Conversion Of Handwriting To Text

Ahmet Onut Akman
Computer Engineering Department
Yıldız Technical University, 34220 Istanbul, Turkey
l1116059@yildiz.edu.tr

***Özetçe* —Bu çalışma insan eliyle oluşturulmuş bir el yazısı metnin dijital metne dönüştürülmesini işlevinde bir program ortaya koymayı amaçlar. Günlük hayatta ortaya çıkan verilerin dijitalleştirimesi için yapılan çalışmaların bir parçası olan karakter/dijit tanımlama çalışmaları sayesinde, üretilen yazılı belgelerin daha iyi ve daha düzenli saklanabilmesi mümkün olmuştur. Bu proje ile bu çalışma alanı üzerine araştırma yapılmış ve bu alana yönelik bir çalışma ortaya konmuştur.**
Program temel olarak görüntü işleme, karakter ayıklama, karakter tanımlama ve metin işleme olmak üzere 4 farklı bölümden oluşmaktadır. İnsan eliyle yazılmış bir metnin fotoğrafı, bu program içerisinde önce işlenir, sonra metindeki karakterler ayrıştırılır, ayrıştırılan karakterler EMNIST dataseti ile eğitilmiş olan CNN mimarisinde bir yapay sinir ağları modeli ile sınıflandırılır ve sınıflandırılan karakterler dijital metin haline getirilir. Bu metin kullanıcıya sunulmadan önce bir dizi hata ayıklama işleminden geçer. Kullanıcı en sonunda sağladığı görseldeki metni çıktı olarak alır.
Programın belirtilen modülleri farklı yaklaşımların denenmesi sonucu elde edilen en optimal yapının seçilmesi ile oluşturulmuştur.

Anahtar Kelimeler—El yazısı metnin dijital metne dönüştürülmesi, Yapay sinir ağları, CNN, karakter tanımlama.

***Abstract*—This study aims to present a program in the function of converting handwritten text into digital text. Thanks to character / digit identification studies, which are a part of the studies for digitizing the data that emerge in daily life, it has been possible to store the written documents better and more organized. By this project, research has been done on this field of study and a functioning product has been put out.**
The study basically consists 4 different parts: image processing, character extraction, character identification and text processing. In this program, the photograph of a handwritten text is processed, then the characters in the text are extracted, the parsed characters are classified by the artificial neural network model in CNN architecture, trained with the EMNIST dataset and the classified characters are converted into digital text. This text goes through a series of correction before it is presented to the user. The text which was obtained from the given image is presented as the output to the user.
The specified modules of the program were created by selecting the most optimal structure obtained by trying different approaches.

Keywords—Conversion of handwritten to text, Artifical Neural Network, CNN, character identification.

I. Introduction

Data and data management have gained importance as a result of population growth and rapid development of technology in our age. So much so that; Thanks to sensors that are all over our lives, the electronic devices we use, the content we share or highlight on the internet, and many other factors, existing data increases exponentially every day. In addition, that the rate of data depreciation is much slower than the rate of new data formation caused the data to accumulate. It is becoming more and more difficult to store and manage this phenomenon, which is rapidly increasing and diversifying. For this reason, the money spent on data storage and data management technologies has increased and studies for data management technologies have gained importance.

With the increasing and diversification of data day by day, our old data storage methods are no longer sufficient, technological devices have started to be used for this purpose and the physical space required to store a certain amount of data has been getting smaller and smaller in a few decades. Thanks to its physical size and low cost, it has been possible to capture and record the data that occurs every day in data storage units we have.

Although the development and cheapening of data storage units increase the amount of data that can be stored, the management and evaluation of the stored data has become more difficult. Today, even the memory of computers released for individual use can store data of a few terabytes. The data that has been recorded and required to be processed has reached a huge extent with the fact that everyone has the opportunity to store such large data, the commercial institutions have made huge investments in the data technologies and even the state structures have started to be digitalized. However, the stored data does not make sense unless it is useful. It is one of the biggest goals of information technologies to keep only the necessary data and to keep this data organized, fast, accessible and easy to use. The amount and complexity of data we have mentioned left the human brain inadequate for this purpose. As a result of this inadequacy, data management technologies have emerged and these technologies are among the fastest developing technologies of today.

