YOLOv4 for Kuzushiji Recognition With Synthetic Training Data Generated by GAN

Mingyuan LI[1], Xuebin YUE[1] and Lin MENG[2,*]

[1]*Graduate School of Science and Engineering, Ritsumeikan University*
[2]*College of Science and Engineering, Ritsumeikan University*

Abstract
Character recognition of ancient Japanese documents is a significant research topic, we can comprehend Japanese history and culture from them. However, Kuzushiji (classical cursive handwriting characters) is pretty hard to understand for almost all young Japanese. Digitizing these books through character recognition technology can greatly assist us recognize. But the current dataset does not have enough samples for many classes, this makes it difficult to accurately recognize these classes. In this research, we propose a method to extend the dataset by using GAN (Generative Adversarial Network) to generate virtual images. Then we combine the generated images and merge these images with the original dataset. For this experiment, we trained with a decoupled YOLOv4 model. Experimental results show that applying this method to the classes with the number of samples greater than 50 and less than 300 can improve the overall accuracy by 21.65%.

Keywords
YOLOv4, GAN, Kuzushiji, Japanese historical character

1. Introduction

There are nearly 2 million ancient Japanese documents, from which we can learn not only about the ancient Japanese people's food, clothing, transportation but also about Japan's geographical and ecological information in the past. Many people have contributed to tidy these documents [1] [2]. Kuzushiji is the main script in these ancient documents, but Kuzushiji faded out of the Japanese public as the writing style altered in 1,900. Although many relevant workers have done a lot of work to popularize it; only a few experts are able to read Kuzushiji now. Therefore, in order to let the public better understand the contents of these ancient Japanese documents, it becomes very important to use deep learning to recognize Kuzushiji. However, there is a very serious problem: Kuzushiji samples are imbalance. In this experiment, a total of 43 ancient documents are selected as the training dataset, which contained 4,328 classes and more than 1,000,000 characters, but the sample distribution is very uneven, the specific distribution is shown in Figure 1. There are only 1,075 classes that appear more than 50 times, and many characters with only a few or a dozen samples, which are clearly not trainable data.

The 5th International Symposium on Advanced Technologies and Applications in the Internet of Things (ATAIT 2023), August 28-29, 2023, Kusatsu, Japan

[*]Corresponding author.

gr0606ee@ed.ritsumei.ac.jp (M. LI); gr0468xp@ed.ritsumei.ac.jp (X. YUE); menglin@fc.ritsumei.ac.jp (L. MENG)

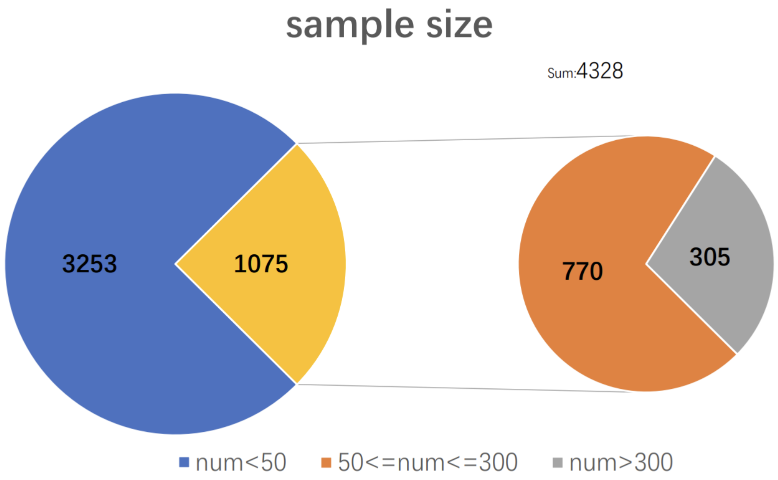


Figure 1: Sample size of the dataset

This paper analyze the data and find some prepositions occur more than 30,000 times, like “の”. Some characters we use a lot, like“赤”, can appear hundreds of times. But some characters, such as “酎”appears less than 5 times in all the data, which we can hardly find in a book. Data distribution is shown in Figure 2. Because of the number of occurrences, we believe it is enough for us to understand ancient Japanese documents by training the classes which sample size greater than 50. Due to some classes having too many data, we set the upper limit of sample size for all classes to

