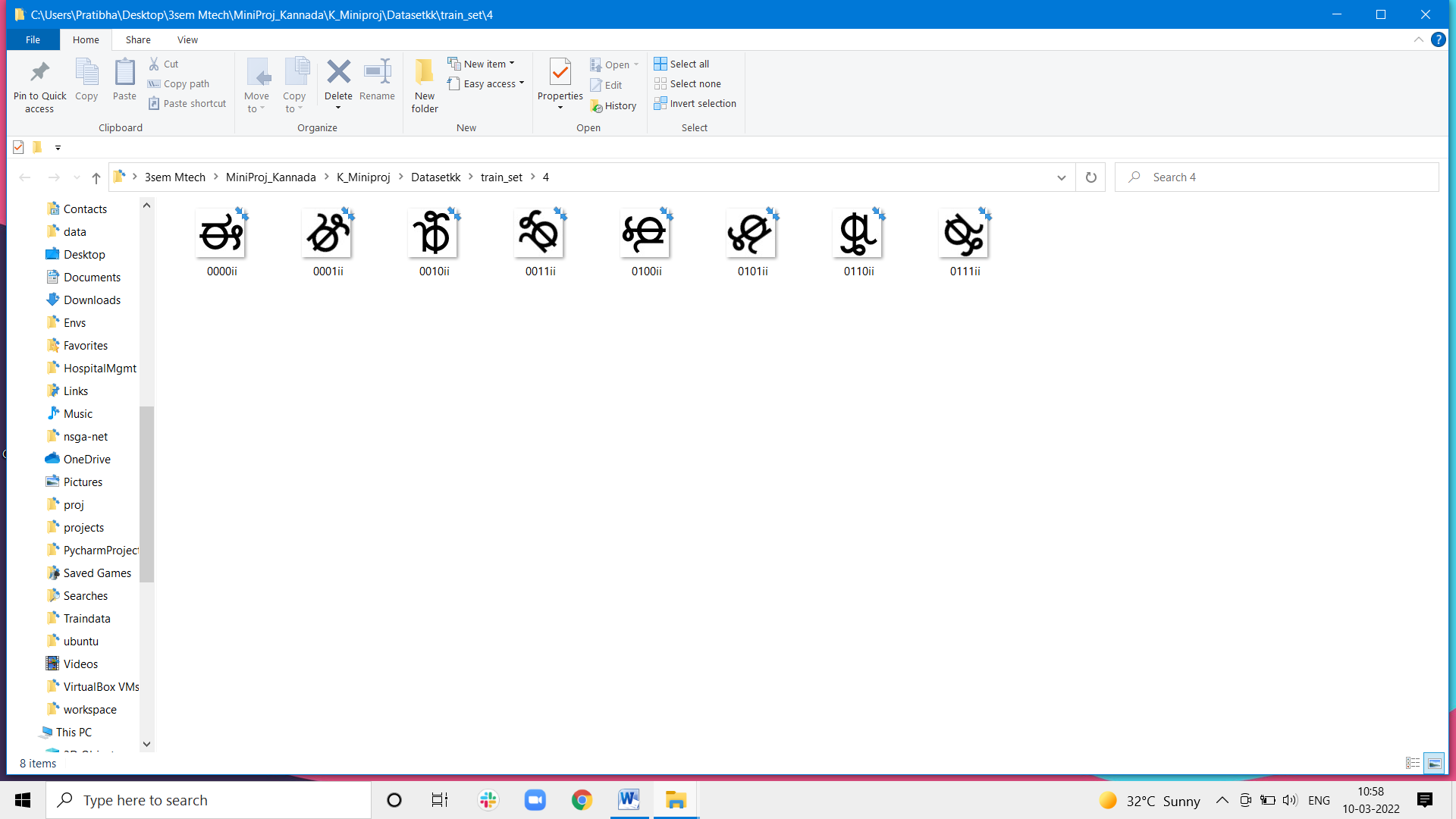
5.1 **Dataset Description**

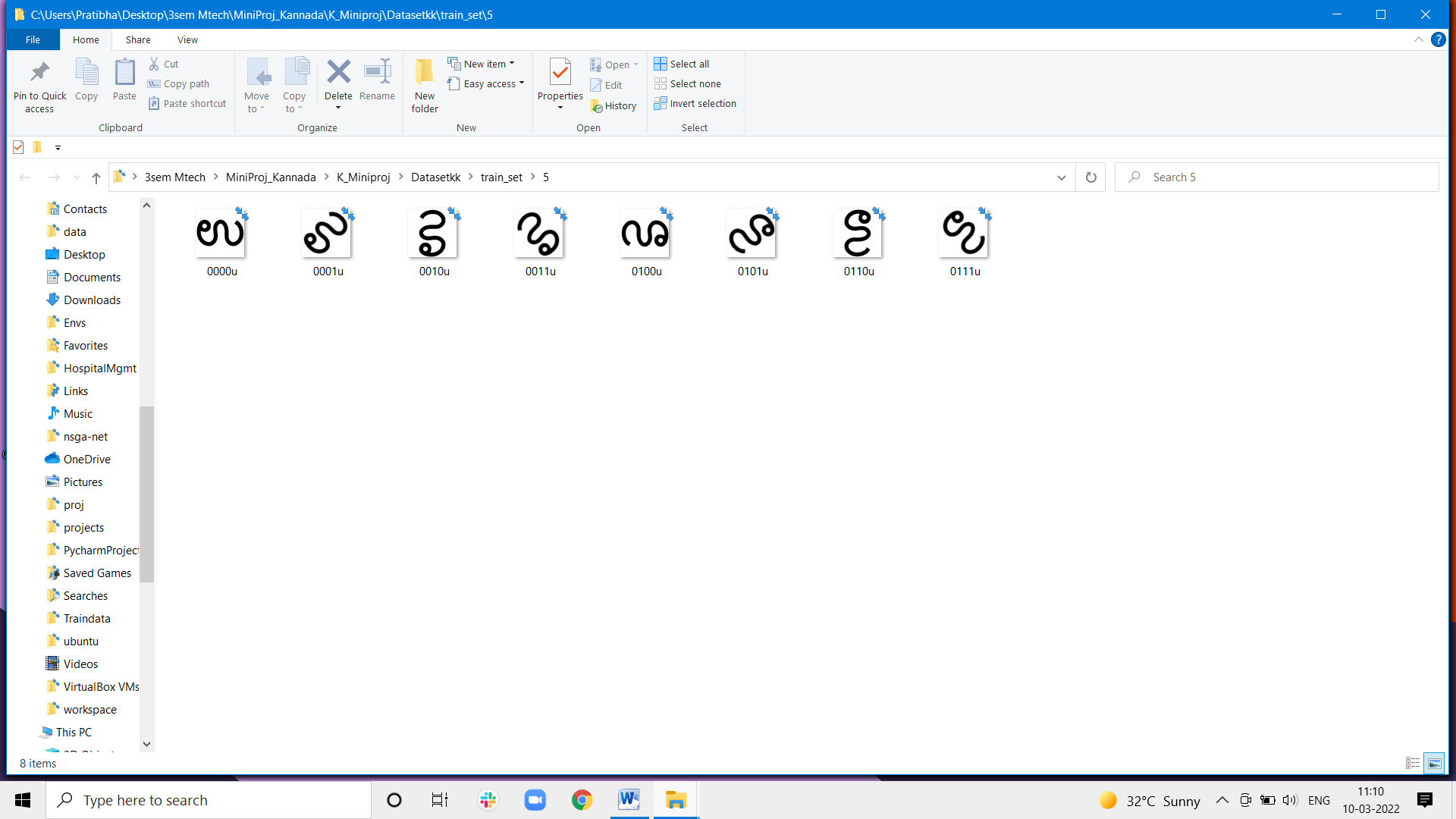
* Installing baraha software.
* Create kannada characters.
* Resize the image 64\*64 and save it as .bmp files
* Upload it to drive.
* Training Dataset contains 4,448 images with 556 class.
* For Testing we consider kaggle Kannada recognition dataset images.

