T.C.

MARMARA UNIVERSITY

FACULTY of ENGINEERING

CSE4088 Introduction to Machine Learning

Final Report

---

28 December 2018

"Recognition of Handwritten Köktürk Digits"

HİLAL EKİNCİ - 150114057

OĞUZHAN BÖLÜKBAŞ - 150114022

1. Abstract

Our project is based on handwritten recognition. Handwriting recognition (HWR) is the ability of a computer to receive and interpret intelligible handwritten input from sources such as paper documents, photographs, touch-screens and other devices [1]. In this project, we are working on handwritten digit recognition. The digits which we are dealing with for this project is Köktürk digits. Köktürks were the Turkish lineage who lived as a sovereign nation under the name Köktürk State in Ötüken, Central Asia between AD 552-745 [2]. They used different alphabet and numbers which is not similar to the Turkish alphabet and digits we currently use. Because there is no previous work on Köktürk digit or alphabet recognition, we are interested in Köktürk digits and we decided to prepare a machine learning project about Köktürk digits. We will design a Köktürk handwritten digit recognition project using a convolutional neural network (CNN) as a machine learning technique in Python.

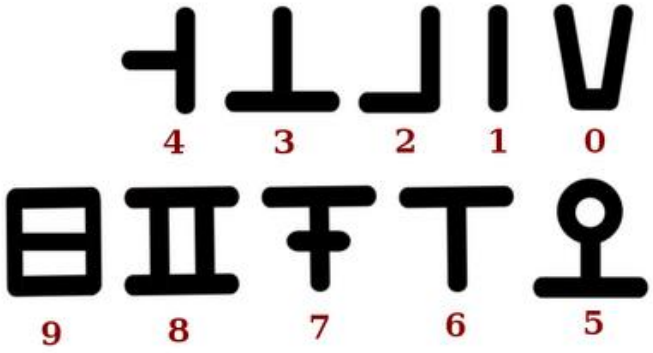


Figure 1: Köktürk Digits [3]

2. Overview of Project

There are a lot of works about handwritten digit recognition, most of work are based on Latin digits which we currently use. Our project is different from those because we are dealing with different digits which are Köktürk digits shown in Figure 1. We researched works on handwritten digit recognition different from Latin digits. For example, in the work of S. Azeem and end et al [4], they worked on Arabic handwritten digit recognition by using multi-class nonlinear SVM and they obtained 98.75% accuracy. In another example, Z. Alom and et al. [5] worked on Handwritten Bangla Digit Recognition by using CNN and they obtained 98.64% accuracy. As no work on Köktürk digit, there is no dataset publicly available. So, we are responsible also for building dataset of Köktürk digits, this is the one of our subtasks. After building dataset, we implemented a convolutional neural network (CNN) to classify Köktürk digits.

2.1 Project Plan and Schedule

We divided our project into 5 subtasks which are declared below. GANNT chart of our management plan is shown Table 1. We both worked and made efforts on each task. The percentage of work sharing is shown in the Table 2.

Task 1: Literature survey on handwritten digit recognition

Task 2: Building our own dataset

Task 3: Designing and implementing the CNN model

Task 4: Training the CNN model with our dataset

Task 5: Testing the CNN model with new data

	Week 1	*Week 2*	*Week 3*	*Week 4*	*Week 5*	*Mid-term Report*	*Week 6*	*Week 7*	*Week 8*	*Week 9*	*Week 10*	*Final Project Report*
Task 1												
Task 2												
Task 3												
Task 4												
Task 5												