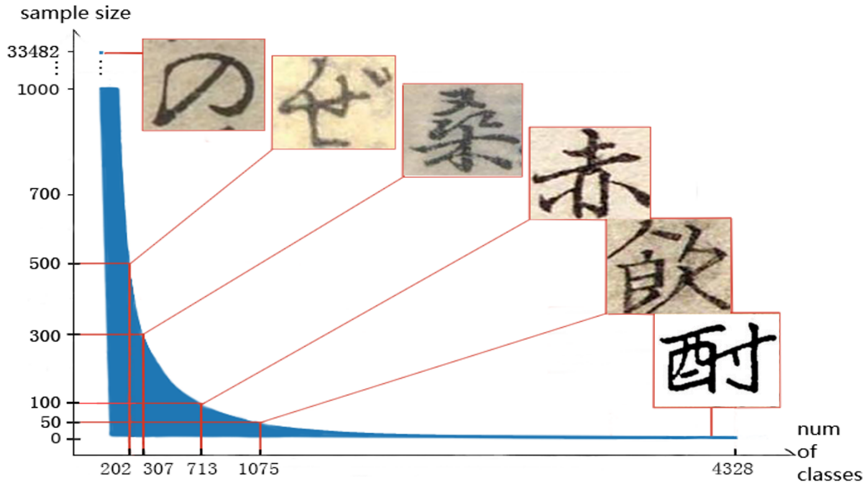


Figure 2: The distribution of dataset

500. For the classes with a sample size greater than 300, because of the enough sample size, we don't do anything to them. However, for the class with a sample size of 50 to 300, the sample size is far from enough to achieve a good recognition effect. If we can expand the sample size of this part, and make it achieve a good training effect, it can provide great help for the reading of ancient Japanese documents.

In order to increase the sample size of these classes, we propose using GAN [3] to expand them. GAN is often used for image generation, and it has a good effect. Therefore, we hope to use GAN to generate virtual images to achieve the purpose of dataset expansion. After expansion, we use the YOLOv4 [4] model to recognize the large database. The experimental results show that our method is better than other expansion methods in the recognition of datasets with a small number of samples. For the classes of sample size 50-300, the individual training accuracy is increased by 17.04% and the training accuracy of all datasets is increased by 21.65%.

Section 2 of this paper reviews related work on GAN and YOLO [5]. Section 3 introduces our methodology, it contains how do GAN and YOLO work, how to expand our database with GAN. Section 4 introduces how do we do the experiment setup and show the result. In section 5 is concluded with a brief summary and mention of future work.

2. Related works

2.1. Kuzushiji recognition

Since 2016, organizations such as the Center for Open Data in the Humanities (CODH) have successively released many datasets on ancient Japanese documents, and many researchers have devoted themselves to the study of ancient Japanese documents.

In 2018, Clanuwat et al. [6] created three datasets for Kuzushiji recognition, namely Kuzushiji-MNist, Kuzushiji-49 and Kuzushiji-Kanji, which provided more data resources for ancient documents recognition. Alex et al. [7] proposed a network called KuroNet which used the residual U-Net architecture to detect and identify full-page Kuzushiji. DILBAG et al. [8] proposed DKNet, which used a filter to improve the visibility of the image, and then used a recognizer based on mobilenet [9] to recognize the character. Aravinda et al. [10] realized the recognition and classification of the Kuzushiji image with high accuracy through image preprocessing, image segmentation and feature extraction. Lyu et al. [11] developed a MobileNetV2-based [12] method using classical deep learning techniques for detecting. Hu et al. [13] realized the recognition of Kuzushiji by an End-to-End method with attention mechanism based on LSTM [14] network. These recognition methods have a good recognition effect on the category with a large number of samples, but can not achieve a good recognition effect if the number of samples is rare.

