MINISTRY OF EDUCATION AND SCIENCE OF

THE KYRGYZ REPUBLIC

ALA-TOO INTERNATIONAL UNIVERSITY

ENGINEERING AND INFORMATICS FACULTY

COMPUTER SCIENCE DEPARTMENT

RECOGNITION OF THE HANDWRITTEN TEXT IN KYRGYZ LANGUAGE

DIPLOMA THESIS

By Fatima Bekmamatova

Thesis Supervisor: M.Sc. Azamat Kibekbaev

Bishkek – May 30, 2023

MINISTRY OF EDUCATION AND SCIENCE OF THE KYRGYZ REPUBLIC

ALA-TOO INTERNATIONAL UNIVERSITY

ENGINEERING AND INFORMATICS FACULTY

COMPUTER SCIENCE DEPARTMENT

RECOGNITION OF THE HANDWRITTEN TEXT IN KYRGYZ LANGUAGE

DIPLOMA THESIS

By Fatima Bekmamatova

Thesis Supervisor: M.Sc. Azamat Kibekbaev	**Date:**
Head of Department: Dr. Ruslan Isaev	**Date:**

Bishkek – May 30, 2023

УДК: 004

RECOGNITION OF THE HANDWRITTEN TEXT IN KYRGYZ LANGUAGE

Fatima Bekmamatova
Student of the Computer Science Department of Engineering and Informatics Faculty at Ala-Too International University, Kyrgyzstan, Bishkek.
E-mail: fatima.bekmamatova@iaau.edu.kg / fattijenishbek01@gmail.com
Azamat Kibekbaev
Senior Data Scientist, Lecturer in Ala-Too International University, Team Lead of Data Science Department in MBank
E-mail: kibekbaev_a@auca.kg

Abstract

The purpose of this project is to develop and implement an algorithm using Python, TensorFlow, CNN model, and OCR for the recognition of handwritten text in the Kyrgyz language. Although the recognition of handwritten text is available in widely used languages such as English and Russian, it is important to implement it for the Kyrgyz language as well. Despite the existence of a theoretical algorithm for Kyrgyz Language, it has not been put into practice yet. Implementing this algorithm is important for several reasons: it saves time for people when converting handwritten text to digital form, a significant amount of data in the Kyrgyz Republic is still paper-based, digitization is important for preservation, and for writers, writing with a pen is more comfortable and typing everything takes time when it comes to converting, so this saves them time and energy or money. Most of the implemented projects used IAM dataset of handwritten words. We started implementation with a dataset of letters.

Keywords: Algorithm, Handwritten text, TensorFlow, Python, CNN model, OCR, IAM dataset, character dataset.

КЫРГЫЗ ТИЛИНДЕГИ КОЛ ЖАЗМА ТЕКСТТИ ТААНУУ

Кыскача мазмуну

Бул проекттин максаты - кыргыз тилиндеги кол менен жазылган текстти таануу үчүн Python, TensorFlow, CNN моделин жана OCRди колдонуу аркылуу алгоритмди иштеп чыгуу жана ишке ашыруу. Кол менен жазылган текстти таануу англис жана орус сыяктуу кеңири колдонулган тилдерде бар болгону менен, аны кыргыз тили үчүн да ишке ашыруу маанилүү. Кыргыз тилинин теориялык алгоритми болгонуна карабастан, ал азыркыга чейин практикада ишке аша элек. Бул алгоритмди ишке ашыруу бир нече себептерден улам маанилүү: кол менен жазылган текстти санариптик түргө өткөрүүдө адамдардын убактысын үнөмдөйт, Кыргыз Республикасындагы маалыматтардын олуттуу көлөмү дагы эле кагаз түрүндө, аларды санариптик түрдө сактоо маанилүү, ал эми жазуучулар үчүн калем ыңгайлуураак жана конвертациялоодо бардыгын терүү убакытты талап кылат, андыктан алардын убактысын, күчүн же акчасын үнөмдөйт. Ишке ашырылган долбоорлордун көбү колжазма сөздөрдүн IAM маалымат топтомун колдонушкан. Биз ишке ашырууну тамгалардын маалымат топтому менен баштадык.