Table 1: GANNT chart our management plan

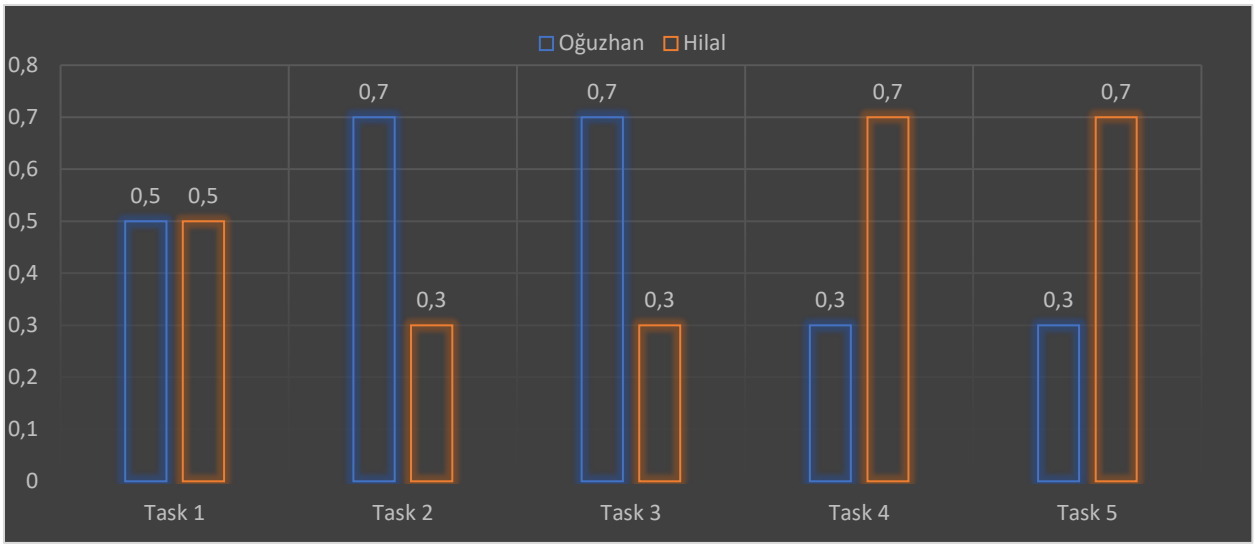


Table 2: Work Sharing Chart for our management plan

3. Project Accomplishment

3.1 Subtask Accomplishment

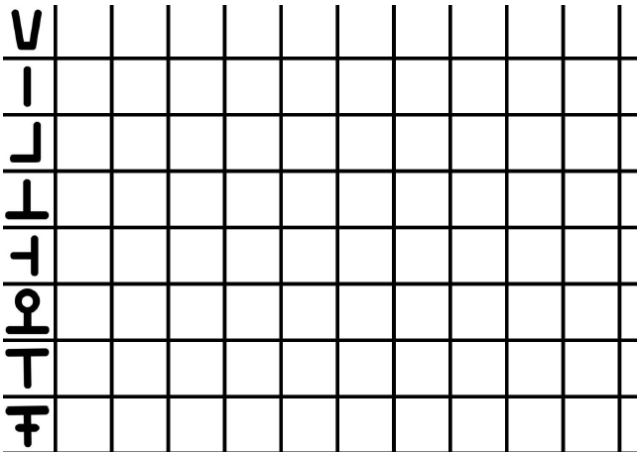
3.1.1 – Task 1: Literature survey on handwritten digit recognition

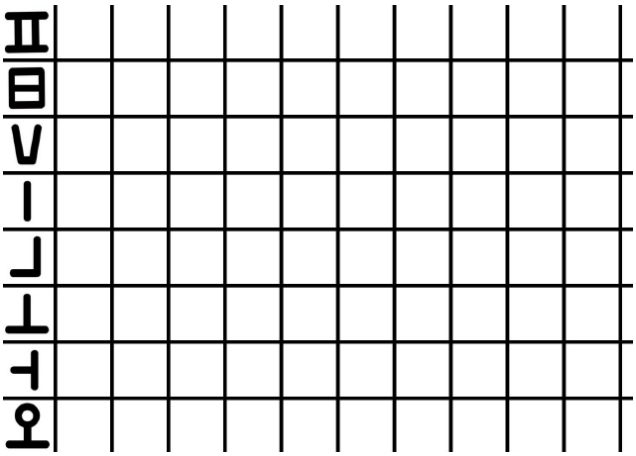
We started our project by researching the handwritten digit recognition projects and their results and accomplishments. We saw that many works were done on this subject, for example, Arabic handwritten digit recognition [6], Indian handwritten digit recognition [7] etc. We have researched stages of multiple image classification such as preprocessing, feature extraction, classification, and recognition. After the literature survey, we learned how the classification of handwritten digit recognition is done, so this task is finished. Designing and implementing machine learning model will easier and clearer after this part.

There is a lot of work about handwritten digit recognition so far including dataset (e.g. MINIST dataset [8]). However, there is no work on Köktürk digits so, we had to build our own dataset.

3.1.2 – Task 2: Building our own dataset

The main difficulty of our project "Handwritten Köktürk Digit Recognition" was building the dataset because there is no dataset publicly available. Firstly, we prepared empty sheets which are shown below to gather samples of handwritten Köktürk digit samples from different people.