2.2. GAN augments the dataset

Goodfellow et al. [3] proposed GAN, which generated virtual images by training a generative model and a discriminant model. This method has been widely used to generate training data sets when the number of samples is insufficient.

Bowles et al. [15] proposed the use of GAN to augment the dataset in brain segmentation tasks. DEWI et al. [16] used GAN to augment the national traffic sign dataset. Frid-Adar et al. [17]

proposed a training scheme that first used classical data augmentation to enlarge the training set and then further enlarged the data size and its variety by applying GAN techniques for synthetic data augmentation. Zhou et al. [18] designed a new generator and discriminator of the GAN to generate more discriminant fault samples using a scheme of global optimization. Mariani et al. [19] also encountered the problem of insufficient image, and they also used GAN to generate minority-class images. Yue et al. [20] proposed to use WGAN-GP to expand the dataset of Oracle, and proposed the C-A Net to detect the large dataset; it solved the problem of data imbalance meanwhile improved the accuracy of recognition.

2.3. YOLO for character recognition

In 2016, Redmon et al. [5] proposed the YOLO model, it enabled a single neural network to predict bounding boxes and class probabilities directly from full images in one evaluation. It made it possible to greatly reduce the detection time when there were many samples in an image. This is very suitable for ancient Japanese documents' recognition which contains many characters in a single image. Laroca et al. [21] proposed a robust and efficient system based on the state-of-the-art YOLO object detector. Tang et al. [22] proposed a YOLOv3 [23] based detector to recognize Kuzushiji characters. In 2020 Santoso et al. [24] used YOLO to identify the 8th century AD Kawi character; this character is just like Kuzushiji that only a few people understand. Using YOLO for detection and identification allowed more people to better understand the history and culture at that time.

3. Methodology

3.1. GAN generates dataset

The core of GAN is generator and discriminator, the former is responsible for generating data based on random signals, and the later is responsible for determining the authenticity of the data generated by the generator. During each round of gradient backpropagation, the discriminator is trained first and then the generator is trained. Specifically, supposing the GAN is now trained for the Kth time. The discriminator is first trained and the gennerator is fixed at this time, that is, the parameters of the gennerator are not updated currently. The real image and the virtual image generated in the previous round $(K-1)$ are stitched together and labeled respectively with labels 1 and 0. The stitched image x is input into the discriminator for scoring and score G*(x) is obtained. According to G*(x) and the loss function, the gradient can be backpropagated to update the discriminator parameters. When the generator is fixed, the optimal solution (maximum point) of the discriminator is as follows, the P_{data} means sampling from origin database, and P_g means sampling from generator's output.

$$G^{*}(x) = \frac{P_{data(x)}}{P_{data(x)} + P_{g(x)}} \tag{1}$$

Then the generator is trained, and the discriminator is fixed currently. The generator generates virtual images based on the input random signal $(K-1)$ input discriminator to score D*G(x). The difference in value between D*G(x) and label 1 is backpropagated as a loss function. The loss function updates the parameters of the generator by minimizing the value when $P_{data(x)}$ and $P_{g(x)}$ are closest, that is, the Kth picture is exactly the same as the $(K-1)$th picture, and that is what we want. The process is shown in the Figure 3:

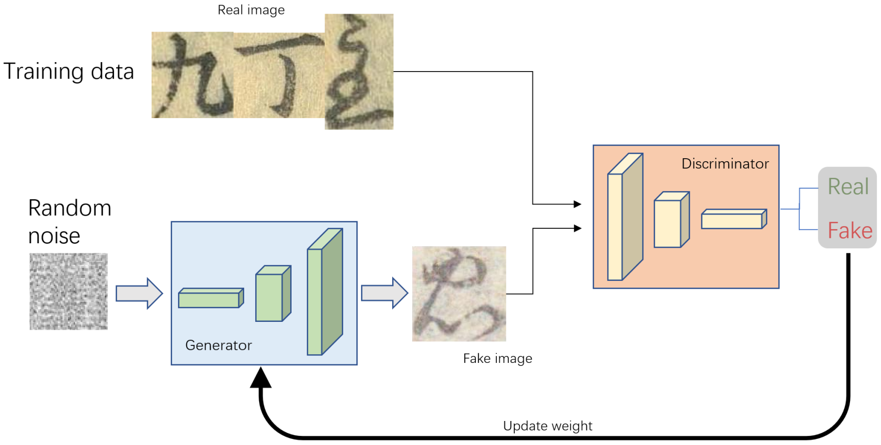


Figure 3: Basic structure of GAN

We crop the images of the classes with the sample size of 50-300 and randomly select 30 of them to input the GAN. After training the network, the images are output in different epochs; within 500 epochs, we will get 250 different images as new data.

3.2. YOLOv4 recognizes dataset

After the new images are generated, we use YOLOv4 to recognize them. Since YOLO requires the coordinate information of the target while taking object detection, we traverse the generated images to combine them and generate the location information of them. Our combination method is to first obtain the size of the generated image, and then take a blank image as the bottom, paste the generated image to the blank image in turn, each paste we will save the location information of the pasted image according to the size of the image and the order of the image paste. In order to ensure the uniqueness of the sample, we adopt the forward order traversal and reverse traversal, and limit the number of sample classes in an image, so as to ensure that there are no repeated classes in each image, and the order is not the same. We merge the generated dataset with the original dataset to obtain a large dataset with enough samples. After that, we put the large dataset into the YOLOv4 model for training. The YOLOv4 structural model is shown in Figure 4.

Kuzushiji recognition using YOLOv4 proceeds as follows.

Step1: Dividing the image into S×S grids, if the center of an object falls in the grid, the grid is responsible for detecting the object, and each grid outputs B bounding box information and C conditional class probabilities.

Step2: Putting image into the DarkNet53, and the feature information of the image is obtained from 3 different dimensions.

Step3: After feature information is extracted, putting it into the PANet layer to carry out multi-scale fusion of features, to enhance the localization ability on multiple scales.

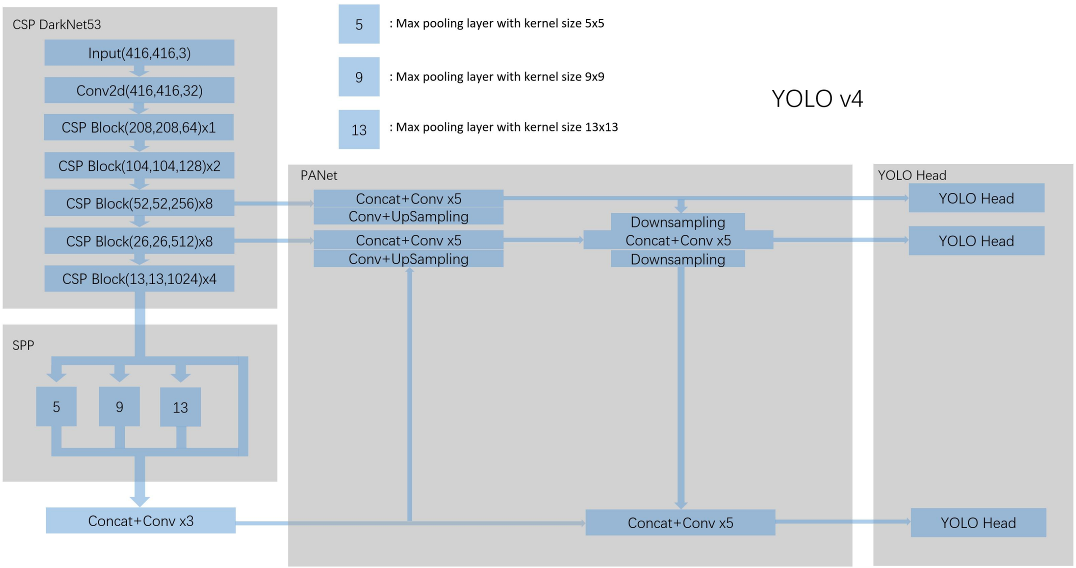


Figure 4: The structure of YOLOv4

Step4: Using YOLO head for prediction. By setting a threshold of 0.5, CIoU loss function is used to predict the location information of bounding box and the probability information of C objects belonging to a certain class. We use decoupled YOLO head to alleviate the inherent conflict between classification and regression tasks, and better fit our Kuzushiji dataset.