Ачкыч сөздөр: Алгоритм, кол жазма текст, TensorFlow, Python, CNN модели, OCR, IAM маалымат топтому, символдордун маалымат топтому.

РАСПОЗНАВАНИЕ РУКОПИСНОГО ТЕКСТА НА КЫРГЫЗСКОМ ЯЗЫКЕ

Аннотация

Целью данного проекта является разработка и внедрение алгоритма с использованием Python, TensorFlow, модели CNN и OCR для распознавания рукописного текста на кыргызском языке. Хотя распознавание рукописного текста доступно на широко используемых языках, таких как английский и русский, важно внедрить его и для кыргызского языка. Несмотря на существование теоретического алгоритма для кыргызского языка, он еще не реализован на практике. Реализация этого алгоритма важна по нескольким причинам: это экономит время людей при переводе рукописного текста в цифровой вид, значительный объем данных в КР до сих пор находится на бумажных носителях, оцифровка важна для сохранности, а для писателей – писать с ручка более удобна, и ввод всего требует времени, когда дело доходит до конвертации, так что это экономит их время, энергию или деньги. Большинство реализованных проектов использовали набор рукописных слов IAM. Мы начали нашу реализацию с набора данных символов.

Ключевые слова: Алгоритм, Рукописный текст, TensorFlow, Python, модель CNN, OCR, набор данных IAM, набор символов.

INTRODUCTION

Along with the improvement of IT technologies, the digitization of content and data collection has also increased in the world. Many people are converting paper-based text content, handwritten or from books to digital. And it is one of the most time-wasting processes if someone will sit and convert each word by typing as people were doing before. However, people have already developed algorithms and applications that recognize handwritten or typed texts with further automatic conversion to the desired format of the document. From the beginning of text recognition, most scientists and developers were focused on texts in English [1]. It is because of globalization and they wanted to serve more people in the whole world. Later many people started developing text recognition apps with their own languages [1]. For now, along with many popular languages in the world, recognition of handwritten text in English or Russian is available in many apps and we can also use them. However, we don't have such a program or algorithm with Kyrgyz language yet. Because, even though our letters are similar to Russian, we have extra three letters which are not in Russian, i.e: Ө, Ү, Ң. So, the purpose of this project is to realize an algorithm for the recognition of handwritten text in Kyrgyz language and digitize it.

There are several options for implementing handwritten text recognition, including [11]:

- Optical Character Recognition (OCR): This involves using image processing techniques to recognize individual characters in an image of handwritten text.
- Deep Learning-based Approaches: This involves using deep neural networks, such as Convolutional Neural Networks (CNNs), to recognize handwritten text. These models are trained on large datasets of handwritten

text and can accurately recognize text even if it is written in different styles or with varying levels of quality.

- Hidden Markov Models (HMMs): This method is used to recognize sequences of characters in handwritten text. HMMs are often used for handwriting recognition in conjunction with other techniques, such as dynamic programming, to increase accuracy.
- Hybrid Approaches: This involves combining different recognition techniques, such as OCR and deep learning, to achieve higher accuracy.

For the recognition of the handwritten text in Kyrgyz Language, there was given an algorithm idea of using HMM (Hidden Markov Models). This time, we will try to implement the project by using provided algorithms, i.e using OCR, Deep Learning methods and demonstrate the results with accuracy.

BACKGROUND AND LITERATURE REVIEW

Text recognition started with *OCR* [2]. And from time to time recognition of text, handwritten text still continues to improve using deep learning, and big data, so that it becomes available in different languages.

1.1. Recognition of handwritten text in other languages with Cyrillic letters

According to my research, the recognition of handwritten text containing Cyrillic characters has been accomplished in both Russian [2] and Kazakh [4] languages. Also, there was written a dissertation about algorithms for the recognition of handwritten letters including Kyrgyz language [5].

1.1.1. Recognition of handwritten text in Russian and Kazakh languages

To achieve accurate recognition of handwritten text, a significant amount of words as IAM dataset is required [4]. The students of Satbaev University created the HKR For Handwritten Kazakh & Russian Database [4], which consists of around 63,000 sentences and over 715,699 symbols. This comprehensive dataset encompasses contributions from approximately 200 distinct writers. The authors of the dataset utilized this collection to elucidate their approach to recognition.