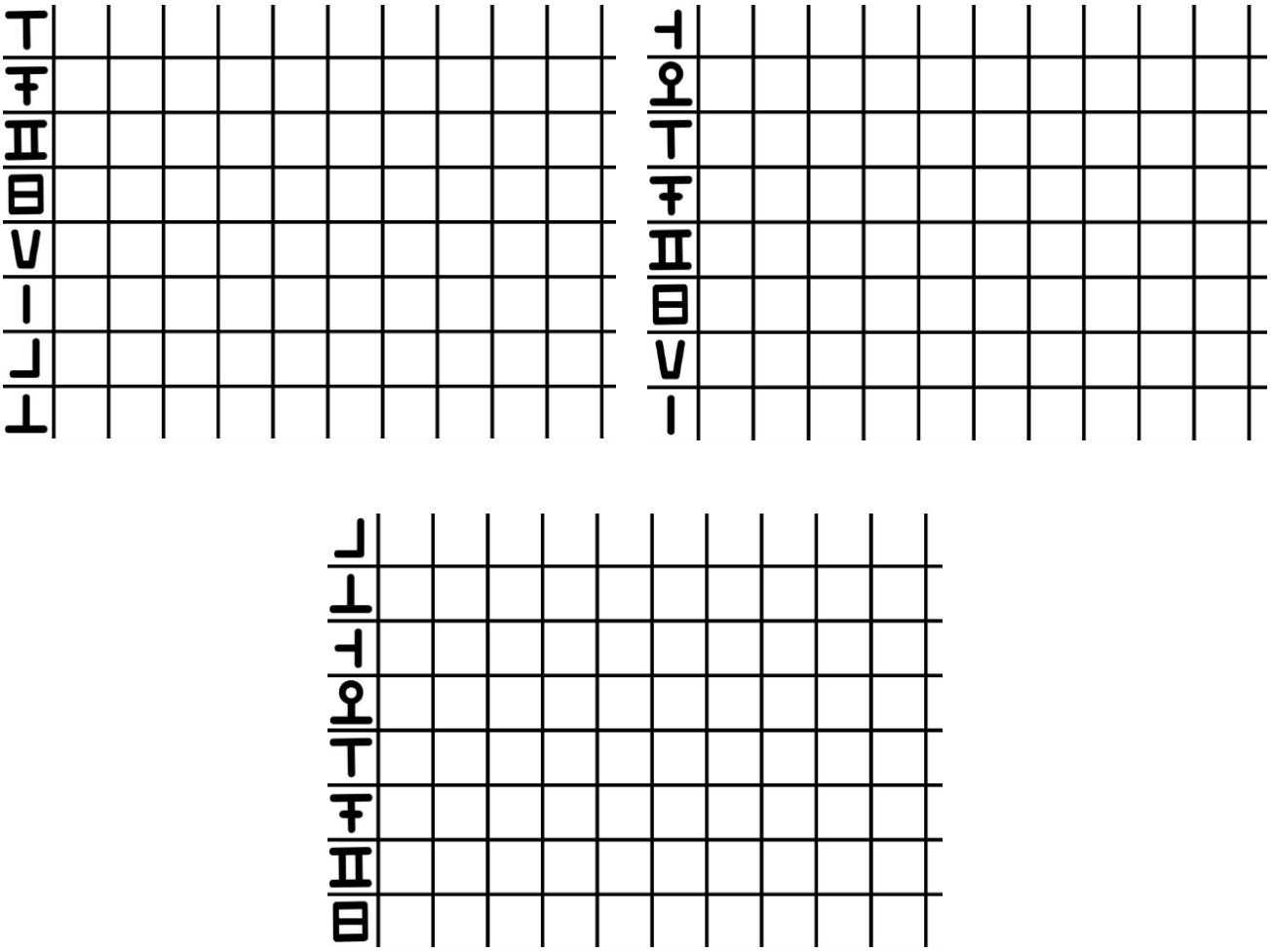
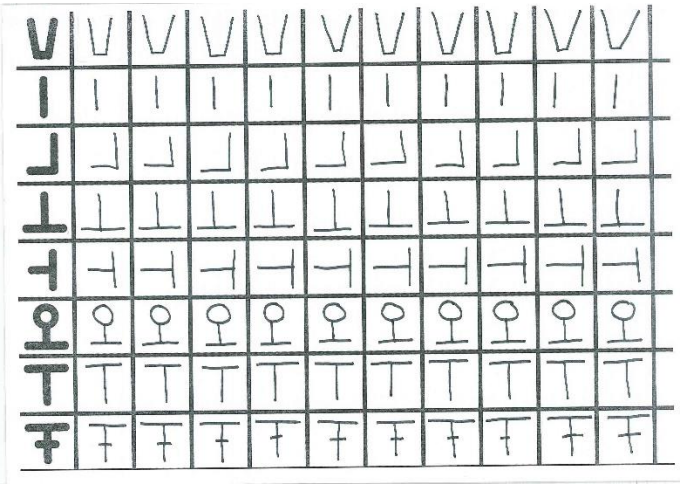
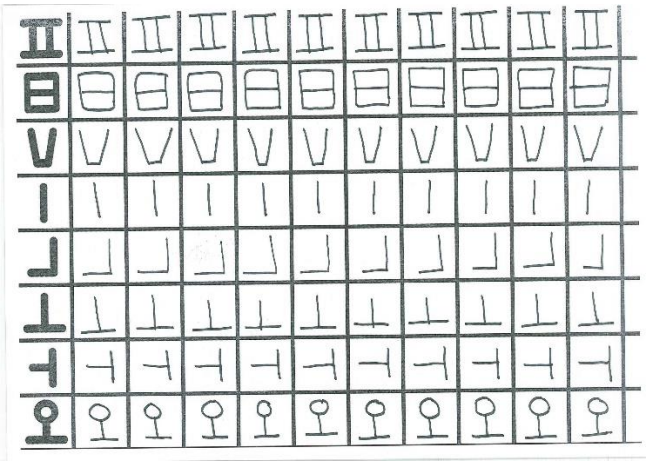