Although the development of data storage and data processing technologies is fast, it is not correct to say that these technologies have the capacity to fully meet our needs. The studies on some areas are insufficient, even if the studies in some areas are sufficient, the resulting products have not become easily accessible to everyone yet. Many developed countries in many parts of the world have not been able to move forward in the digitization process, which has made the available data still difficult to access. Difficult accessible data slows down daily life and increases the required manpower.

In addition to these shortcomings, there are still structures that still use these technologies and are still waiting to take full advantage of the conveniences these technologies can offer. An examples of this are police units. Police

departments in many developed countries make use of the data they have accumulated in order to investigate crimes and identify those responsible. Every event that is analyzed and concluded also generates new data. However, the data storage and processing studies have not been fully digitalized and unfortunately it will not be possible to reduce the error rate of the human brain to zero.
Today, thanks to the advancedness of technology, we are able to obtain much more efficient results by processing physical data created by human hand better. Examples of these are studies based on face, text and voice perception / identification. Considering the computing power of the computers we have today and the amount of data they can store, it will be concluded that the potential of these studies is huge.
One of the most popular pursuits of these fields of study is the analysis of human-made drawings. A human brain has the potential to read the text written by another person and to identify the person who wrote it by comparing it with the handwriting of the people he knows. The biggest aim of the efforts we have mentioned is to create this skill in digital environment, reduce the error rate as much as possible and to produce products that provide a wide range of services by making use of the amount of data that computers can store.
In this project, our focus will be to perceive and define a text written on a paper by a human hand. These and such studies undoubtedly have the potential to help us store data more efficiently and access it more easily.
Of course, the technologies we have are quite enough to keep a photo of these documents. However, every data we store is not required as a whole. Storing such documents as text documents will allow a document with known content to be found and changed more easily, the content of the document will be converted to different formats and can be easily cited. Moreover, this technology is used and used not only for storing documents but also in many different areas such as instant translation, instant writing / sign detection, dictating and analyzing perceived writing.
This project will not be a first in its field. The most important goal in the realization of this project is to put forward a study with high accuracy and to add a perspective to this development-oriented occupation and to lay a good foundation for the studies I will present in this field in the future.

II. Related Works

Although image processing based studies are becoming popular rapidly, products using these technologies have entered our lives a long time ago. Many programs with image processing that we use today can select and extract faces, colors, objects, fonts, numbers and many more out of its context. Although it is difficult to compete with the capacities of a human, this field, which is extremely open to development and whose works are exciting, will undoubtedly take a much larger place in daily life with future studies.
Applications that were out forward in this field have taken their place in daily life, and they have attracted the attention of people and attracted many curious people who are in the world of programming. Some of these applications have already been one of the essential parts of our daily life. While one of these applications can detect the texts in an area shown by the camera and translate these texts instantly, another one analyzes the handwriting you show[1], and gives another chance to distinguish parts of the human face and modify the proportions of the parts on that face[2]. Undoubtedly, these applications are the cornerstones of more complex technologies that will emerge in this field in the future.
The robotic field is one of the areas that have become popular in recent years[3]. As can be guessed, it is not far from image processing, the future impact of technology on human life is among the most discussed areas of study[4]. So much so that the successful result of a study in robotics somewhere in the world can become known globally within days. Well, if the products of these works that excite everyone are not able to use the functions they can perform on site, what will be the place of these products in our lives? Undoubtedly, the first skill a mechanical robot needs to be functional is to be able to collect data and process it well. Nowadays, many devices start working as soon as you press the button. However, the first thing people imagine when it comes to the future is not the machines that do their job better when the button is pressed, but the devices that know when to function without a button press.
The subject to be worked on in this project is also a very important part of many exciting fields of studies, such as the robotics mentioned earlier. It was decided to work on such topic with the excitement of the idea of instilling the ability to read text on a machine just like a human being. One of the aims of this project is to have an idea about whether it is possible to increase the accuracy rates of the projects carried out so far.