1.1.2. Recognition of handwritten text in Kyrgyz language

In case we have only the dataset as letters and not a collection of words, HMM approach help us a lot. This algorithmic approach allows for segmenting the text into letters and comparing them to the letters within the dataset. By doing so, it accurately gathers letters to form words and presents them as the output. This method was explained in the dissertation [5] authored by a Ph.D. student who searched and developed an algorithm for recognizing handwritten text in the Kyrgyz language.

1.2. Approaches for the Recognition of handwritten text

Based on my research findings, there are two initial approaches for recognizing handwritten text. The first approach, as described in [6], involves

segmenting the text into words and applying an algorithm specifically designed for word recognition. For this method, a dataset such as IAM containing words is required. The second approach, mentioned in [7], involves segmenting the text into words and further segmenting those words into individual letters. Then, an algorithm is applied to recognize the letters, which are subsequently combined to form words, providing the final output. For this method, a dataset containing only letters is necessary.

Hypothesis

Since I have only letters as a dataset, it's better to start with segmenting words into letters and apply an algorithm to the recognition process. As an output give a word gathered from letters. There might be a very low accuracy of the result since segmenting cursively handwritten words is complicated. On the other hand, if the word is written with even a small space between each letter, the chance of getting a good result is high.

APPROACH

Our approach is to try as many as possible options of getting the high accuracy result for recognition. For example, Pytesseract, DocTR (document text recognition), models like AlexNet or SVM, and of course our own model. Since segmenting word into letters might be complicated, we will try both cases where the word is written fully cursively without any space between letters, and where words are written with a small space between letters. After getting predicted letters we show the result by concatenating letters.

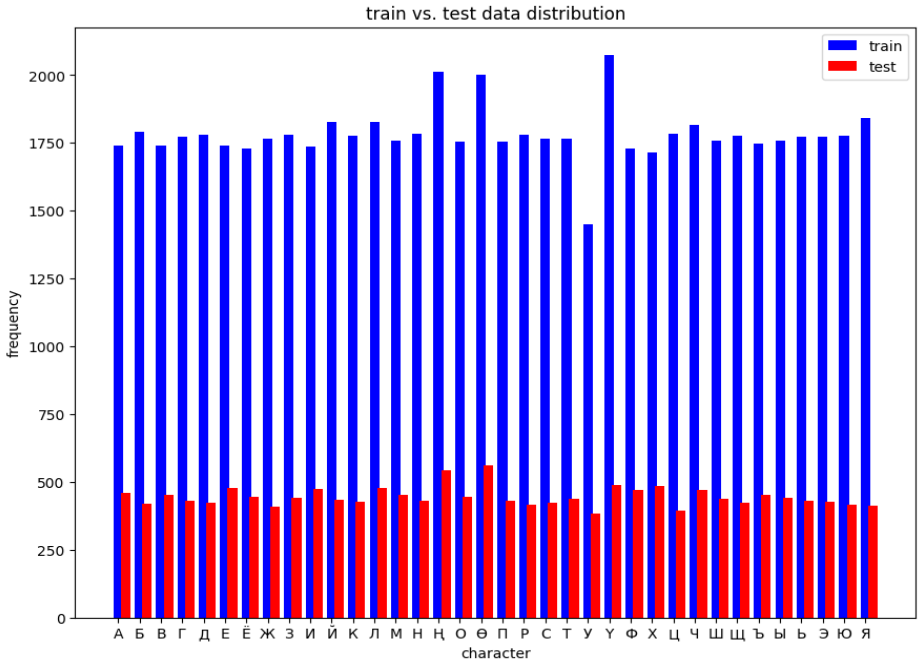
IMPLEMENTATION

First of all, for implementing this algorithm, we need to have a dataset collected of Kyrgyz letters. So, we have 89117 pictures collected, which is divided into train = 80213, and test = 8904 pictures each. We have exactly 36 uppercase letters dataset.