Figure 2: Our sheets for building the dataset

Secondly, we gave the sheets to people and we gathered a lot of samples of handwritten Köktürk digit samples which are shown below from about 30 different people.





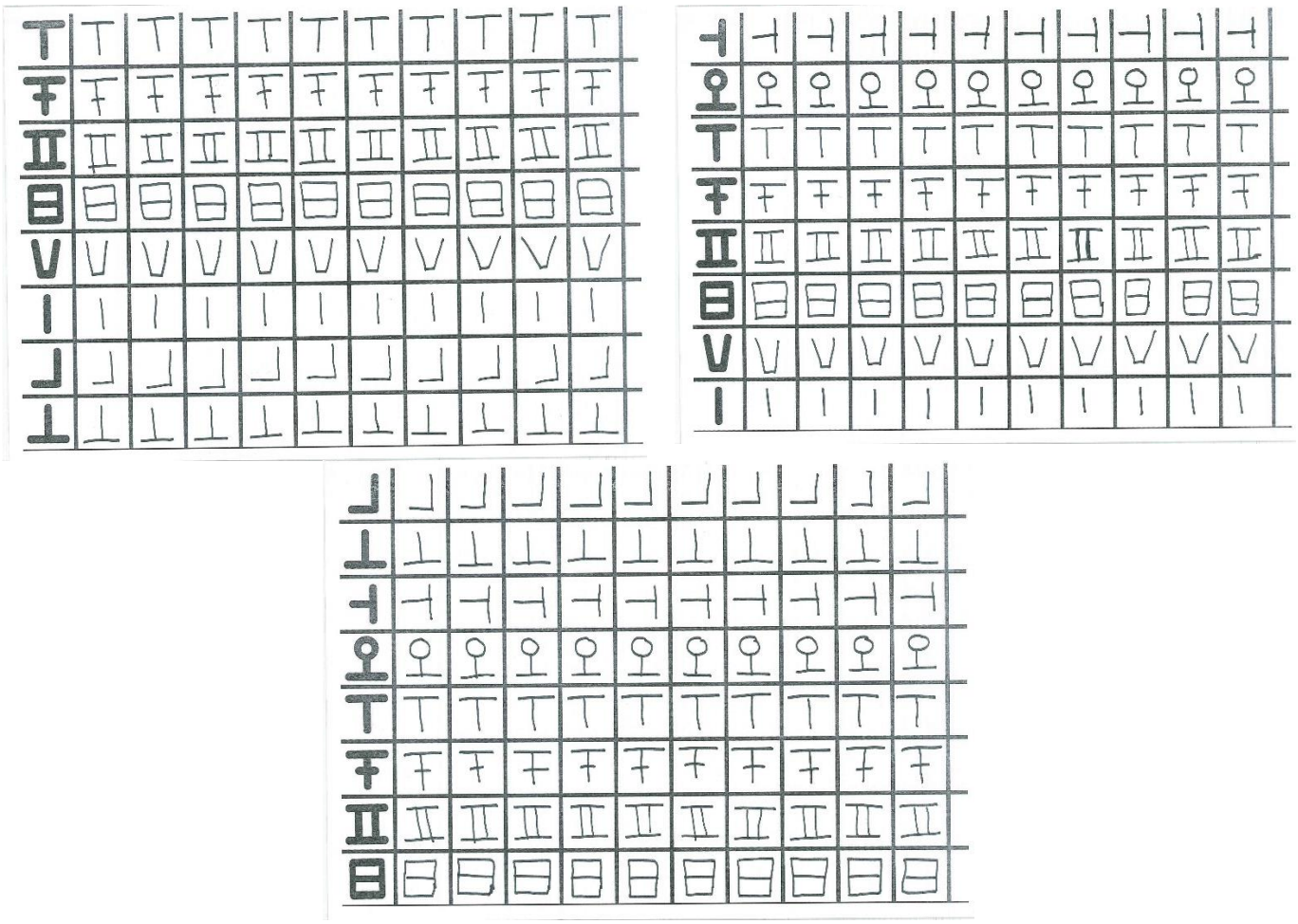
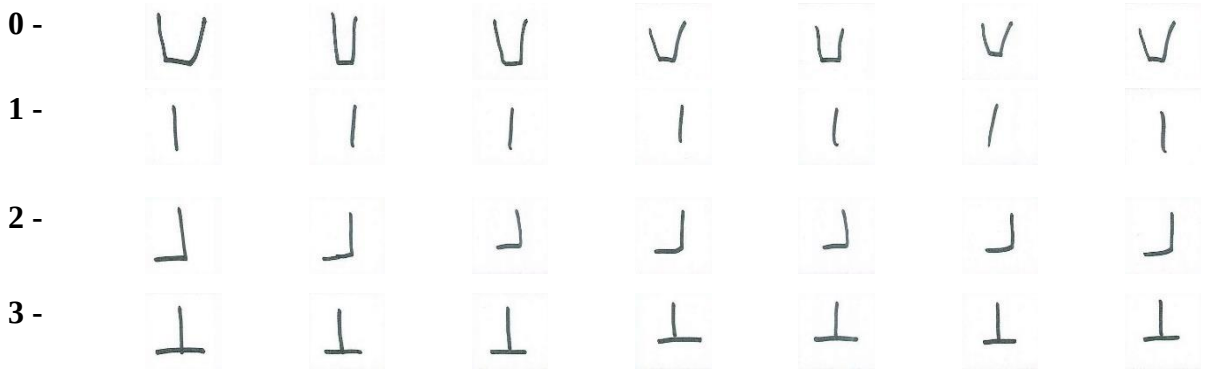
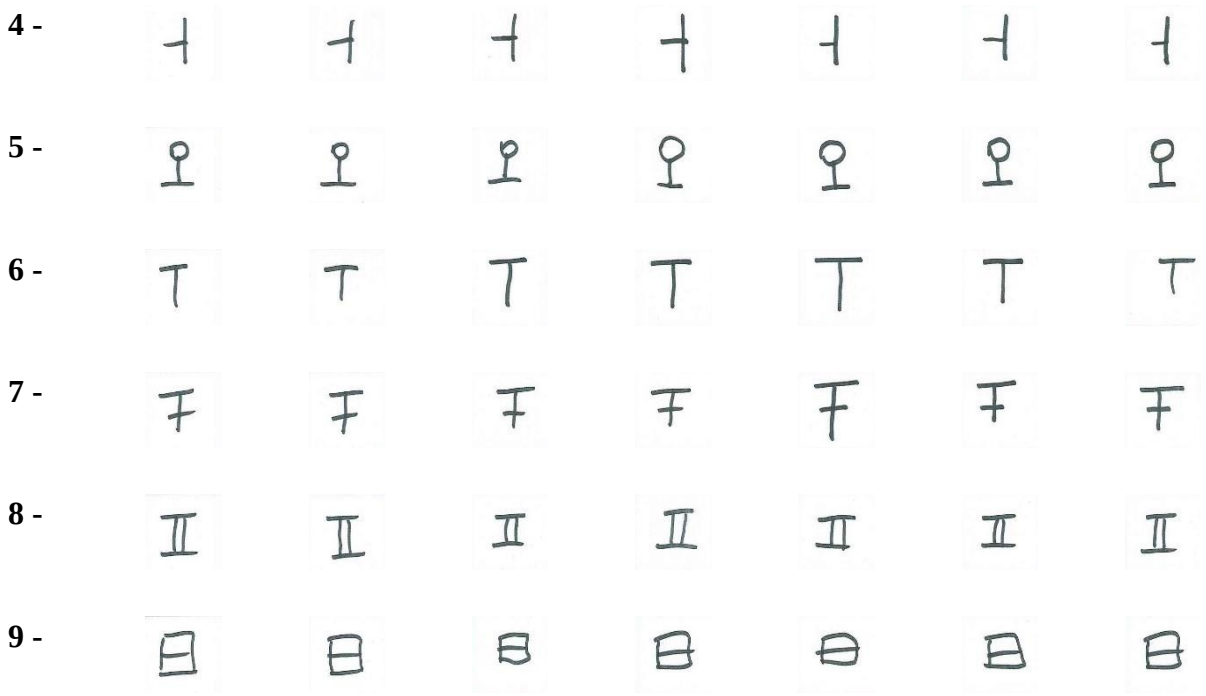


Figure 3: Handwritten Köktürk digit samples from different people

Finally, we implemented a Python script to crop the sheets filled by people and we built our Handwritten Köktürk dataset.

A sample of Köktürk Digit Dataset





3.1.3 – Task 3: Designing and implementing the CNN model

Firstly, we use OpenCV for this project to prepare the image data and convert the images into n-dimensional pixel arrays then, we convert our image data into a numpy array of size 28x28. After resizing, all the images are converted into grayscale images because the color of the digits doesn't matter in our project. The flowchart of our project design is shown in the Figure 4.