III. Image Processing

In order for the image given by the user to be processed, it must be passed through certain operations. Although the user is expected to provide as little visual noise as possible, it is not always possible to reduce this to zero in real life. For this reason, the need to reduce the noise of the image arises.// After that, in order to understand the text and background in the photo in the most accurate way, the photo must go through binerization. As a result of this process, a clean, highly recognizable text that is completely against the background was obtained. Python's image processing libraries are extremely sufficient for this process. It was decided to use Python's PIL, Numpy libraries for these processes.

Figure 1 Example image passed through the filter.

A. Putting Letter Images In A Suitible Shape For The Prediction Stage

Two different methods, which can be used to reduce the size of a letter image that has been square but not

yet suitable for processing by the model to the desired 28x28 dimensions, have been compared and the results of these methods have been examined. As a result of this comparison, it was concluded that the resizeimage library is more functional.

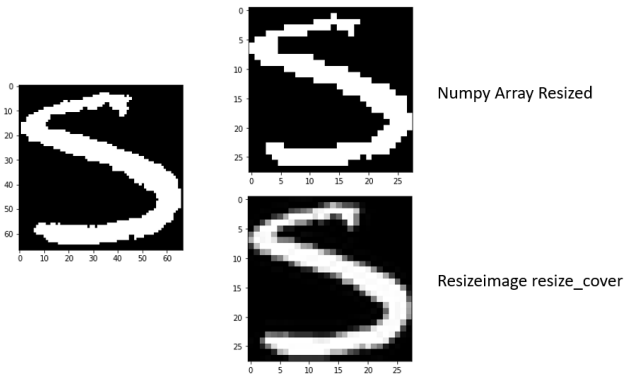


Figure 2 Two methods for downsizing an image.

After the processes of passing the input image through the image filter, extracting letters and processing letter images, the letter images have become ready to be classified by the model.

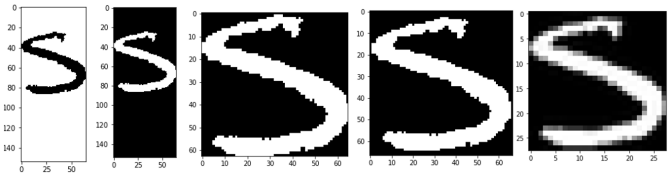


Figure 3 The complete journey of the character image from extracting to classifying steps.

IV. Artificial Neural Network Architecture

A. Data Set

Undoubtedly, in order for each neural network structure to infer, it must first have seen data similar to the data it will extract and learned what it is. Likewise, the programmer will still need sample data to observe the correctness of the structure created. It should be noted that the quality of the prediction efficiency of the said structure and the correct training of the model depend on a quality data set. Considering that the EMNIST data set has become a standard and gives positive results in similar projects, it has been decided to use it.
EMNIST is a handwritten digit / character dataset based on NIST dataset. EMNIST is an enlarged and enriched version of the MNIST dataset with a narrower data width, which only consists of digits. EMNIST is seperated into classes which have different amount of data from different types. It is concluded that the most appropriate class for this project would be the class named "Balanced", which has equal number of drawings from each letter / digit.[5]

Name	Classes	No. Training	No. Testing	Validation	Total
By_Class	62	697,932	116,323	No	814,255
By_Merge	47	697,932	116,323	No	814,255
Balanced	47	112,800	18,800	Yes	131,600
Digits	10	240,000	40,000	Yes	280,000
Letters	37	88,800	14,800	Yes	103,600
MNIST	10	60,000	10,000	Yes	70,000

Figure 4 EMNIST classes.[5]

B. Neural Network

According to the researches and projects investigated, it was concluded that CNN architecture is extremely useful in image identification / classification studies. Therefore, it is concluded to work with a CNN architecture.

1) Keras: Within the scope of the project, Keras library was used for the neural network structure.
Keras is a neural network library built on an open and machine learning themed library called Tensorflow. It can work with Tensorflow as well as Theano. It is preferred because it is easy to use and tips can be found easily on the internet.

2) Hyperparameters: Even if the optimal setup for neural network architecture is provided, one of the factors that most affects the accuracy of a model is the so-called hyperparameters. While weights of interneuronal connections and values within the frames can adapt to the most optimal setting as the training process progresses, hyperparameters are the parameters chosen by the programmer and do not change during the training process. These parameters can be exemplified as Learning Rate, Epoch number, Hidden Layers, Hidden Units, Activations Functions, Batch Size.

