Paper:

Toward Construction of Handwritten Indus Signs Dataset

Sujata Saini[*1], **Hiroki Shibata**[*2], **and Yasufumi Takama**[*3]

[*123]School of System Design, Tokyo Metropolitan University
6-6 Asahigaoka, Hino, Tokyo,
191-0065, Japan
Email: [*1]saini-sujata@ed.tmu.ac.jp, [*2]hshibata@tmu.ac.jp, [*3]ytakama@tmu.ac.jp

[Received 00/00/00; accepted 00/00/00]

Abstract. Indus script is a writing system developed in the Indus civilization. Despite so many attempts over the years, it is still undeciphered. The decipherment of the Indus sign has yet to be resolved due to a lack of knowledge and limited data available. The demand for the large volume of datasets has increased recently with the hope of applying the state-of-the-art machine learning algorithms. The aim of the paper is to construct the dataset of the top 10 frequently appeared Indus signs found in multiple sites in the Indus civilization. As the number of images of actual Indus signs available is limited, a dataset is constructed by asking people to draw the Indus signs referring to the actual images. The handwritten images of Indus signs are collected from 36 participants with a web application. To examine the quality and characteristics of the constructed dataset from the viewpoint of machine learning, the dataset is used to train Convolutional Neural Networks (CNN) with the Tensorflow. The result shows the obtained model achieved 93% classification accuracy.

Keywords: Indus sign dataset, image classification, dataset creation

1. Introduction

A writing system that was developed in the Indus civilization (2600-1700 BCE) is called the Indus script, which is mostly still undeciphered despite several attempts over the last 80 years [1,3]. About 4000 inscriptions have been discovered so far over numerous artifacts (i.e., seals, tablets, pottery, stoneware, copper plates, bangles, tags, miniature tablets, weapons, tools, wood, etc.) from various sites of the Indus civilization [4]. The decipherment of the Indus script is still unresolved due to limited knowledge of the spoken language, the absence of bilingual texts, and the small size of the corpus [2].

An Interactive Concordance of Indus Text (ICIT) is a web-interface-based database that has been developed to collect digitalized images of various seals and signs of inscriptions. There are 709 different signs in the sign list of the ICIT database, and each sign has different variants. The current version 2.7 of the ICIT database contains 17942 sign images in the corpus [5]. One of the potential applications of the collected images toward the decipherment is to train a classification model for recognizing the Indus script in the images of inscriptions. However, in the current situation, due to the limited number of images of each sign, it's challenging to perform such a computational approach.

Over the last few decades, the demand for a dataset that is large enough to train classification models has increased among data scientists and machine learning researchers. However, there are two challenges to construct such datasets for Indus scripts: there are few experts who can correctly label the images of the Indus script, and the number of images containing actual Indus scripts is too small. That is, the historical language dataset is rare, and limited to books, website, and other resources.

The objective of this paper is to construct the artificial handwritten image dataset of the top 10 frequently appeared Indus signs. This approach can collect many handwritten images from a single image of an actual Indus script. To collect Indus signs data, an interactive, user-friendly web application is developed: with using it users can draw the Indus sign's image on canvas with a Mouse and a Pen Tablet. Furthermore, it is expected that different people draw the same script differently, which would cause the variation of data that is important for training robust models.

It is important to examine the quality and the characteristics of the constructed dataset from the viewpoint of applying machine learning. This paper examines the classification accuracy of models trained from the constructed datasets. Experimental results show that the Convolutional Neural Network (CNN) model with Tensorflow achieved 93% classification accuracy.

2. Related Work

Several studies have been conducted to apply computational approaches to ancient scripts, including the Indus script. Using a simple Markov chain model [3] to capture sequential dependencies between signs in the Indus script, generate new sample texts, and reveal recurring patterns of signs of the Indus seal was proposed. The results indi-

cated the syntactic structure of the Indus script could be identified, and it was possible to recognize the damaged objects, missing, ambiguous or unreadable characters [4]. Indus texts in Indus script represent the linguistic writing system than a nonlinguistic symbol system. A statistical approach, especially n-gram Markov chains, was used to analyze Indus script syntax, which achieved 75% accuracy [2]. A soft computing technique that focuses on hybrid feature extraction and convolutional neural networks was proposed to recognize the eleventh-century ancient Tamil character and convert them into current-century words; it achieved 92.78% accuracy [9].