Step5: Using NMS (non maximum suppression) to search and discover potential objects. First, suppress targets with CIoU<0.5. If there are many classes of objects with CIoU >=0.5, this method can get the objects from the higher CIoU score to the lower.

Step6: The final step generates the result of Kuzushiji recognition.

4. Experiments and results

4.1. Data generation

4.1.1. Training Condition

The training environment of GAN is Nvidia RTX3090TI GPU accelerator and intel core i7-12700KF processor. In the training parameters, the batch size is set to 8, and the epoch is set to 6000.

4.1.2. Data generation results

Figure 6 shows the training results for each class. The generated image is different from the original image in subtle places, but we can't determine which image is real and which is virtual. Most of the generated images are high definition, only a few images are blurred or even missing. This is because some of the training data has the same problem, we don't process these data to simulate

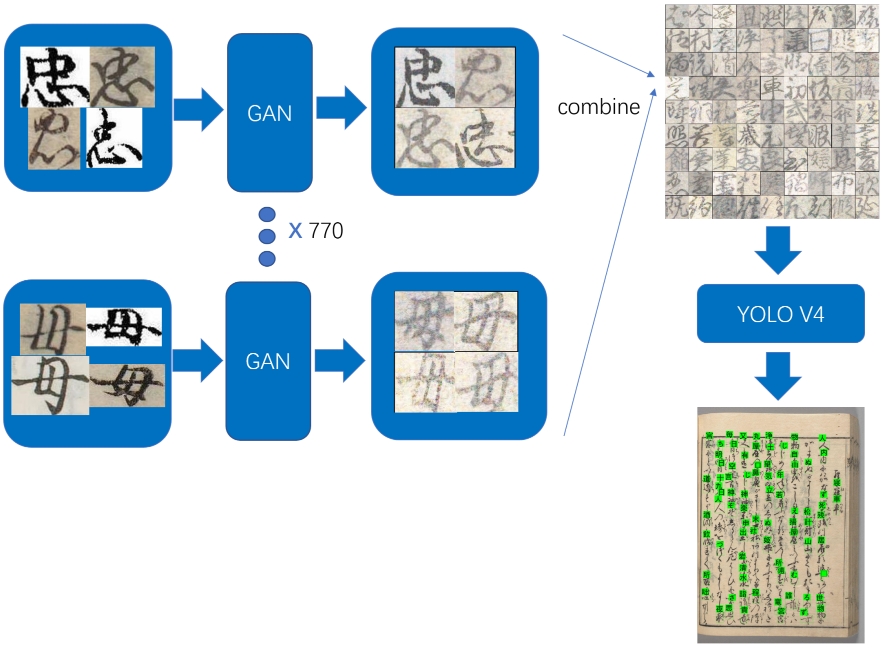


Figure 5: An overview of our methodology

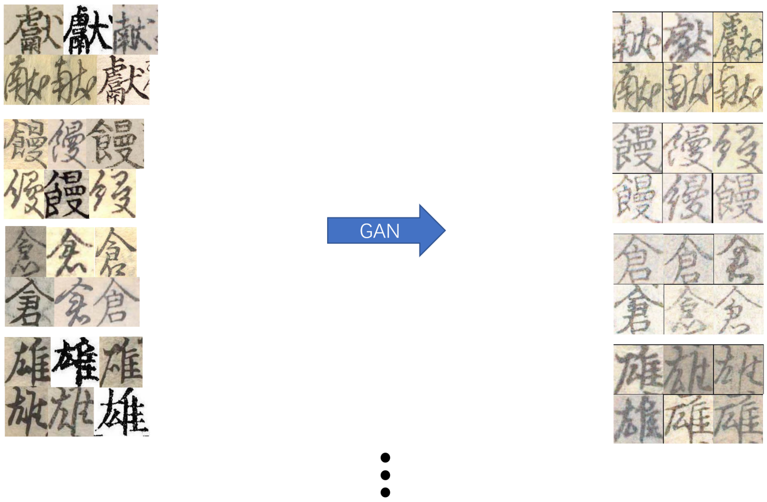
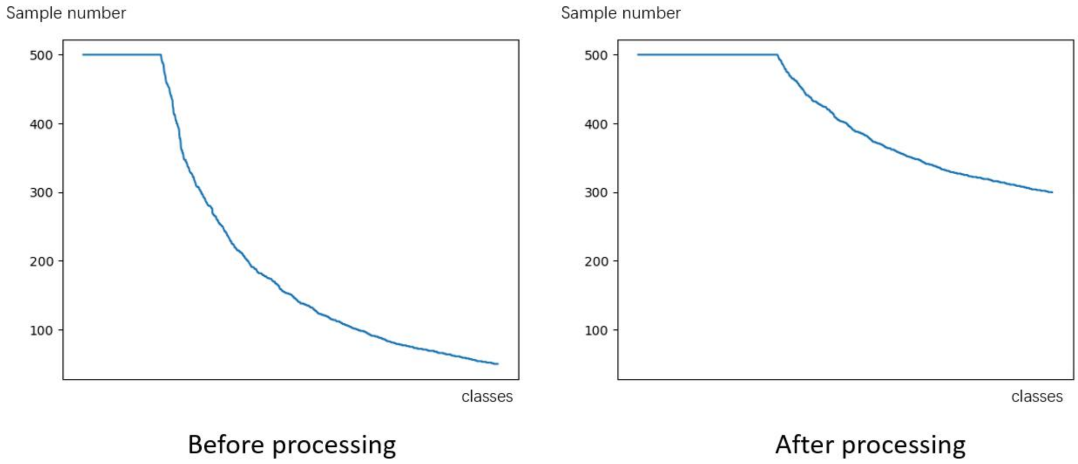


Figure 6: Training results for single class

the real phenomenon that appears in the documents. After all the classes are generated, we stitch the images. We stitched a total of 2,250 images. After combining the original 5,647 images, they are used as final training dataset.

We also deal with the problem of balance of the data. From Figure 2, we can see that the training data is seriously unbalanced, and some classes even have 30,000+ samples. To prevent the experimental results from being one-sided, we set the number of samples within 500 for all classes. So, for the class with a sample size of 50 to 300, we generated 250 images per class. Figure 7 shows the distribution of the number of samples in the dataset before and after processing.



Before processing

After processing

Figure 7: Comparison of data distribution

As can be seen from the Figure 7, these 1,073 categories are adjusted from extremely imbalanced data to sample number difference less than 200. Therefore, we can use the optimized dataset for YOLO detection.

4.2. YOLO recognition

4.2.1. Training Condition

In order to enhance the recognition effect of YOLO in the training phase, the initial learning rate is set to 0.001, and confidence is set to 0.5. Due to the large number of classes, we use Adam optimization function to speed up the convergence of the model, the number of epochs is set to 600 and use an uncoupled YOLO head which can predict the position of the bounding box, the classes and whether there is an object in the bounding box, respectively. The use of coupled YOLO head will result in uncertain internal conflicts in the prediction phase. Table 1 shows the comparison between the coupled and uncoupled YOLO head in terms of prediction accuracy. Using the uncoupled YOLO head can improve the mAP by 6%.