We designed and implemented a convolutional neural network (CNN, or ConvNet) because CNN is very good for analyzing visual imagery. We have three convolution layers with a single stride, zero padding, and relu activation. After three convolution layers, we have one dropout layer and this is to avoid overfitting problem. We trained our model for 50 epochs which is shown in the Figure 5 (for every epoch the model will adjust its parameter value to minimize the loss.)

- **Technologies used in this project**

This project was written using Python 3.6.6 in JetBrains PyCharm Community Edition as Python IDE.

Many libraries were used in this project such as os, numpy, opencv. For our CNN implementation, we used Keras using backend TenserFlow.

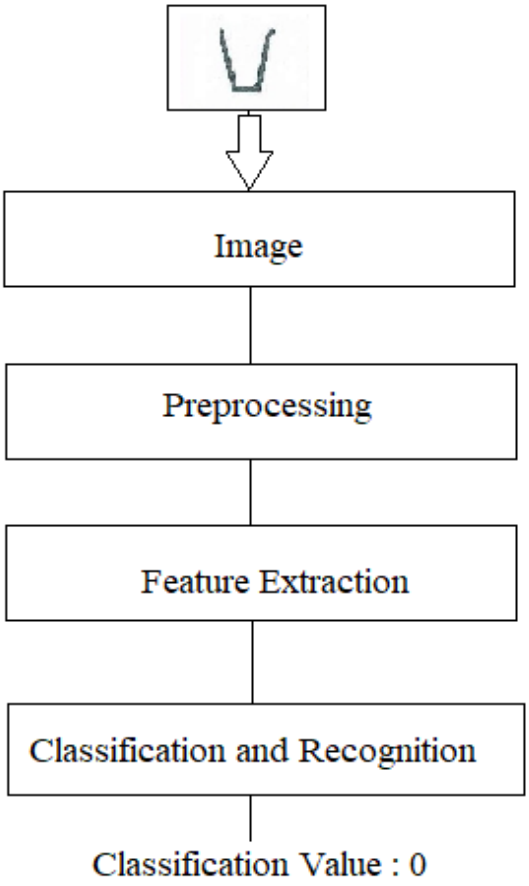


Figure 4: Flowchart of handwritten Köktürk digit recognition system

```
Epoch 50/50

 100/1000 [==>............................] - ETA: 0s - loss: 0.0280 - acc: 0.9900
 200/1000 [=====>.........................] - ETA: 0s - loss: 0.0199 - acc: 0.9950
 300/1000 [========>......................] - ETA: 0s - loss: 0.0194 - acc: 0.9933
 400/1000 [===========>...................] - ETA: 0s - loss: 0.0151 - acc: 0.9950
 500/1000 [==============>................] - ETA: 0s - loss: 0.0123 - acc: 0.9960
 600/1000 [=================>............] - ETA: 0s - loss: 0.0107 - acc: 0.9967
 700/1000 [====================>.........] - ETA: 0s - loss: 0.0095 - acc: 0.9971
 800/1000 [=======================>......] - ETA: 0s - loss: 0.0212 - acc: 0.9963
 900/1000 [==========================>...] - ETA: 0s - loss: 0.0191 - acc: 0.9967
1000/1000 [==============================] - 1s 1ms/step - loss: 0.0250 - acc: 0.9960
```

Figure 5: Output of our keras model summary after 50 epochs

3.1.4 – Task 4: Training the CNN model with our dataset

We have divided the dataset into training and testing datasets. We have a total of 1000 images as our training dataset including 100 samples from each class. We have trained our convolutional neural network (CNN) with 1000 images.

3.1.5 – Task 5: Testing the CNN model with new data

We have a total of 200 images as our testing dataset including 20 samples from each class. We have tested our convolutional neural network (CNN) with 200 images.

4. Experiments

We trained our convolutional neural network model with different number of epochs and compared the accuracy values we have obtained. At maximum, 50 epochs are enough, because the accuracy is very high when we trained our model with 50 epochs. We have only 3 convolution layers. We did not need to increase the number of layers because we have already reached our goal of obtaining at least 90% accuracy (shown in Table 3 below).

Number of CNN Layers	Number of Epochs	Training Dataset	Testing Dataset	Training Time Taken	Accuracy
3 layers	10	1000	200	13.4758 sec	91%
	20	1000	200	30.8656 sec	95%
	50	1000	200	68.3884 sec	99.5%

Table 3: Comparison of experiments in this project

5. Summary

In this project, we do handwritten digit recognition by using CNN. This project is different than other works because the digits which we want to recognize are not the same with the digits that we use today. We applied handwritten digit recognition to digits which our ancestors Köktürks used. Therefore, we had no dataset of handwritten Köktürk digits. We built our own dataset by gathering samples of handwritten digits from different people. With 1000 images as training dataset and 200 images as a testing dataset (shown in Figure 6), we obtained very good results. Our program classified 199 images correctly out of 200 images. There was only a digit image is misclassified, the program classified a digit with label '6' instead of the actual value '**7**' (shown in Figure 7). This means 99.5% accuracy we have obtained in our project.