Various studies have been conducted to apply computational approaches to handwritten images dataset. A deep learning approach has demonstrated the potential in computational epigraphy and digital humanities: it achieved 92% accuracy in identifying the presence and absence of jar signs in the Indus script [6]. A high-level morphological comparison with the MNIST dataset using oft-used CNN architecture has shown that such a simple method achieved 96.8% accuracy for the Kannada MNIST test-set and 76.1% for the Dig-MNIST test-set. These datasets provide a sterner challenge in terms of generalizability than MNIST or the KMNIST datasets [7]. A dataset that focuses on Kuzushiji (cursive Japanese), known as Kuzushiji-MNIST, introduced two more significant and more challenging datasets: Kuzushiji-49 and Kuzushiji-Kanji. The classification results with a simple 4-Nearest Neighbor (4-NN) model achieved 91.56% on Kuzushiji-MNIST and 86.01% on Kuzushiji-49 datasets [8].

3. Methodology

Figure 1 shows the architecture of the proposed pipeline, which contains the data augmentation by collecting handwritten images, preprocessing, training models, and evaluation. This paper focuses on data collection and preprocessing.

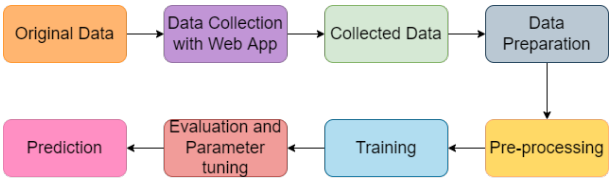


Fig. 1. The architecture of the proposed pipeline.

Table 1 shows the relationship between sign sequence and texts present in the Indus seal. The sign sequence is the list of different Indus signs in a linear order and should always be read from right to left using standardized graphs (even if the original reading direction differs from the right to left).

Targets to be collected are the top 10 most frequently appeared Indus signs in the various sites of Indus civilization. The reason why collecting only the top 10 sign images is because those signs were used much more frequently than others: the data can be useful for training a model to recognize Indus texts presented in various Indus seals. There are 709 Indus signs present in the sign list of the ICIT database [5]. The frequency of the Indus signs in ICIT database is constantly updated based on newly found scripts.

Table 1. Sign sequence in Indus Seal [5].

ID	Image	Text
1082	 [H-22a]	+740-920-320-233-806-002-405-321+

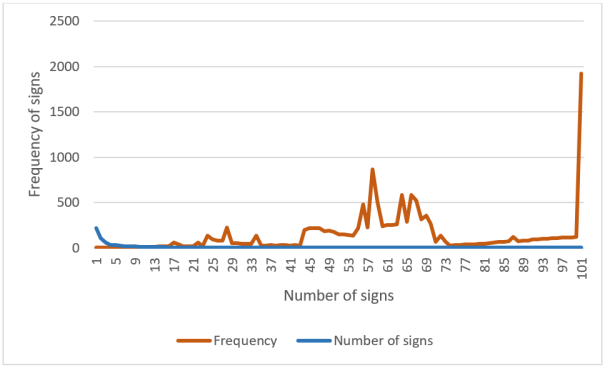


Fig. 2. The distribution of Indus signs as per the frequency of signs and number of signs [5].

Figure 2 shows the distribution of the frequency of Indus signs that appeared in the inscriptions found on different historical sites of the Indus civilization, which was reported in 2022 [5]. As shown in the Figure, there are many signs that appeared 10 or less times, and there are only 7 signs that appeared more than 500 times. S740 appeared 1920 times, which is the highest frequency. There is a total of 19831 signs in the ICIT database, among which 17957 are legible signs.

3.1. Dataset Collection

The original images to be used for Indus signs dataset construction are obtained from ICIT Database [5]. **Table 2** shows the used original image together with its sign id, the number of variants, and the total frequency reported in 2022 [5]. In all different variants of the top 10 Indus signs, only high-ranking variants that appeared on multiple scripts has chosen. The reason for not using the data augmentation technique for data collection is that there are not sufficient data. In the future, it is also possible to extend the constructed dataset by applying the data augmentation technique.

Table 2. Top 10 frequently appeared Indus signs in various sites of the Indus civilization as per ICIT Database [5].