Data Preprocessing:

- We should load the data.csv file where our all characters are saved. We do it with the help of read_csv function from the pandas library.
- Nex step is, we should separate the features (images) and labels (characters) from the DataFrame
- Following step, split the dataset into training and testing sets using the train_test_split function from sklearn. model_selection library.
- Verify the shape of the training and testing sets.
- Save these sets and later use them for training and testing the model.

  Below we can see the result after preprocessing our data and splitting them (picture 1)



Picture 1. characters divided into train and test

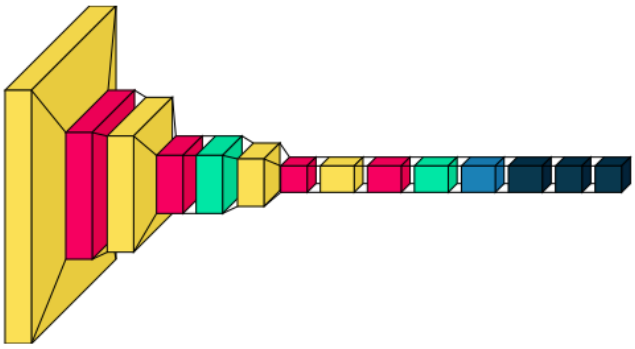
CNN Architecture:

We created neural network model with kreas.Sequential() and trained it with 5 epochs.

- Conv2D layer: This layer performs 2D convolution on the input data. It applies a specified number of filters to the input, each filter learning different features [10].
- MaxPooling2D layer: This layer performs downsampling by taking the maximum value within each pool of pixels. It helps in reducing the spatial dimensions of the feature maps, reducing the number of parameters and computation needed, while retaining the most important features [10].
- Flatten layer: This layer reshapes the input into a 1-dimensional vector, which is required before feeding it into a fully connected layer. It collapses the multi-dimensional input into a single dimension [10].
- Dense layer: This layer is a fully connected layer in which every neuron is connected to every neuron in the previous and next layers. This is the final layer of the model, representing the output layer [10].

This is the visual (Picture 2) of our model and the result (Picture 3):

```
[6]  vk.layered_view(model)
```



Picture 2. Model layers visualization

```
Total params: 1,344,676
Trainable params: 1,344,676
Non-trainable params: 0
```

Picture 3. Model summary

After training our model we got a trained model with the following accuracy (Picture 4):

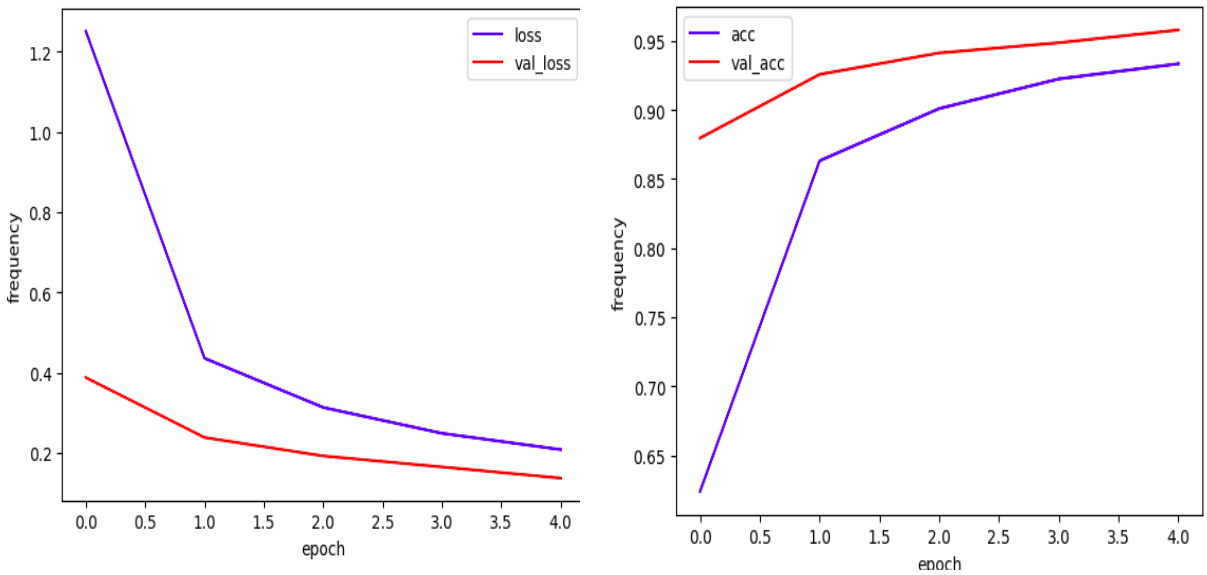
```
Epoch 5/5
535/535 [==============================] - 364s 680ms/step

loss: 0.2083 - accuracy: 0.9335

val_loss: 0.1372 - val_accuracy: 0.9579
```