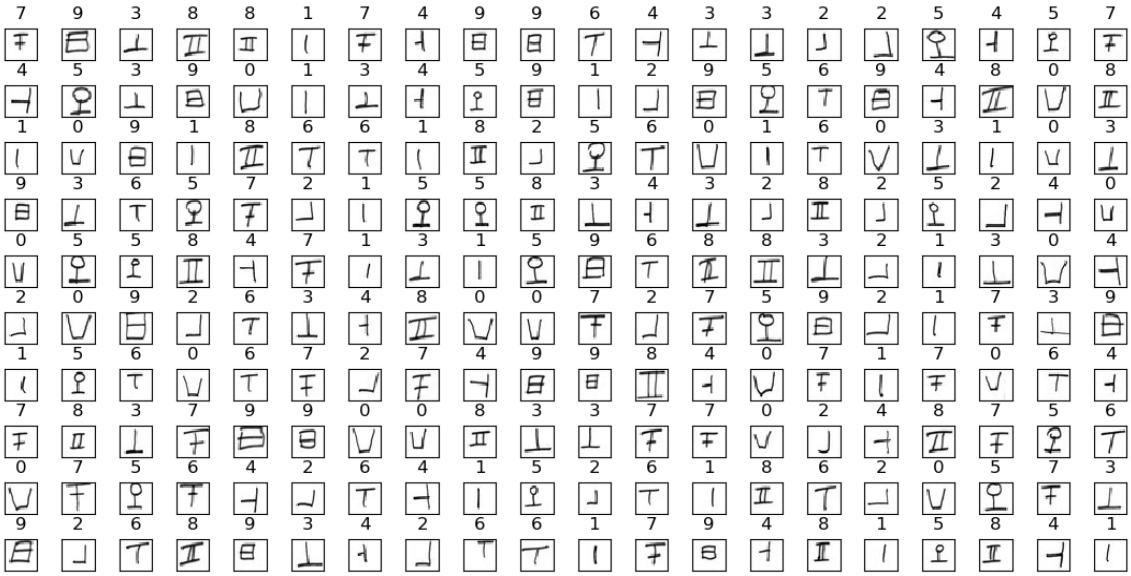


Figure 6: Output of our program

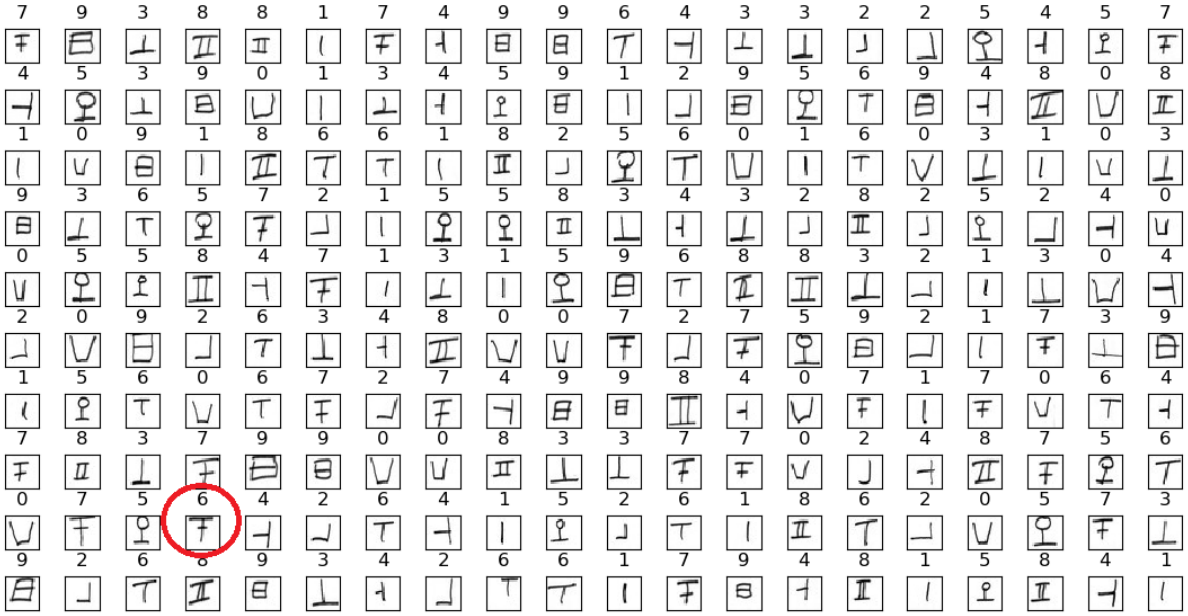


Figure 7: The misclassified digit image

5.1 Demo Part and Results

After presentation in demo part, we used images completely different from training images and testing images. The images of demo were drawn in Windows desktop application "Paint" (shown in the Figure 8). For these 10 images, our program classified all images correctly in demo (shown in the Figure 9).