Sign ID	Image	Variants	Total frequency in all sites
S740		6	1920
S002		1	865
S220		1	477
S032		1	584
S400		6	501
S033		1	522
S240		1	352
S390		2	285
S520		1	316
S820		2	240

3.2. Application for Collecting Data

The handwritten Indus signs data are collected with a crowdsourcing method by developing a web application named IndusDraw [10]. **Figure 3** shows the screenshot of the IndusDraw. The process for collecting data via a web app is as follows:

1) Visit the homepage of the web application (https://indusdraw.tech/)

2) See the actual Indus sign image on the left side and draw the same sign on the canvas located on the right-hand side.

3) After drawing the sign, click on the download button to download the 300x300 pixel PNG image in greyscale format. Arbitrary filenames are given by participants.

4) To draw another sign image, click on the clear button to reset the canvas.

5) To change the Indus sign to be drawn, click on the previous and next buttons below.

6) Organize a set of 10 images so that it contains one image for each sign.

7) Send those images to a designating email address with the email subject "Indus_Signs, [participant's gender]_[participant's age]". More than one set can be sent at a time. We employed this approach so that users can explicitly select and send images to be included in the dataset.

3.3. Data Preparation

Collected data is organized using the sign_id as the label of sign images. The labels are manually given to images by one of the authors. Each sign image has more than one stroke and an average of 3 strokes per image. To load the Indus sign data in an organized format, the dataset has 10 folders, each of which contains the images of different Indus signs. **Figure 4** shows an example of a collected image for ten signs, each of which was written by different participants.

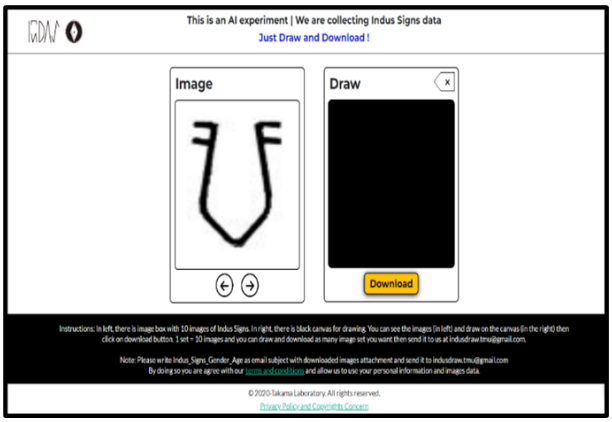


Fig. 3. Web application used for collecting the data.

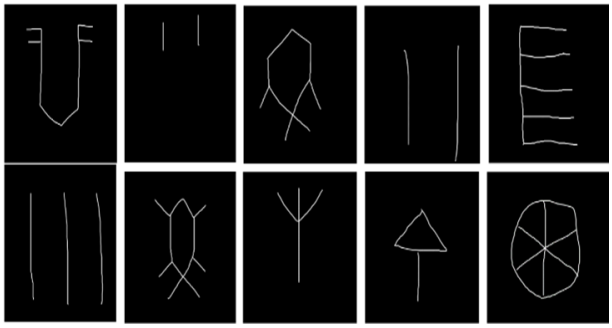


Fig. 4. Examples of collected data from the participants (1 set = 10 images).

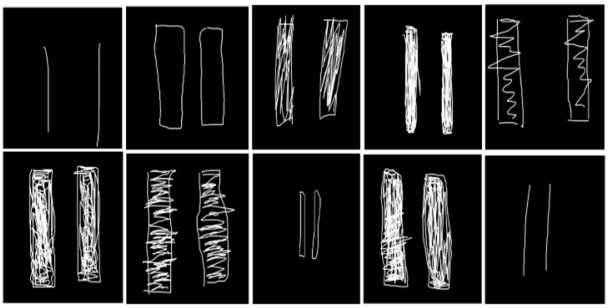


Fig. 5. A representation of sample images drawn by different participants for the same sign.

Figure 5 shows the images drawn by different participants for the same sign (sign_id: S032). As shown in these figures, the dataset contains various images for the same Indus signs. The images were collected during the month of March-April in 2022. We selected images written by the participants aged from 18 to 60 because of their cognitive abilities to perform in the experiment. As a result, the constructed dataset contains 1000 images, 100 images for each sign.

This paper constructed three types of datasets as shown in **Table 3**: All Data consists of images drawn with a mouse or a pen tablet, Mouse consists of images drawn with a mouse only, and Pen Tablet consists of those with

a pen tablet only.

Table 3. Indus sign dataset distribution.

Indus Sign dataset	Total Data	Training	Testing
All Data	1000	800	200
Mouse	580	464	116
Pen Tablet	420	336	84

Table 4. Participants categorized as per their age.