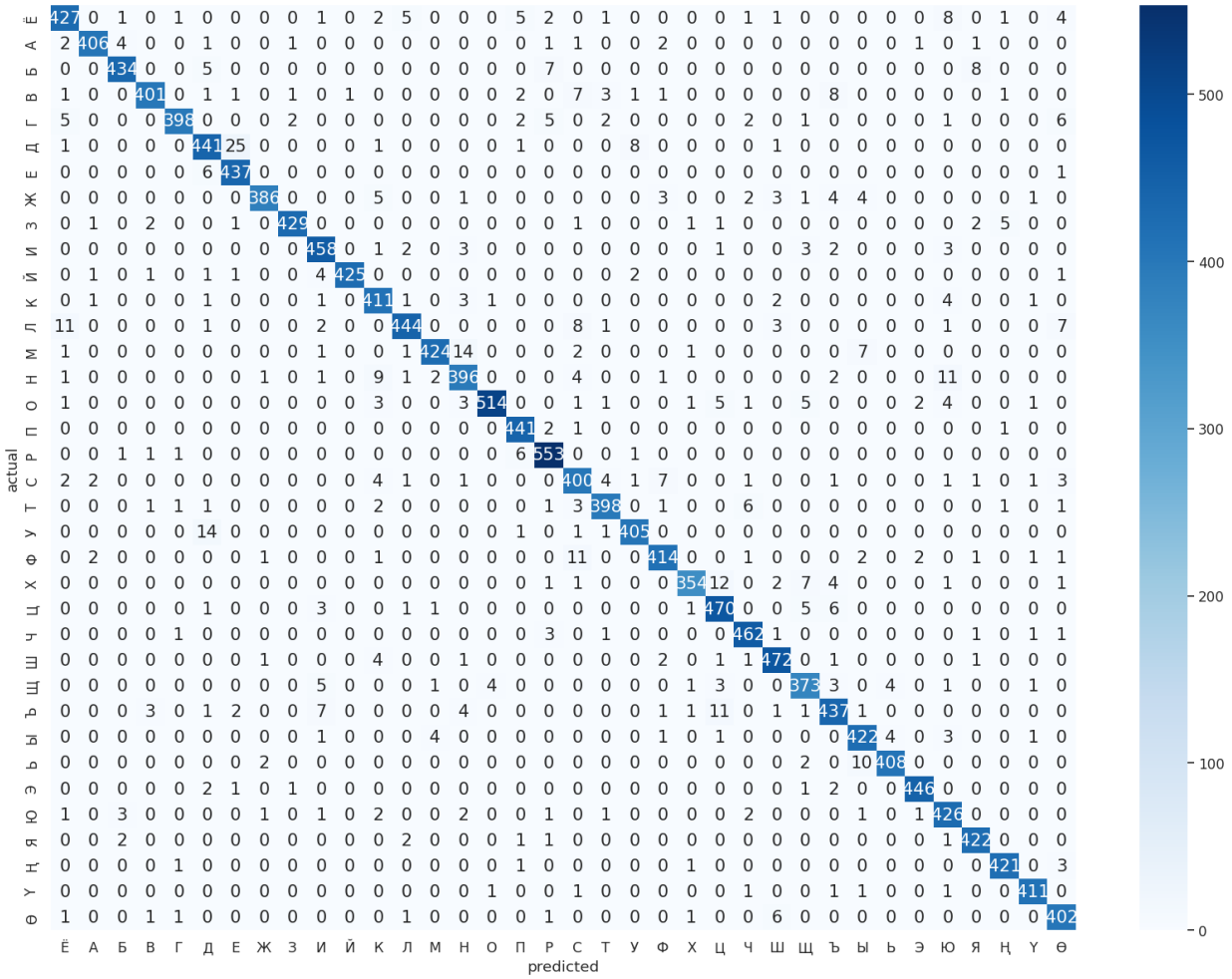
Picture 4. Trained model result

Here we can see the graphs of how loss is decreasing and accuracy is increasing (Picture 5):