3) Underfitting and Overfitting: Underfitting and Overfitting are two of the most undesirable consequences of the wrong selection of hyperparameters just mentioned. Undoubtedly, it is a situation that we aim to have a high accuracy rate during the training we selected. However, while the training process proceeds well, the validation accuracy remains low or, on the contrary, the level of accuracy is not good at all, will signal us for overfitting and underfitting situations. For this reason, it will be our priority to avoid this situation in model selection and hyperparameter selection.

4) Accuracy: One of the first values we need to look at will be accuracy. It is a situation that we want our model to process the validation data with high accuracy or to give high accuracy values after the test process.

$$Accuracy = \frac{Number\,Of\,Correct\,Predictions}{The\,Number\,Of\,Predictions} \quad (1)$$

However, as mentioned in the previous subtitle, correct interpretation of the accuracy value is very important. It should be noted that in the case of overfitting, a model will only produce correct results for training data, it will not be reliable in other uses, and its generalization ability is weak.

5) Loss: Another parameter to be considered is loss outputs. Loss are simply guessing errors our model makes

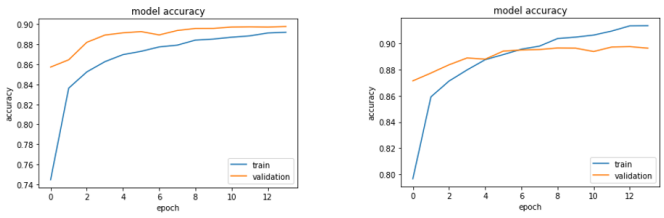
for an input. It should be remembered that the value of loss will not change in the inverse proportion to accuracy under all conditions.
Loss value is calculated with functions called Loss Functions. The training loss value also allows our model to adapt itself to the correct structure in the next training step.

V. Finding The Optimal Artificial Neural Network Architecture

The model being studied belongs to the digit identification project that Chris Deotte shared on Kaggle.com and worked on MNIST. Although the purpose of this project of Chris Deotte and the data set he used was different, the accuracy rate of 0.99757 he achieved was impressive and the thought of the digit identification project being adaptable to the character identification project was effective in this choice. It should be noted, however, that Chris Deotte used this model 15 times and achieved this success rate with the results of these 15 trained models.
Our goal was getting the best results we could out of this model which was built for completely some other purpose. We needed to monitor the effects of change of dropout rates, batch size, number of epochs. Also, we needed to see if data augmentation or the usage of LearningRateScheduler was any helpful.

A. Changing The Dropout Rate

Considering that the data set we use in our project is a much wider data set compared to the MNIST data set used in the project where the neural network architecture is taken, the first question asked was whether the dropout value of 0.4 gave a bad effect for the model to learn. The model has been trained in the same way as two different dropout values in order to measure the reaction of the model against dropout decreasing from 0.4 to 0.2. Since the purpose of these trainings is to observe the general attitude, a low epoch number has been chosen and the batch is left at the default value 32.

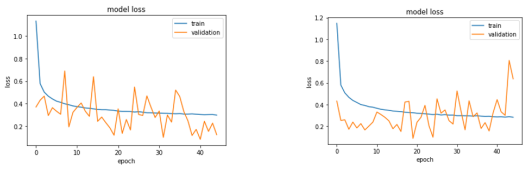


Dropout(0.4) Dropout(0.2)

Figure 5 The effects of the usage of a lower dropout rate.

B. Usage of LearningRateScheduler For Decreasing Learning Rate

In the original project, using 64 batch size and 45 epoch hyperparameter values, it was seen that training data was enriched with data augmentation and learning rate was decreased gradually by using LearningRateScheduler. In this step, it was wondered whether this type of training approach would be positive for this data set. For this, the model was trained just like in the original project, and then the results will be checked in case of training without the use of LearningRateScheduler. During these controls, a very similar validation accuracy was obtained and accuracy graphics were found to be almost the same. However, it is visible that with this approach, validation loss was not doing well.