Age	Number of Participants
18-20	1
21-30	25
31-40	9
41-50	0
51-60	1

Table 4 shows the distribution of participants' age. It is shown that 25 out of 36 are young within the 21-30 age group, which occupies about 70% of the participants. In **Table 5**, participants were categorized by gender, in which the majority of the participants are male.

Table 5. Participants categorized as per their gender.

Male	Female	Others
27	8	1

4. EXAMINATION OF DATASET

4.1. Experimental Setup

The purpose of this experiment is to examine the quality and characteristics of the constructive dataset from the viewpoint of machine learning approach. This paper employs the Convolutional Neural Network (CNN) model with Tensorflow [11].

Experiment was conducted on a PC: Windows 64-bit operating system with 8GB RAM, AMD Ryzen 5 4600H with Radeon Graphics 3GHz configuration. Training and Test of a CNN were conducted in the Google Colab.

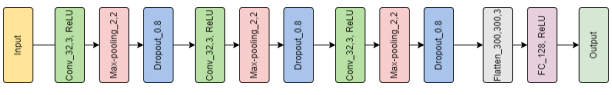


Fig. 6. The Architecture of the Model.

Figure 6 shows the CNN composition of the model. We tested the hyper-parameters settings such as epochs, batch size, and neural network layers, and the result when the highest accuracy was obtained is reported in the next subsection.

4.2. Evaluation

Table 6 shows the classification accuracy results of the CNN model with Tensorflow on the Indus sign datasets with various epochs: 10, 30, 50, and 100. All Data and Pen Tablet datasets achieved 90% or more accuracy even with 10 epochs. The accuracy with 100 epochs is lower than that with fewer epochs, which is caused by the influence of overfitting even though batch size=32 and dropout=0.8 were set for the model training to reduce the overfitting.

Table 6. Classification accuracy of CNN model with Tensorflow at various epochs.

Epochs	All Data	Mouse	Pen Tablet
10	91%	87%	91%
30	91%	85%	92%
50	**93%**	**88%**	**94%**
100	89%	86%	92%

All Data contains images drawn with a mouse and a pen tablet. The accuracy of All Data was similar to that of the Pen Table dataset. On the other hand, the accuracy on Mouse dataset was lower than the other two datasets with all epochs. The reason why the accuracy of the Mouse dataset was lower than others is because the characteristics and quality of the images drawn with a mouse were lower than that with a pen tablet.

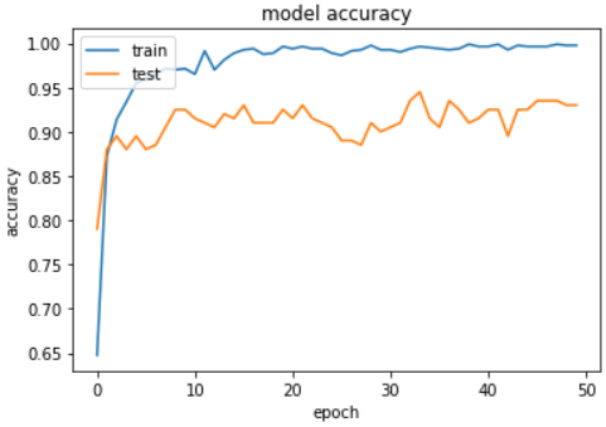


Fig. 7. Accuracy rate of CNN model with Tensorflow with All Data.

Figure 7 and **Figure 8** respectively show how accuracy and loss change with epoch when using the All Data dataset. It was observed in **Fig. 7** that accuracy became stable as the epoch progresses in training phase. In the testing phase, accuracy increased with 10 epochs, then slightly decreased with around 30 and 40 epochs, and finally improved with 50 epochs. The overall accuracy with different epochs is above 87%.

A similar trend was observed with loss in **Fig. 8**. The loss became stable in the training phase with the progress

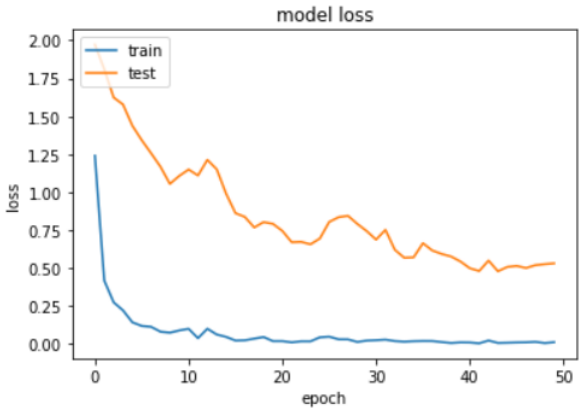


Fig. 8. Loss of CNN model with Tensorflow with All Data.

of training. The decrease in the testing loss was slower than the training loss at the beginning. The minimum loss was obtained between 40 to 50 epochs.