Picture 5. Accuracy and loss graph

Accuracy match is shown in the graph below (Picture 6):



Picture 6. Accuracy match graph

Project:

In order to divide words into letters we used the threshold methods OpenCV threshold and Otsu's threshold.

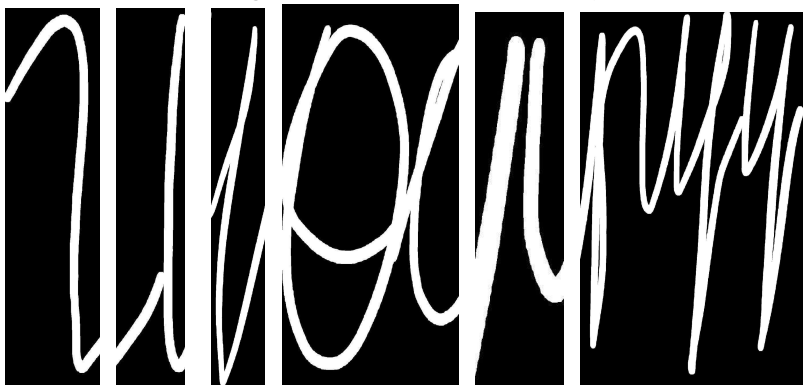
Otsu's thresholding method helps to automatically determine an optimal threshold value for image segmentation. It tries to find a threshold that maximizes the inter-class variance of the grayscale values in the image [8].

cv.treshold is the thresholding method that converts a grayscale image into a binary one by assigning a threshold value that separates the pixels into two categories based on their intensity values [9]. For example: If the pixel value is greater than the threshold, it is set to a maximum value, otherwise, it is set to 0.

1. **Word written without spaces (**Picture 7)

Picture 7. Input image sample 1

In order to recognize the letters of words we wrote a program with divides word into characters. After the implementation of the program, we got the following images of letters (Picture 8):

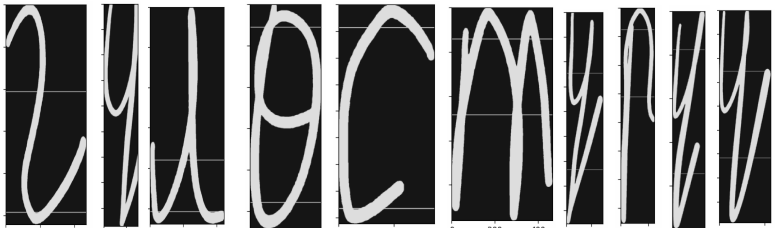


Picture 8. Word segment result 1

2. **Word written with small spaces.**

Picture 9. Input image sample 2

Then, we tried to apply our program which divides word into letters with input images that written with spaces. After the implementation of the program, we got the following images of letters (Pictture 10):



Picture 10. Word segment result 2

After the application of the model to each letter and concatenating them as one word, we got the following result.

RESULTS

Based on the observations mentioned earlier, the initial approach of recognizing words written without spaces gave us extremely poor accuracy. It even failed to correctly identify the number of letters in the word.

However, when we employed the second approach, which involved words written with spaces between letters, we achieved a significantly higher accuracy of 95% based on our model. To assess the performance of our program and model, we tested it using an uploaded image that contained a word written in uppercase Kyrgyz letters (as shown in the Picture 11).

Picture 11. Image for testing

As the result we got (Picture 12):

```
КТЛҮҮ
[(11, 36), (21, 36), (12, 36), (23, 36), (23, 36)]
```

Picture 12. Recognition result

CONCLUSION

Our goal was to make research and on base that to implement the project. So far we tried with the letter dataset two kinds of approaches. As a result, our approach can be used for handwritten texts which are written with space, not fully cursively. So, with this result, we can move not so far forward in Handwritten Text

Recognition in Kyrgyz Language. In other words, for high accuracy and full implementation, using only letters is not enough. So, there is work left to collect handwritten cursive words which is in the process now.