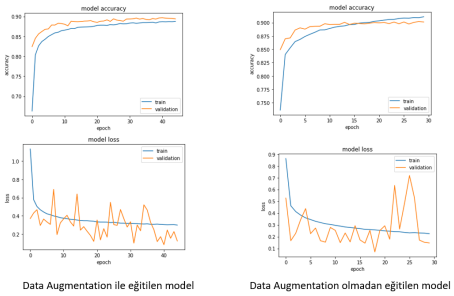


Data Augmentation Data Augmentation ve Azalan Learning Rate

Figure 6 The effects of the usage of a decreasing learning rate.

C. Data Augmentation

Data augmentation is an approach applied to teaching neural network structure more with the data in hand by changing the training data we have within the limits determined at each step. In the previous step, we saw that a model trained with decreasing learning rate and data augmentation failed against a model that uses data augmentation. In this step, we asked if a model trained with a data set containing approximately 110,000 data really needs data augmentation. For this purpose, the model trained with the data augmentation used in the previous step was compared with the trained version with the same batch size but without the data augmentation. Since the non-Augmentation model is more likely to memorize the data over time and overfit, the epoch number of the method with the augmentation was chosen as the same as the previous tryout , while the epoch number of the other model is set to be 30 (to be increased afterwards if successful results are obtained).



Data Augmentation ile eğitilen model Data Augmentation olmadan eğitilen model

Figure 7 The effects of the usage of data augmentation.

D. Increasing The Epoch Count

In the previous examples, although we caught a validation loss value that fell below the gradual loss value,

we could not encounter a drawing of a consistently falling validation loss. In this step, the question of whether this is due to insufficient number of epochs was asked. For this, the same model was put into a full 26-hour training with the same training parameters, replacing only the number of epoches with 150. In the new validation loss curve to be created, it will be checked at any time whether a better result can be obtained compared to the previous gauge. However, the result was not really moving comparing to the other graphic.

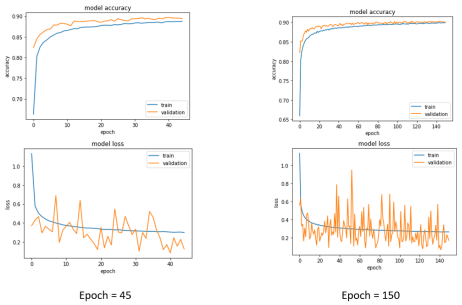


Figure 8 The effects of training the model for longer.

E. The Effect Of The Batch Size

In this last attempt, the question of how our results will change if the batch size trained with the 64 batch size we have is reduced to another popular batch size value of 32. The same training approach was used for this experiment, 64 batches were used in the training of only one model and 32 batches in the other.

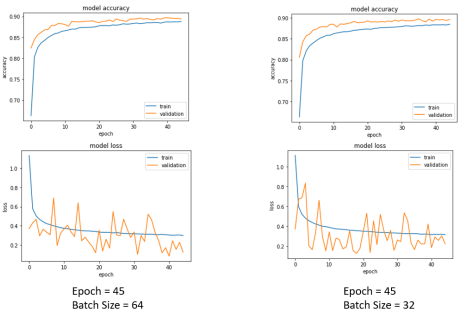


Figure 9 The effects of the usage of a smaller batch size.

As a result of these two different training, close validation accuracy values of 0.8940 and 0.8959 were observed. Validation loss values were obtained in both models (again) with ups and downs but with a tendency to decrease based on the average over a certain period of time.

F. The Final Decision

In the experiments, it is aimed not to decrease the "validation accuracy" value below 0.89 and try to go above 0.9 value, as well as to transform the unstable validation loss graph that we encounter in each trial into a more declining structure.

At the final point, as a result of all the trials shown (and the ones that are not shown here because they are not significant), the approach which was better than the model it has compared with almost every trial step, with 0.8940 validation accuracy, which tends to decrease its validation loss based on the average in a certain range as the number of epoch increases was chosen. This approach is the original model trained with 64 batch size, 45 epoch, 0.2 split rate of training / validation data and data augmentation technique. This decision was made not only based on the results in these graphs, but also based on the results it made against the actual trial data created.