4.2.2. YOLO recognition results

Figure 8 shows the recognition of using YOLOv4 to train the original dataset and the large dataset. As can be seen from the figure, the model trained with the original dataset fails to recognize many infrequently used characters. However, the model trained with the larger dataset is able to recognize more characters. The detection accuracy of the former is 15.59% for the number of samples is greater than 50, and 46.81% for the number of samples is greater than 300. The latter are 37.24%

classes	coupled YOLO head	decoupled YOLO head
sample num>300	41.28%	46.81%
sample num>400	50.95%	57.42%
sample num>500	56.36%	64.97%

Table 1
Comparison of different yolo heads

classes	origin training set	large training set
sample num>50	15.59%	37.24%
50<sample num<300	37.19%	56.23%
sample num>300	46.81%	45.96%
sample num>400	57.42%	56.53%
sample num>500	64.97%	63.55%

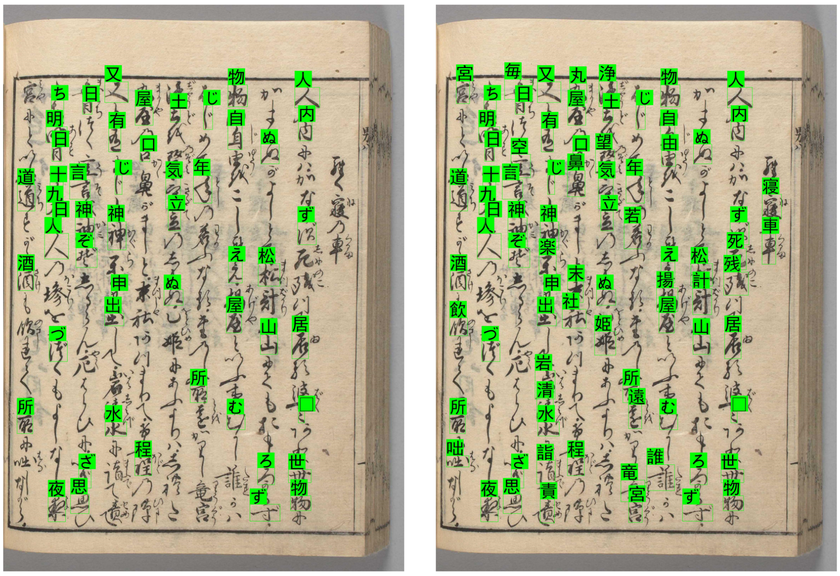
Table 2
Comparison of accuracy before and after dataset processing

and 45.96%, respectively. It can be seen that with the increase of the number of data sets, the introduction of training data unrelated to the unprocessed data sets will lead to a small decline in the recognition accuracy of this part of data, but the impact is not significant. However, for the data with a sample size of 50 to 300, the recognition accuracy of this part of data has been greatly improved due to the addition of this part of training data. This proves that training with an insufficient number of samples is far less effective than training with a large dataset which is augmented by our method.

5. Conclusion

In this paper, we use GAN to generate virtual images to augment the Kuzushiji dataset. This method generates amounts of new images by randomly sampling the potential space as input and modifying the parameters of the generated network through continuous discriminant correction. We combine the generated data with the original data to augment the dataset. And we propose to use decoupled YOLO head to train the large dataset, alleviating the inherent conflict between classification and regression tasks. According to our experimental results, the mAP of the original dataset is 15.59%, the model's mAP of the large dataset is 37.24%. The results verified the effectiveness of the proposed method. We provide a method to solve the imbalance of Kuzushiji, so that we can train a YOLO model that can achieve better detection results. It can better help people understand the ancient history of Japan through AI methods.

In the future, we hope to experiment with different GANs, such as DCGAN, LSGAN, WGAN to compare the data generated by different GANs, and find a network that better fits the dataset. Secondly, trying to improve the YOLO to achieve the highest performance. Thirdly, we want to try another way of stitching the generated images, which is to insert the generated image into the original book image. Augmenting the dataset with the same number of images.



Origin dataset trained model detection result

Large dataset trained model detection result

Figure 8: Comparison of detection results

References

[1] E. Palmer, Harima Fudoki: A Record of Ancient Japan Reinterpreted, Translated, Annotated, and with Commentary, Brill, 2015.