5. CONCLUSION

This paper proposed a method to construct the Indus sign dataset and examined the characteristics of the dataset with machine learning. The proposed approach collected handwritten images of the top-10 frequently appeared Indus signs, which were written by the participants with a mouse or a pen tablet via a web application. To examine the quality of the constructed dataset, the Convolutional Neural Network (CNN) Model with Tensorflow was trained with three datasets: All Data, Mouse, and Pen Tablet datasets. The results show that it achieved 93% accuracy for All Data, 88% for Mouse, and 94% for Pen Tablet. The influence of drawing devices on the quality of the collected images was also examined: it was found that the classification accuracy when using images drawn by a mouse was lower than when using images drawn by pen tablets. As a mouse is a more common device than a pen tablet, how to improve the quality of images drawn by a mouse is one of future works. It is also important to verify that the classification model trained with the constructed dataset can classify actual images of Indus signs correctly. For that purpose, collecting more actual images for testing is also one of our future works.

Acknowledgements

We want to acknowledge the contribution of all those volunteers who participated in the experiment for data collection. We acknowledge the generous guidance of Dr. Andreas Ful for providing access to the ICIT database and Indus sign images.

References:

[1] A. Fuls, "Ancient Writing and Modern Technologies – Structural Analysis of Numerical Indus Inscriptions," Stud. Indus Scr., pp. 57–90, 2020.

[2] N. Yadav, H. Joglekar, R. P. N. Rao, M. N. Vahia, R. Adhikari, and I. Mahadevan, "Statistical Analysis of the Indus Script Using n-Grams," PLoS One, vol. 5, no. 3, 2010, doi: 10.1371/journal.pone.0009506.

[3] R. P. N. Rao, N. Yadav, M. N. Vahia, H. Joglekar, R. Adhikari, and I. Mahadevan, "A Markov Model of the Indus Script," Proc. Natl. Acad. Sci. U. S. A., vol. 106, no. 33, pp. 13685–13690, 2009, doi: 10.1073/pnas.0906237106.

[4] R. P. N. Rao, N. Yadav, M. N. Vahia, H. Joglekar, R. Adhikari, and I. Mahadevan, "Entropy, the Indus Script, and Language: A reply to R. sproat," Comput. Linguist., vol. 36, no. 4, pp. 795–805, 2010, doi: 10.1162/coli_c_00030.

[5] Wells, Bryan, and Fuls, Andreas: Interactive Concordance of Indus Texts (ICIT): An Online Database of Indus Inscriptions and Iconography. Berlin (last update 26.03.2022). URL: https://www.indus.epigraphica.de/ (last access 29.03.2022).

[6] S. Palaniappan and R. Adhikari, "Deep Learning the Indus Script," Feb. 2017, arXiv, Available: http://arxiv.org/abs/1702.00523 (last access 25.04.2022).

[7] V. U. Prabhu, "Kannada-MNIST: A new handwritten digits dataset for the Kannada language," Aug. 2019, CoRR, vol. abs/1908.01242, arXiv, Available: http://arxiv.org/abs/1908.01242 (last access 29.04.2022).

[8] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, and D. Ha, "Deep Learning for Classical Japanese Literature," Dec. 2018, CoRR, vol. abs/1812.01718, arXiv, doi: 10.20676/00000341.

[9] V. Vani and S. R. Ananthalakshmi, "Soft Computing Approaches for Character Credential and Word Prophecy Analysis with Stone Encryptions," Soft Comput., vol. 24, no. 16, pp. 12013–12026, 2020, doi: 10.1007/s00500-019-04643-7.

[10] IndusDraw web application: https://indusdraw.tech/ (last access 13.04.2022).

[11] L. Yu, B. Li, and B. Jiao, "Research and Implementation of CNN Based on TensorFlow," IOP Conf. Ser. Mater. Sci. Eng., vol. 490, no. 4, 2019, doi: 10.1088/1757-899X/490/4/042022.