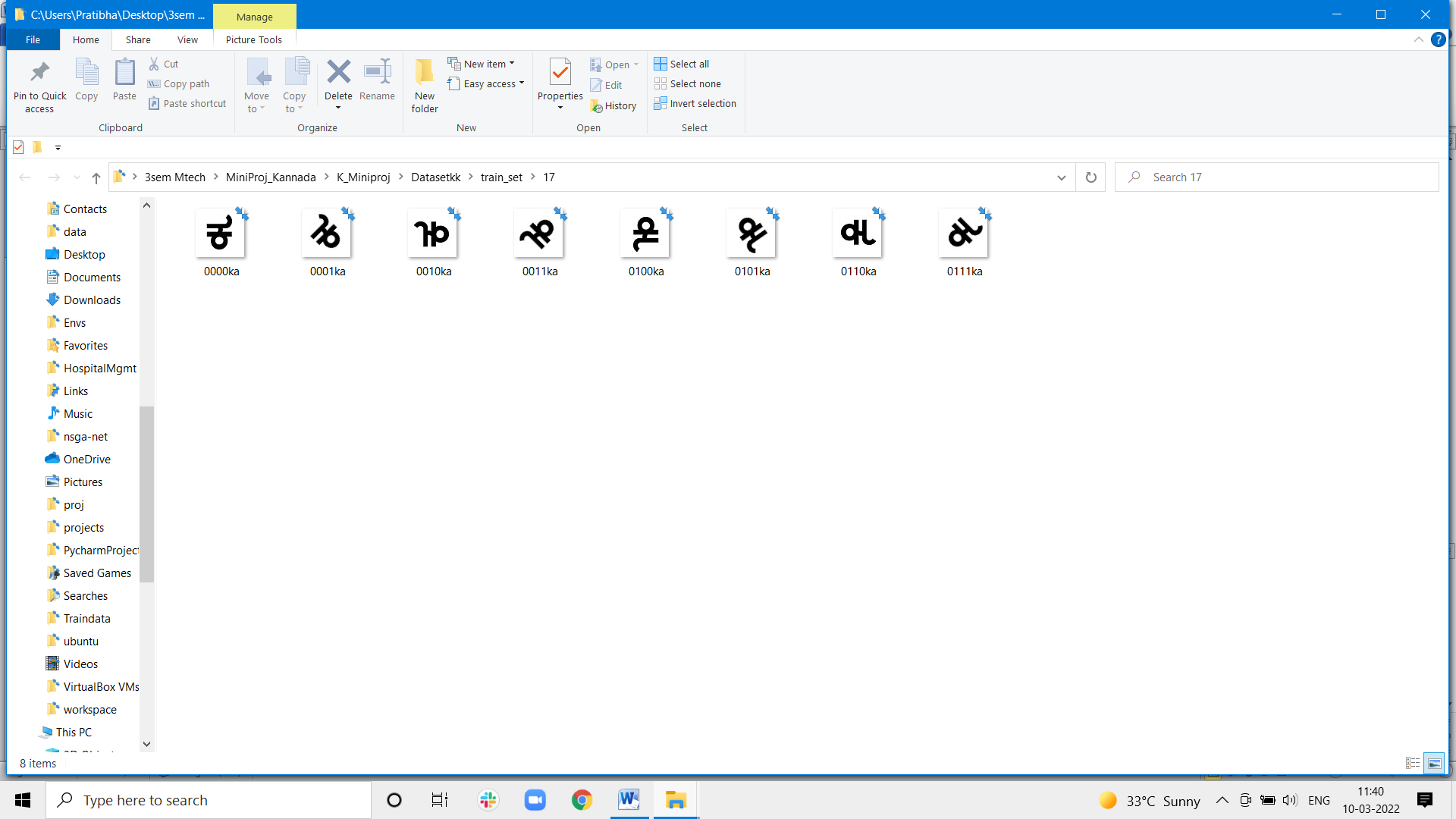


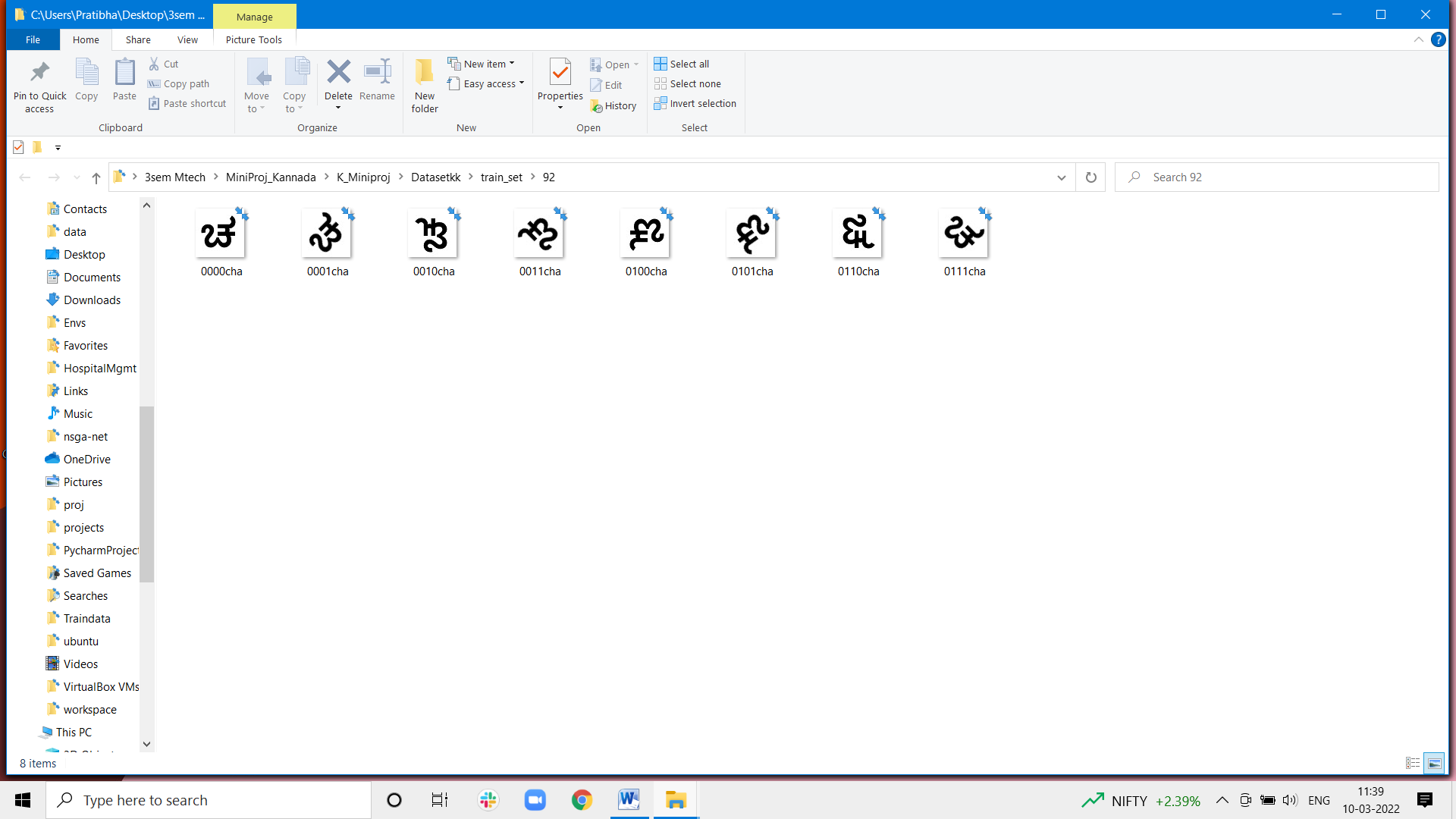


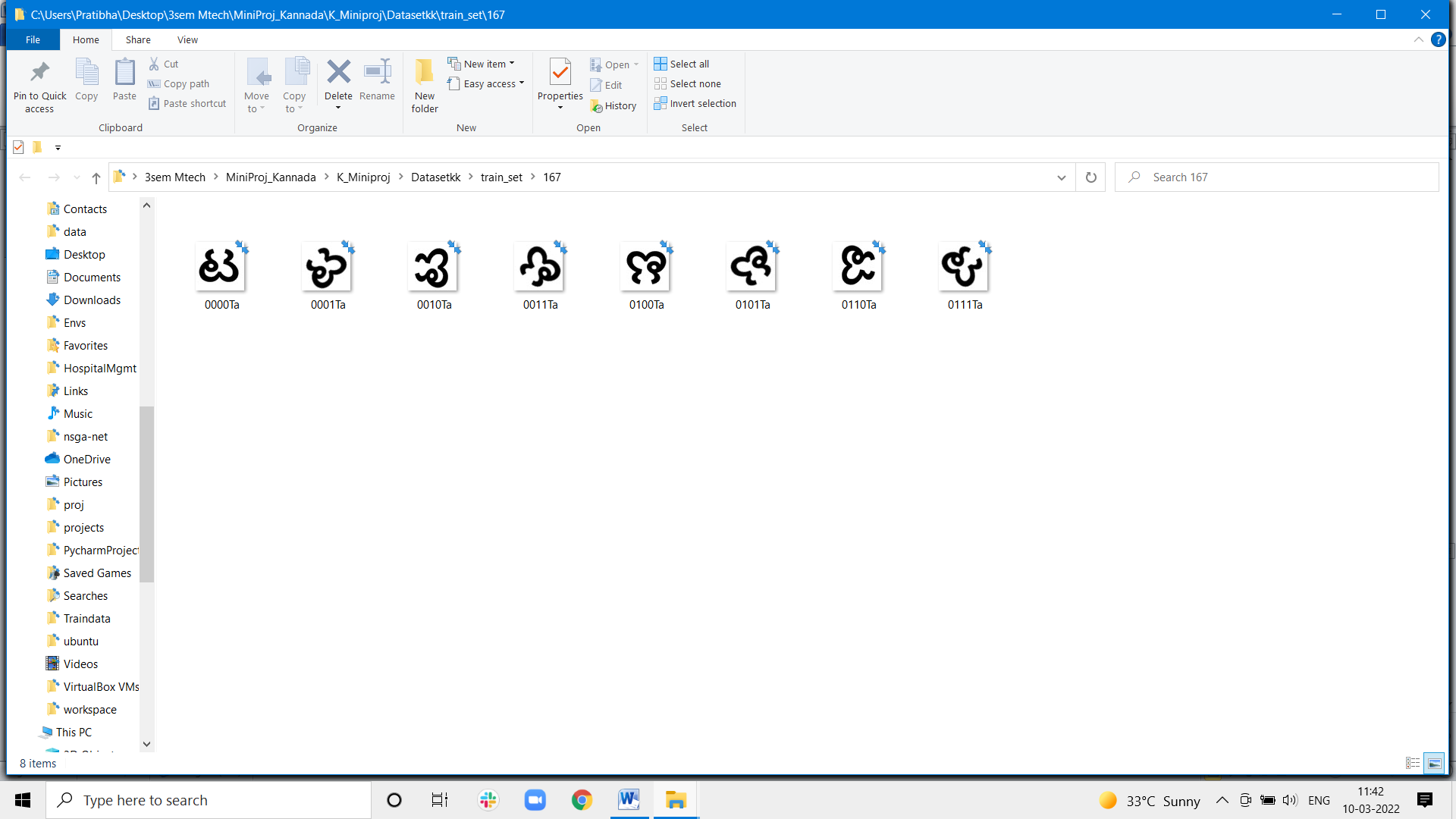


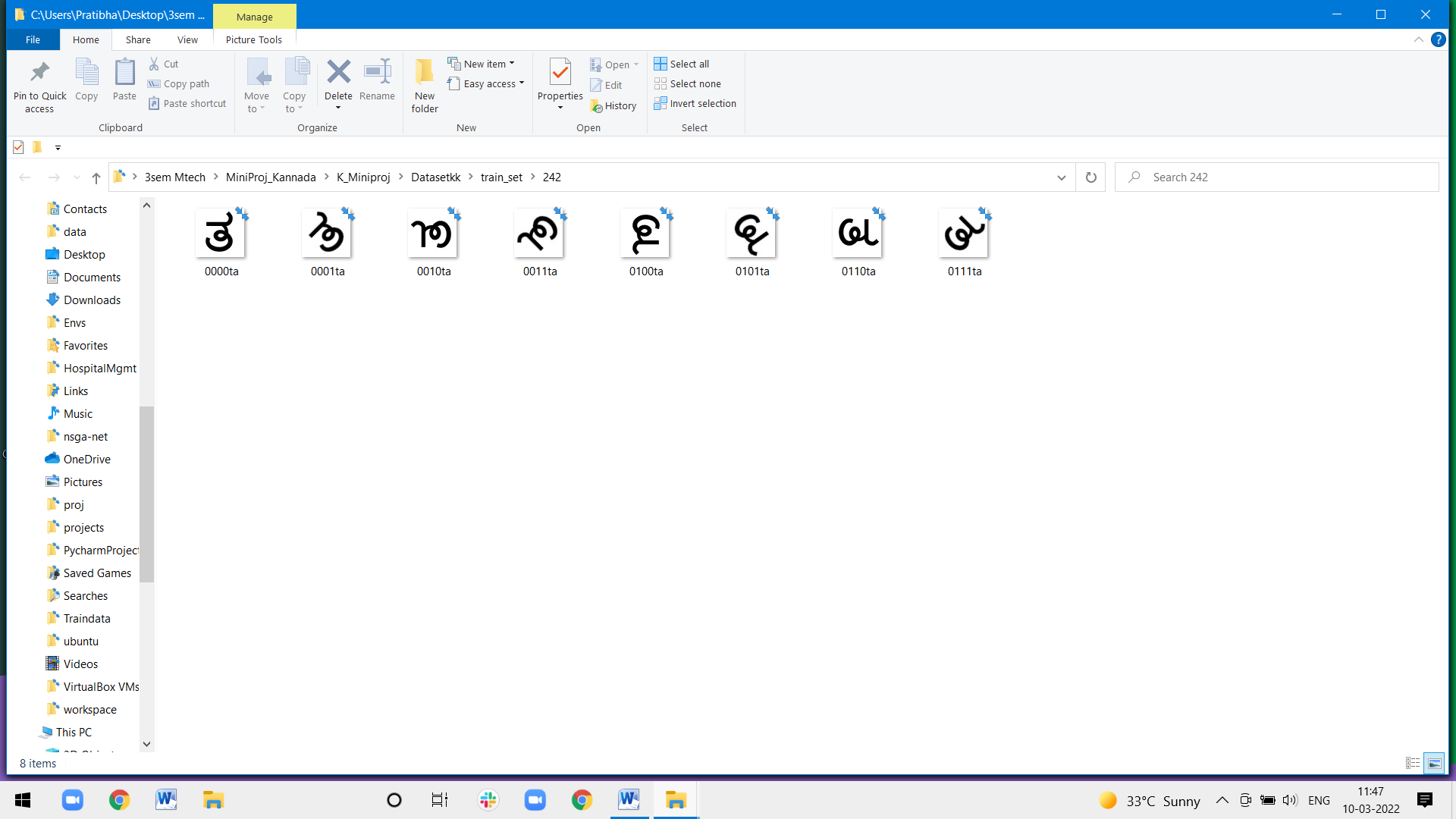


Figure 5.2 : Sample Dataset images

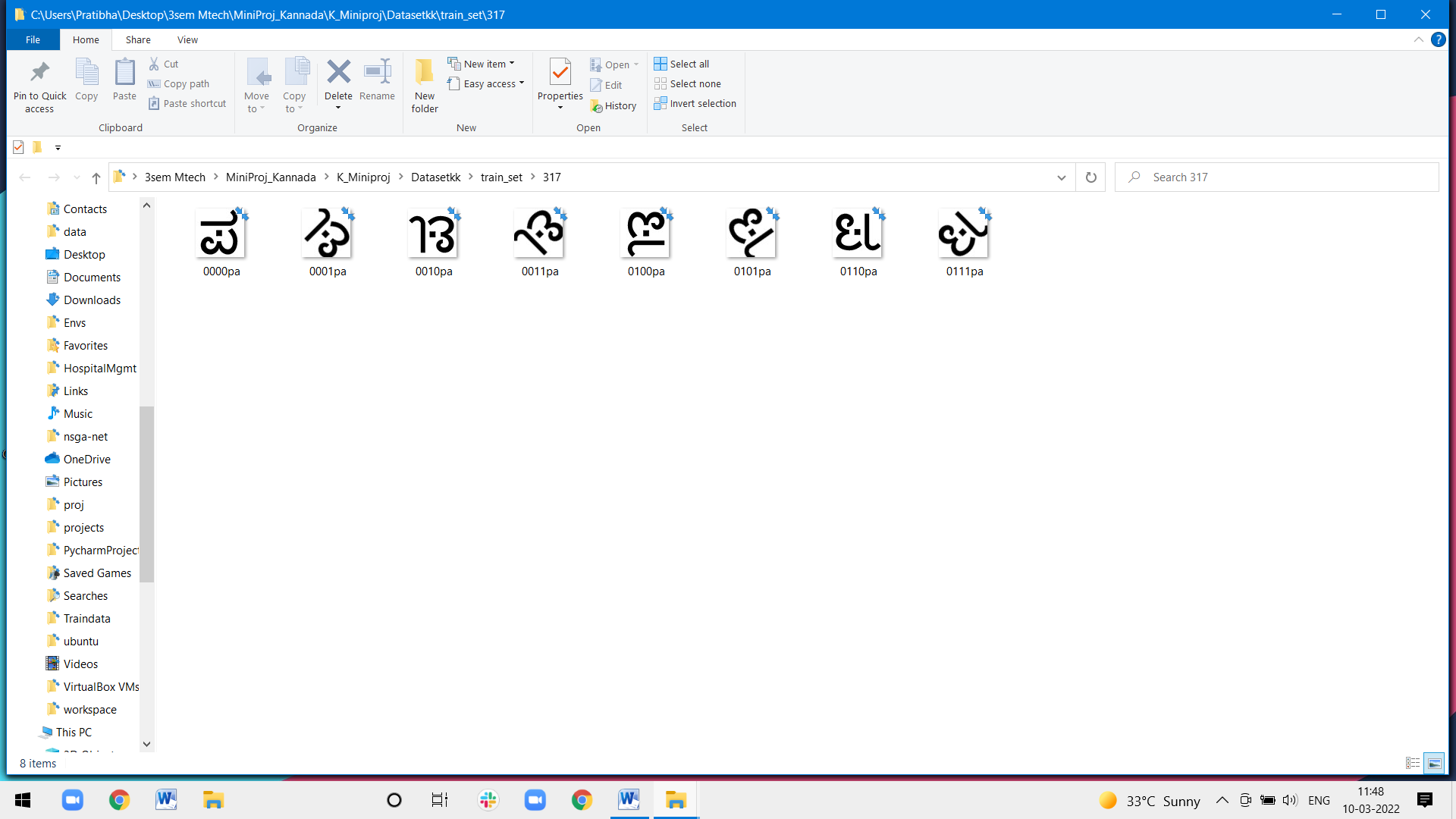


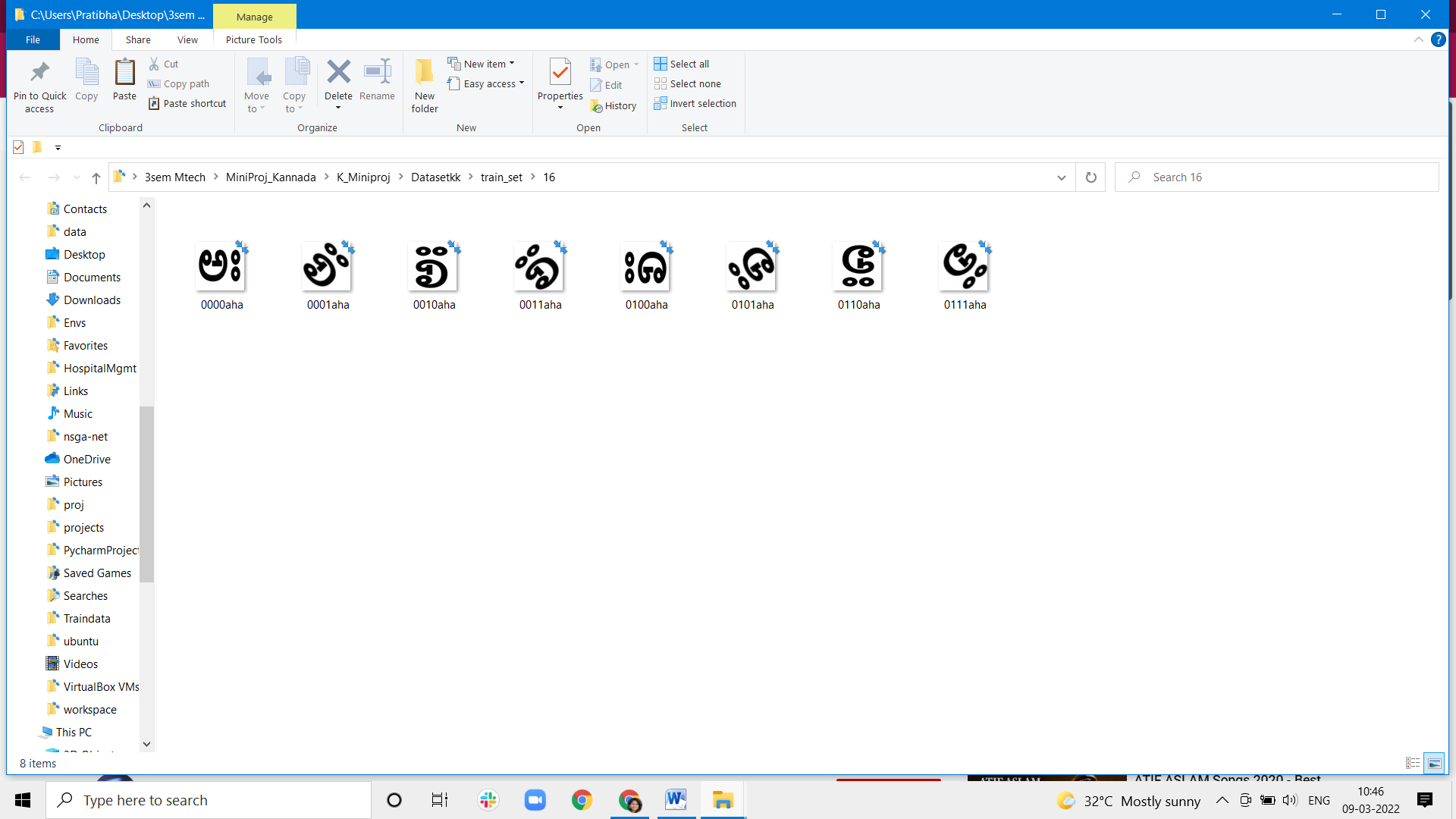


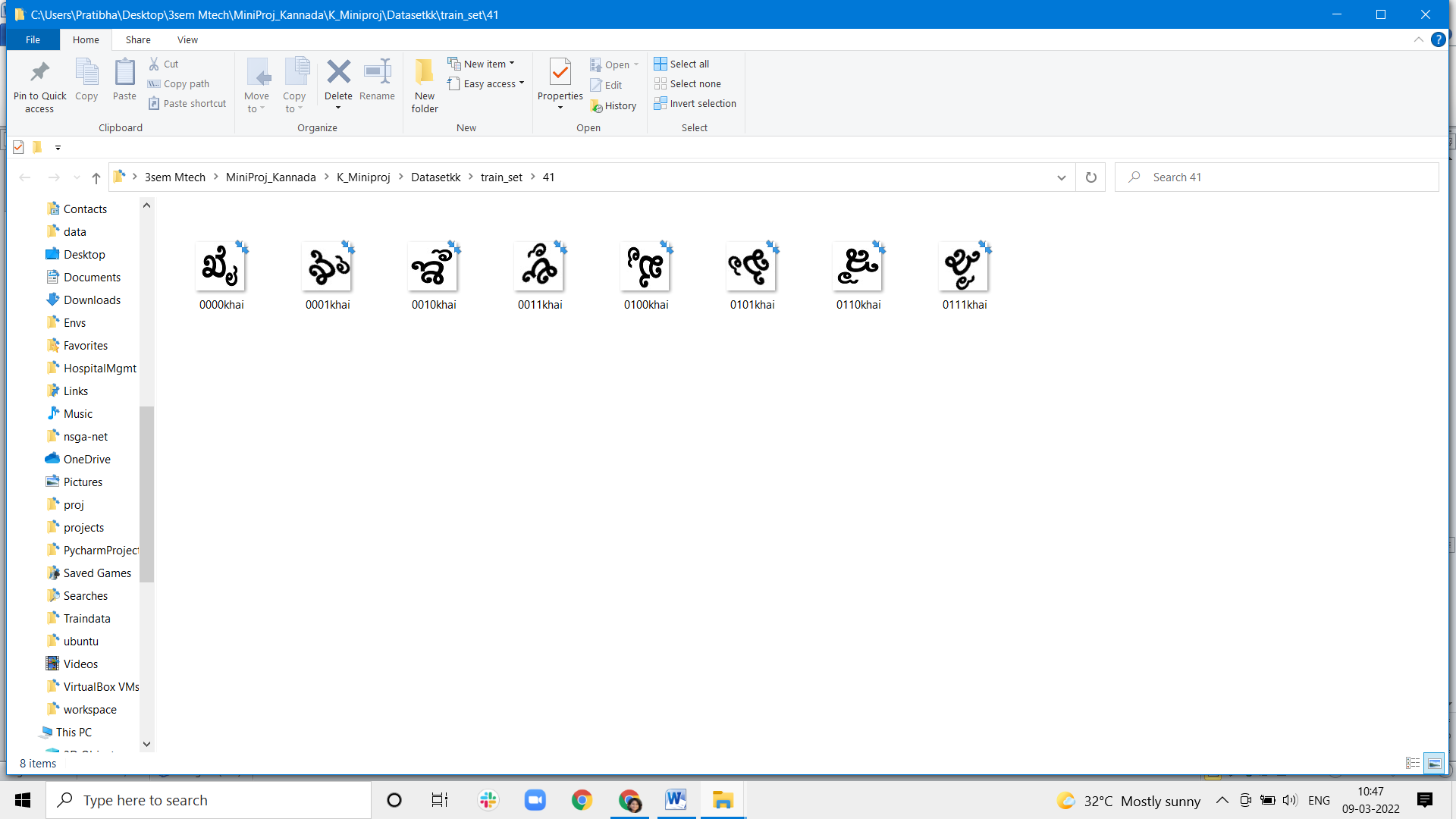


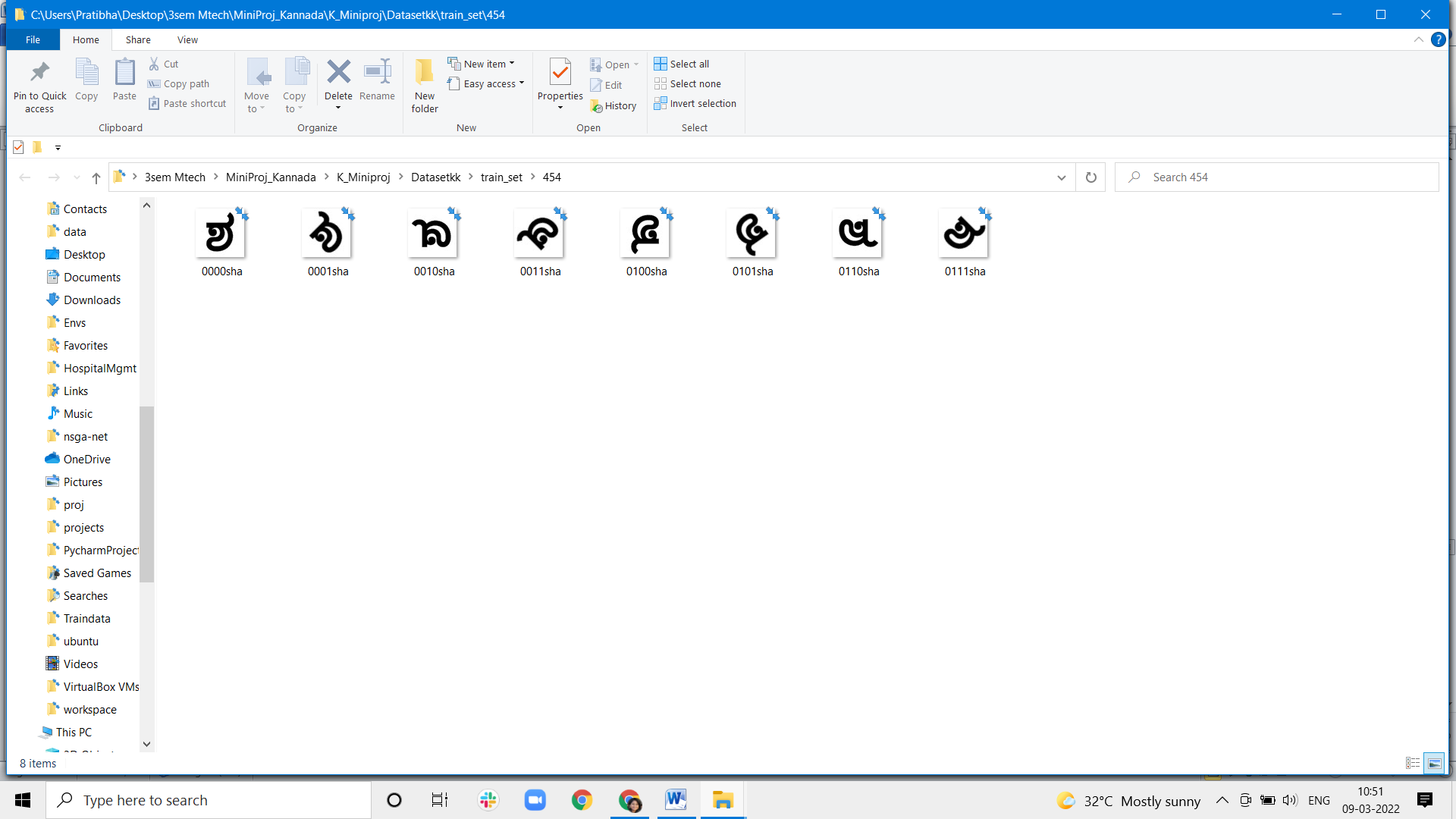


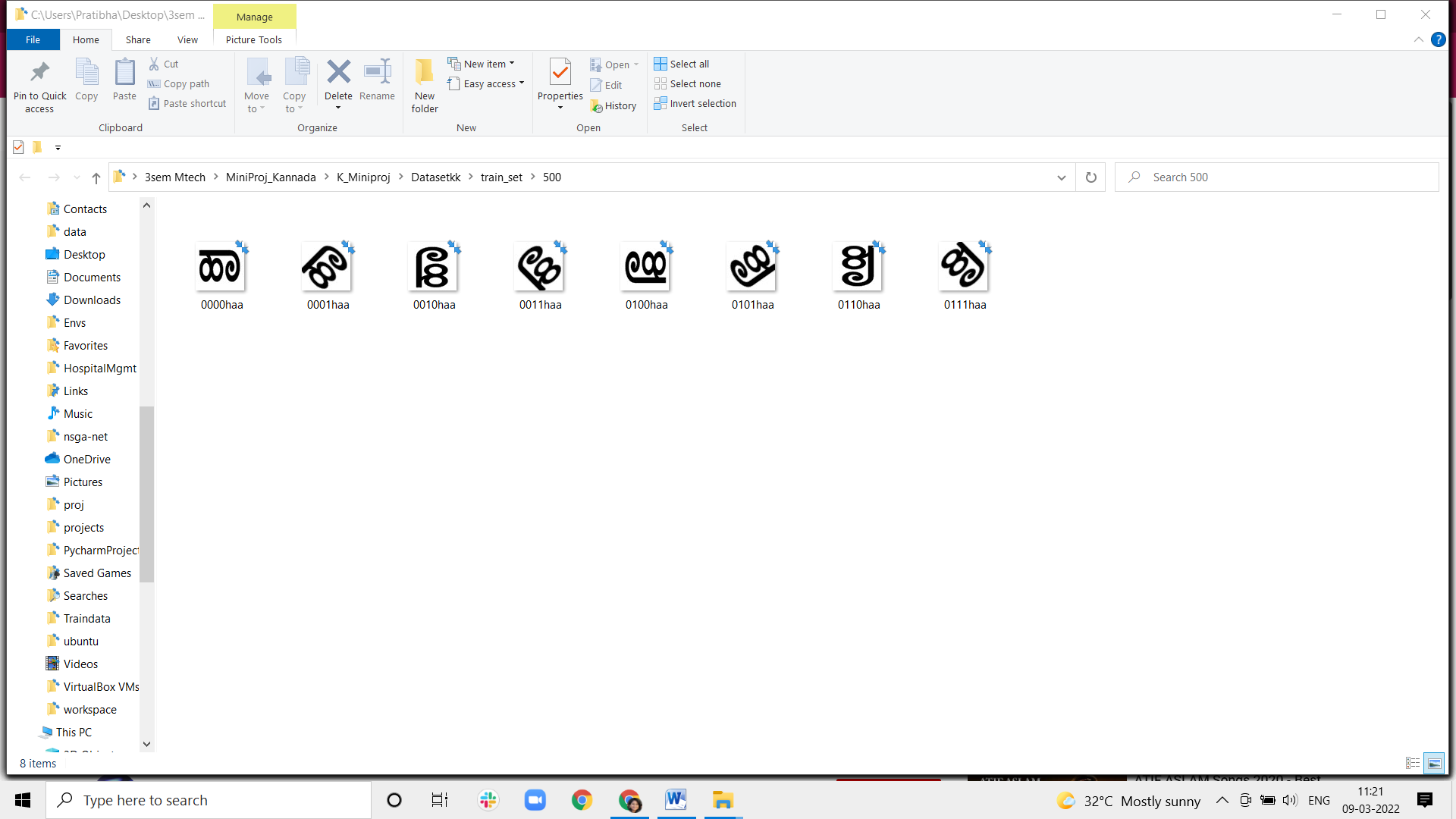
# 

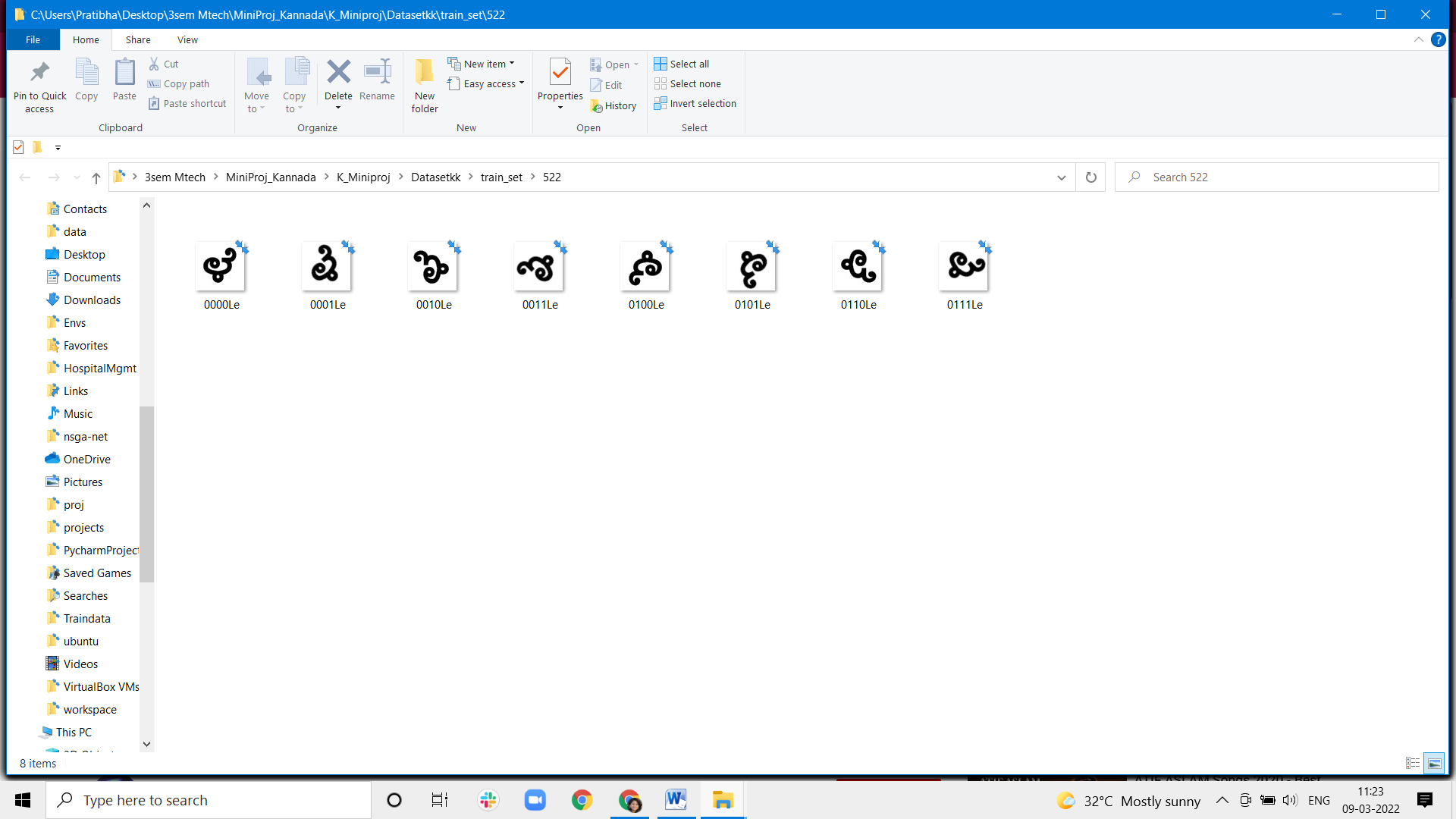












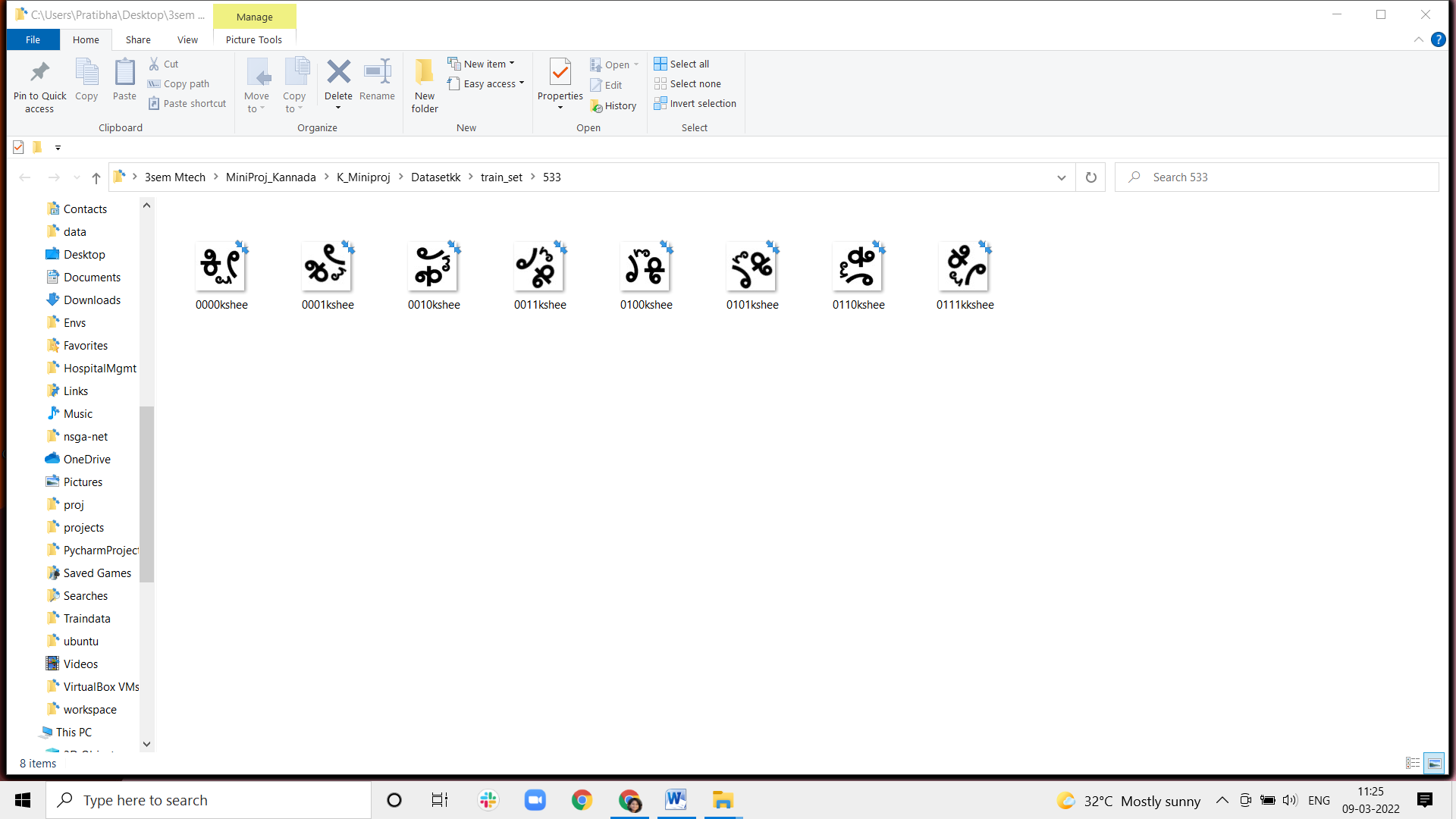


Figure5.3: Sample Dataset images with augmentation

**Chapter 6**

## RESULTS AND ANALYSIS

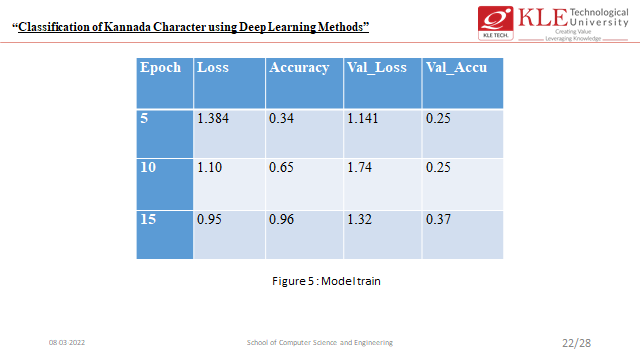


Figure 6.1:Training accuracy ,loss,validation loss,validation accuracy of

CNN model for epochs 5,10,15

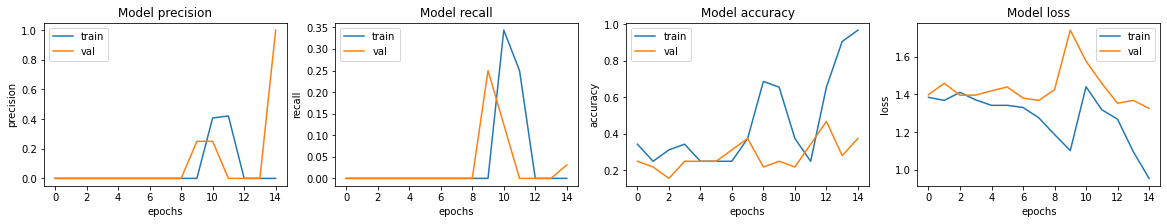


Figure 6.2: Model accuracy