[2] W. T. De Bary, C. Gluck, A. Tiedemann, Sources of Japanese tradition: 1600 to 2000, Columbia University Press, 2005.

[3] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, Y. Bengio, Generative adversarial nets, 2014.

[4] A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection, 2020. ArXiv:2004.10934 [cs, eess].

[5] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, 2016, pp. 779–788.

[6] T. Clanuwat, M. Bober-Irizar, A. Kitamoto, A. Lamb, K. Yamamoto, D. Ha, Deep Learning for Classical Japanese Literature, 9999. ArXiv:1812.01718 [cs, stat].

[7] A. Lamb, T. Clanuwat, A. Kitamoto, KuroNet: Regularized Residual U-Nets for End-to-End Kuzushiji Character Recognition, SN Computer Science 1 (2020) 177.

[8] D. Singh, C. V. Aravinda, M. Kaur, M. Lin, J. Shetty, V. R. Reddicherla, H.-N. Lee, DKNet: Deep Kuzushiji Characters Recognition Network, IEEE Access 10 (2022) 75872–75883.

[9] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017. ArXiv:1704.04861 [cs].

[10] A. C. V, L. Meng, A. Masahiko, U. Kumar, A. Prabhu, A Complete Methodology for Kuzushiji Historical Character Recognition using Multiple Features Approach and Deep Learning Model, International Journal of Advanced Computer Science and Applications 11 (2020).

[11] B. Lyu, H. Li, A. Tanaka, L. Meng, The early Japanese books reorganization by combining image processing and deep learning, CAAI Transactions on Intelligence Technology 7 (2022) 627–643.

[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, MobileNetV2: Inverted Residuals and Linear Bottlenecks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Salt Lake City, UT, 2018, pp. 4510–4520.

[13] X. Hu, M. Inamoto, A. Konagaya, Recognition of Kuzushi-ji with Deep Learning Method: A Case Study of Kiritsubo Chapter in the Tale of Genji (2019).

[14] S. Hochreiter, J. Schmidhuber, Long Short-Term Memory, Neural Computation 9 (1997) 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[15] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, D. Rueckert, GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks, 2018. ArXiv:1810.10863 [cs].

[16] C. Dewi, R.-C. Chen, Y.-T. Liu, X. Jiang, K. D. Hartomo, Yolo V4 for Advanced Traffic Sign Recognition With Synthetic Training Data Generated by Various GAN, IEEE Access 9 (2021) 97228–97242.

[17] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, H. Greenspan, Synthetic Data Augmentation using GAN for Improved Liver Lesion Classification, 2018. ArXiv:1801.02385 [cs].

[18] A. Bissoto, E. Valle, S. Avila, GAN-Based Data Augmentation and Anonymization for Skin-Lesion Analysis: A Critical Review, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Nashville, TN, USA, 2021, pp. 1847–1856.

[19] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, C. Malossi, BAGAN: Data Augmentation with Balancing GAN, 2018. ArXiv:1803.09655 [cs, stat].

[20] X. Yue, H. Li, Y. Fujikawa, L. Meng, Dynamic dataset augmentation for deep learning-based oracle bone inscriptions recognition, J. Comput. Cult. Herit. 15 (2022). URL: https://doi.org/10.1145/3532868.

[21] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Goncalves, W. R. Schwartz, D. Menotti, A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE, Rio de Janeiro, Brazil, 2018, pp. 1–10.

[22] Y. Tang, K. Hatano, E. Takimoto, Recognition of japanese historical hand-written characters based on object detection methods, in: Proceedings of the 5th International Workshop on Historical Document Imaging and Processing, 2019, pp. 72–77.

[23] J. Redmon, A. Farhadi, YOLOv3: An Incremental Improvement, 2018. ArXiv:1804.02767 [cs].

[24] R. Santoso, Y. K. Suprapto, E. M. Yuniarno, Kawi Character Recognition on Copper Inscription Using YOLO Object Detection, in: 2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), IEEE, Surabaya, Indonesia, 2020, pp. 343–348.