VI. Error Fixing After The Classification Process

As described, it is highly probable that there are errors in model estimates, since a completely error-free model cannot be obtained in our CNN model. Our methods of reducing these errors; Inline blanks calculation, digit-letter confusion clearing and dictionary control steps for words are processed as shown in the picture.

```
-
Original: P$11$r$25$e$27$S$27$I$28$d$14$e$20$A$27$t$122$M
$20$I$20$n$11$J$19$S$13$t$23$e$18$r$76$J$27$U$20$d$16$g$17$e
P$9$0$11$Z$13$I$19$t$15$J$20$V$18$I$25$t$26$Y$122$C$9$a$21$r
$21$a$16$M$14$e$20$L$94$P$15$I$22$e

Modified: PreSIdeAt MInJSter JUdge
P0ZItJVItY CaraMeL PIe

Modified: presideat minjster judge
pozitjvity caramel pie

['president', 'preside at', 'preside-at', 'preside',
'presidia', 'desiderate', 'desiderata']
['minster', 'minister', 'Mister', 'minter']
['positivity']

Text: president minister judge
positivity caramel pie
```

Figure 10 Error fixing step

VII. Testing Our Program

Some examples of the results obtained from real-life trials are shown below.

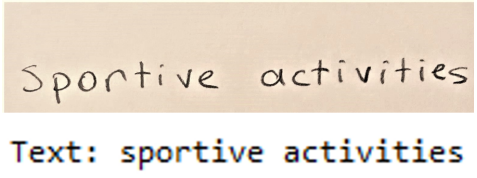


Figure 11 Testing our program

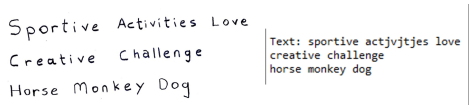


Figure 12 Testing our program

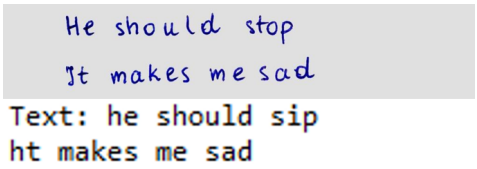


Figure 13 Testing our program

VIII. Conclusion

In this project, it was aimed to create a program that converts handwritten text on paper into computer text. For this purpose, projects created for such purposes were researched, their approaches were examined and lessons were learned. New perspectives on this field have been gained in every research carried out in the name of solving the big or small problems that occurred while building this structure that started from scratch.

In program, it is provided to make the image received from the user suitable for reading. Then, the method of separating the drawings in this image one by one and putting them into the desired format is planned. In the meantime, the search for the neural network model that is suitable for the purpose of the project and promising the best results and the data set that will train these neural networks in the best way has been attempted. Lessons were learned from every step of this search and the most correct combination was chosen. Then, it was ensured that the data taken from the photograph was transferred correctly to the neural networks, and then the results were combined to create a text. In order to get even better results in this working system, each module has been developed and every step of this development work has been recorded. The product obtained at this point of the project has reached the target set at the beginning of the project and has become a program that can fulfill the desired task.

The program we have is able to successfully separate an image (which provides the conditions and text) into its drawings and classify these drawings with a neural network architecture with 89.4% success rate. Our program is also designed to perform accuracy checks before outputting to the user. It has been observed that more than half of the mistakes made by the program can be prevented thanks to this control mechanism.

Although our program meets the requirements, there is no doubt that it has improvable aspects. For example, a more effective letter separation algorithm (provided that it does not cause loss of performance); A training approach that will be reconstructed with a dataset containing punctuation marks to be able to read punctuation marks as well as digits and characters. Also, much more work can be spent in the search for a "more consistent validation loss curve" and a more preferable situation can be obtained than we have. In addition to these, more trials and enhancements can always be made to achieve the best for the neural network architecture used in the program and the training parameters of this neural network, without the limit called the best.

References

[1] M. He, S. Zhang, H. Mao, and L. Jin, "Recognition confidence analysis of handwritten chinese character with cnn," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2015, pp. 61–65.

[2] K. Horii, "Facial image processing method and facial image processing apparatus," Dec. 15 1998, uS Patent 5,850,463.

[3] A. Sander and M. Wolfgang, "The rise of robotics," *Bcg perspectives*, 2014.

[4] C. Clifford, "Elon musk: Robots will take your jobs, government will have to pay your wage," *CNBC. com*, vol. 4, 2016.

[5] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik, "Emnist: Extending mnist to handwritten letters," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2921–2926.