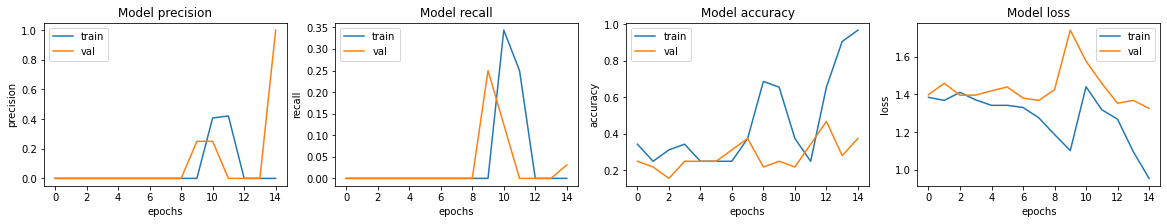
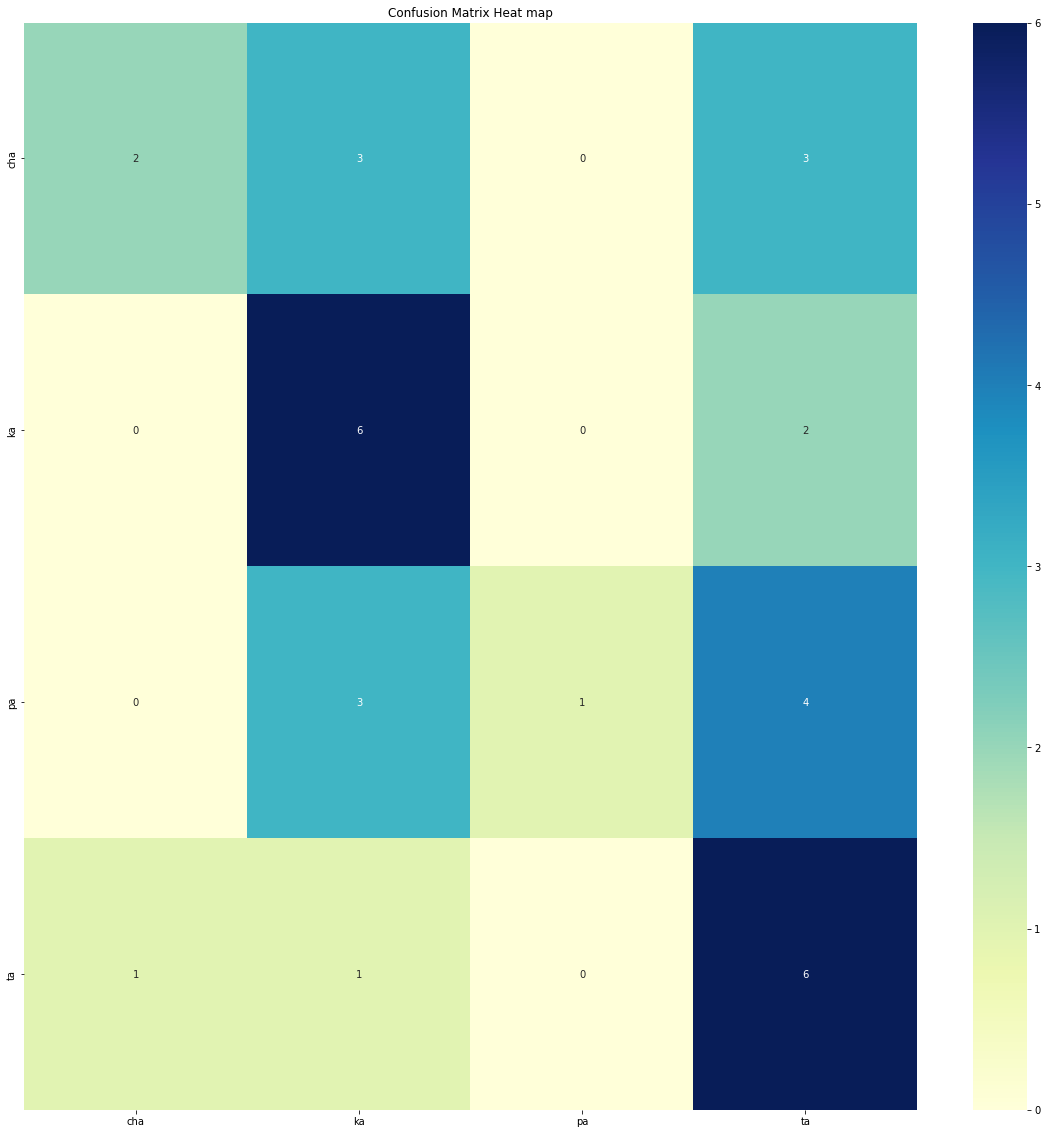


Figure 6.3 : Model Loss



## Figure 6.4: Confusion Matrix with Heat Map

## CHAPTER

## CONCLUSION

CNN was initially applied to character recognition task CNN and its variants are gradually adapted to various applications. CNN is designed to imitate human visual processing, and it has highly optimized structures to process 2D images. Furthermore, CNN can effectively learn the extraction and abstraction of 2D features. In detail, the max-pooling layer of CNN is very effective in absorbing shape variations.

The Deep Learning Pipeline required for implementing the sequential model for that Collect the Dataset required to implement the system, import the all-basic libraries and Select the Appropriate model which can recognize the handwritten grapheme and digit recognition in the given image correctly.