As it was said that Deep Learning and a large amount of training data would dominate in soon future [1], we are also in the process of collecting large data of handwritten kyrgyz words.

FUTURE WORK

In light of the challenges encountered during the current project, it has become evident that the recognition of handwritten text using only individual letters poses significant difficulties. The segmentation of cursive words, in particular, proves to be a complex task, requiring further exploration. Therefore, for the future work, our focus will shift towards the collection of a comprehensive dataset consisting of handwritten words.

To address this, we will initiate the collection of handwritten words written in the Kyrgyz language. While we have started by providing self-written examples, we recognize the importance of obtaining a diverse and representative dataset. To accomplish this, we plan to collaborate with schools and engage students in the process of writing words that reflect the natural variations encountered in everyday handwriting. This approach will not only contribute to the expansion of the dataset but also foster valuable engagement with the local community.

In the event of a successful outcome, our vision extends beyond the completion of the project. We plan to collaborate with friends and experts in the fields of web development and application design.

REFERENCES

1. Y. Zhu, C. Yao, and X. Bai, “Scene text detection and recognition: recent advances and future trends,” Frontiers of Computer Science, vol. 10, no. 1. Higher Education Press, pp. 19-36 doi: 10.1007/s11704-015-4488-0. [Electronic resource]. – Access mode: https://xbai.vlrlab.net/files/FCS_TextSurvey_2015.pdf (Retrieved: 01.02.2016).

2. The History of Optical Character Recognition: [Electronic resource]. – Access mode: https://gorillapdf.com/blog/the-history-of-optical-character-recognition/ (Retrieved: 19.02.2022).

3. Abdelrahman A., Mohamed H., Daniyar N., 2020, Attention-based Fully Gated CNN-BGRU for Russian Handwritten Text, National Open Research Laboratory for Information and Space Technologies at Satbayev University, Almaty, Kazakhstan. [Electronic resource]. – Access mode: https://arxiv.org/pdf/2008.05373.pdf (Retrieved: 20.08.2020)

4. Nurseitov D., Bostanbekov K., 2020, Handwritten Kazakh and Russian (HKR) database for text recognition, Satbayev University Almaty, Kazakhstan. [Electronic resource]. – Access mode: https://www.researchgate.net/publication/342763791_Handwritten_Kazakh_and_Russian_HKR_database_for_text_recognition (Retrieved: 01.07.2020)

5. Kudakeeva G., (2020), DEVELOPMENT OF ALGORITHMS FOR RECOGNITION OF VISUAL IMAGES, Kyrgyz State Technical University, Bishkek, Kyrgyzstan. [Electronic resource]. – Access mode: link (Retrieved: 2020).

6. R. Manmatha and N.Srimal, Scale Space Technique for Word Segmentation in Handwritten Documents, University of Masachusetts, Amherst MA 01003, USA. [Electronic resource]. – Access mode: http://ciir.cs.umass.edu/pubfile/smm-27.pdf (Retrieved: 2010).

7. Dhaval Salvi, Jun Zhou, Jarrell Waggoner, and Song Wang, Handwritten Text Segmentation using Average Longest Path Algorithm, Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208, USA. [Electronic resource]. – Access mode: https://www.cse.sc.edu/~songwang/document/wacv13c.pdf (Retrieved: 2012).

8. An Introduction to Image Segmentation: Deep Learning vs. Traditional [+Examples] [Electronic resource]. – Access mode: https://www.v7labs.com/blog/image-segmentation-guide (Retrieved: 12.08.2021).

9. OpenCV documentation: [Electronic resource]. – Access mode: https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html (Retrieved: 29.12.2022).

10. CNN model layers explained: [Electronic resource]. – Access mode: https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939 (Retrieved: 27.08.2020).

11. Handwriting Recognition: Definition, Techniques & Uses. [Electronic resource]. – Access mode: https://www.v7labs.com/blog/handwriting-recognition-guide (Retrieved: 02.12.2022).