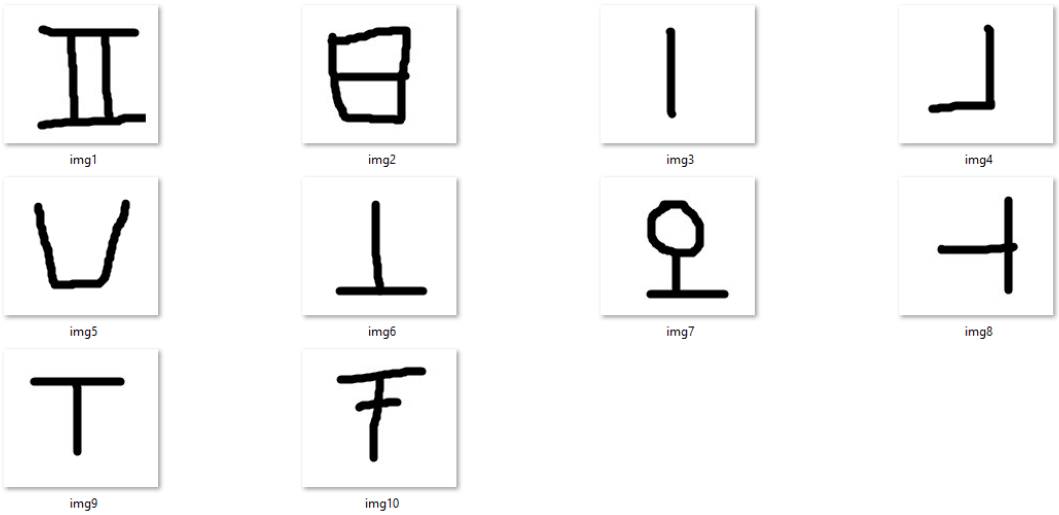


Figure 8: Images drawn for demo

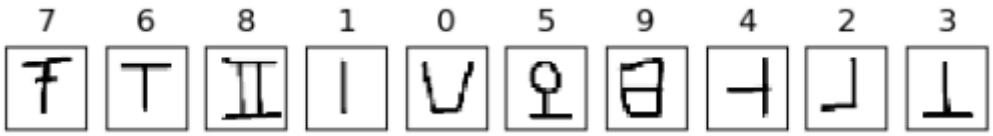


Figure 9: Output of testing in demo

5.2 Future Work

We have obtained very high accuracy in our project so, we do not need any enhancement for our project for example increasing the convolution layers or adding different filter to CNN. However, we can automate this project more. As a future work, this project can be implemented as a real-time application. With a real-time video or real-time computer screen inputs can be used to recognize handwritten Köktürk digits. In addition, we made recognition of only one image of handwritten Köktürk digit. We can develop this project to recognize the number of digits greater than 1 image so, the program can recognize multiple digits simultaneously.

6. References

[1] Handwritten recognition, Wikipedia [Online] Available: https://en.gowikipedia.org/wiki/Handwriting_recognition (Date of Access: 20.10.2018)

[2] Köktürkler, Wikipedia [Online] Available: https://tr.wikipedia.org/wiki/Göktürk_Kağanlığı (Date of Access: 16.10.2018)

[3] Blog of Researches on Köktürk Digits, [Online] Available: http://kokturukce.blogspot.com/2008/09/trk-saylar.html (Date of Access: 16.10.2018)

[4] Azeem S., Meseery M. and et al. Online Arabic Handwritten Digits Recognition [Online] Available at: http://www.iapr-tc11.org/archive/icfhr2012/paper021.pdf (Date of Access: 25.12.2018)

[5] Alom Z., Asari V. and et al, Handwritten Bangla Digit Recognition Using Deep Learning [Online] Available at: https://arxiv.org/pdf/1705.02680.pdf (Date of Access: 25.12.2018)

[6] Sabri A. Mahmoud, Arabic handwritten digits recognition using Gabor-based features, IEEE, 2008

[7] Pauly L., Raj R., Paul B., Handwritten Digit Recognition System for South Indian Languages using Artificial Neural Networks, IEEE, 2015

[8] MNIST database, Wikipedia [Online] Available: https://en.gowikipedia.org/wiki/MNIST_database (Date of Access: 16.10.2018)