Train the model by specifying the number of “Epoches” and by specifying the number of classes here in our model we have used number of classes is 4 and number of epochs is 15 for for kannada character recognition and save the model as and “kannada.h5” after saving model Load the model will correctly recognize the handwritten graphemes and handwritten digit.

We have successfully built deep learning project on Bengali handwritten grapheme classification and Kannada handwritten digit recognition. We have built and trained CNN which is very effective for image classifications.

## 

## REFERENCES

[1] Sandhya, N., and R. Krishnan. "Broken Kannada character recognition—a neural network based approach." In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pp. 2047-2050. IEEE, 2016.

[2] Naveena, C., and VN Manjunath Aradhya. "Handwritten character segmentation for kannada scripts." In *2012 World Congress on Information and Communication Technologies*, pp. 144-149. IEEE, 2012.

[3] Chaithra, D., and K. Indira. "Handwritten online character recognition for single stroke Kannada characters." In *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 548-552. IEEE, 2017.

[4] Fernandes, Roshan, and Anisha P. Rodrigues. "Kannada Handwritten Script Recognition using Machine Learning Techniques." In *2019 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*, pp. 1-6. IEEE, 2019.

[5]Ragha, Leena R., and M. Sasikumar. "Adapting moments for handwritten Kannada Kagunita recognition." In *2010 Second International Conference on Machine Learning and Computing*, pp. 125-129. IEEE, 2010.

[6]Niranjan, S. K., Vijaya Kumar, and Hemantha Kumar. "FLD based unconstrained handwritten Kannada character recognition." In *2008 Second International Conference on Future Generation Communication and Networking Symposia*, vol. 3, pp. 7-10. IEEE, 2008.

[7]Shwetha, D., and S. Ramya. "Comparison of smoothing techniques and recognition methods for online Kannada character recognition system." In *2014 International Conference on Advances in Engineering & Technology Research (ICAETR-2014)*, pp. 1-5. IEEE, 2014.

[8] Aravinda, C. V., and H. N. Prakash. "Template matching method for Kannada Handwritten recognition based on correlation analysis." In *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 857-861. IEEE, 2014.

[9] Sridharamurthy, S. K., and HR Sudarshana Reddy. "PCA based feature vector for handwritten kannada characters recognition." In *2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, pp. 423-428. IEEE, 2015.

[10] Madhavaraj, A., A. G. Ramakrishnan, HR Shiva Kumar, and Nagaraj Bhat. "Improved recognition of aged Kannada documents by effective segmentation of merged characters." In *2014 International Conference on Signal Processing and Communications (SPCOM)*, pp. 1-6. IEEE, 2014.

1. Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa- net: Training low bandwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.
2. Baozhou Zhu, Zaid Al-Ars, and H Peter Hofstee. Nasb: Neural architecture search for binary convolutional neural networks. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.
3. Xie, Lingxi, and Alan Yuille. "Genetic cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 1379-1388. 2017.
4. Kaoutar, Sinai, and Ettaouil Mohamed. "Multi-criteria optimization of neural networks using multi-objective genetic algorithm." In *2017 Intelligent Systems and Computer Vision (ISCV)*, pp. 1-4. IEEE, 2017.
5. Deb, Kalyanmoy, Amrit Pratap, Sameer Agarwal, and T. A. M. T. Meyarivan. "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE transactions on evolutionary computation* 6, no. 2 (2002): 182-197.
6. Srinivas, Nidamarthi, and Kalyanmoy Deb. "Multiobjective optimization using nondominated sorting in genetic algorithms." *Evolutionary computation* 2, no. 3 (1994): 221- 248.
7. Johnson, Franklin, Alvaro Valderrama, Carlos Valle, Broderick Crawford, Ricardo Soto, and Ricardo Ñanculef. "Automating configuration of convolutional neural network hyperparameters using genetic algorithm." *IEEE Access* 8 (2020): 156139-156152.