Paper:

Genetic Algorithm Based Automatic Layer Selection of Transfer Learning for Object Detection

Ryuji Ito*, **Hajime Nobuhara****, **and Shigeru Kato*****

*University of Tsukuba, 1-1-1 Tennoudai, Tsukuba, Ibaraki, 305-8573 Japan
E-mail: ito@cmu.iit.tsukuba.ac.jp
**University of Tsukuba, 1-1-1 Tennoudai, Tsukuba, Ibaraki, 305-8573 Japan
E-mail: nobuhara@iit.tsukuba.ac.jp
***Niihama College, National Institute of Technology, 7-1 Yagumochou, Niihama City, Ehime Prefecture 792-8580, Japan
E-mail: s.kato@niihama-nct.ac.jp
[Received 00/00/00; accepted 00/00/00]

Abstract. A method for automatic re-learning layer selection based on a genetic algorithm is proposed to solve the difficulty of conventional transfer learning of deep learning-based object detection models. The genetic algorithm of the proposed method can select the re-learning layers automatically in the transfer learning process instead of a trial-and-error selection of the conventional method. A transfer learning experiment from the COCO dataset to the Global Wheat Head Detection (GWHD for short) dataset was performed using fine-tuning and the proposed method, and the results were compared. Using the training data and the validation data of the GWHD, we compare the mean average precision of the models trained by the conventional and the proposed methods, respectively, on the test data of the GWHD. It is confirmed that the model trained by the proposed method has higher performance than the model trained by the conventional method. The average of mAP of the proposed method, which automatically selects the re-learning layer (≈0.603), is higher than the average of mAP of the conventional method (≈0.594). Furthermore, the standard deviation of results obtained by the proposed method is smaller than that of the conventional method, and it shows the stable learning process of the proposed method.

Keywords: deep learning, genetic algorithm, object detection, transfer learning

1. INTRODUCTION

Object detection refers to detecting the position and class of a defined object in a given image. Object detection is currently used in a various fields, including autonomous driving and agriculture[17]. Since the technique using deep learning in image recognition demonstrated high performance[7], many models using convolutional neural networks have been proposed for object detection[5].

In deep learning, such as convolutional neural networks, a large amount of data is required to provide high performance to the network. On the other hand, it is often difficult to prepare large amounts of data due to economic and time constraints. Also, even if a large amount of image data is available, it is quite burdensome when labeling it manually. When there is no large dataset, transfer learning is used, in which a previously trained network is reused to reduce the amount of data required for training[11]. By fixing a part of the network that has been trained in the past by another task and re-learning the other layers with the target task, we reuse the trained network.

In transfer learning, it is essential to select the layer where the parameters are fixed (fixed layer) and the layer where they are updated (re-learning layer). A standard method is called fine-tuning, which uses the weights of the pre-trained model as the initial values of the weights and updates the entire layer[12]. Several re-learning layers have been set up manually, and transfer learning experiments have been conducted on classifying Bangali numeral[18]. These experiments show that the choice of the re-learning layer has a significant impact on the performance after learning. However, the number of layers in the network has been increasing rapidly in the past few years, and it has become difficult to manually select the fixed layer and the re-learning layer. Therefore, methods to automatically select the layer to be retrained, such as a method called "Stepwise PathNet"[4] using tournament selection and a method using genetic algorithms[15][10], have been proposed for image recognition tasks. Both methods were tested by transferring a pre-trained ImageNet model to the CIFAR-100 dataset[6] and showed better results than fine-tuning.

Transfer learning is also used in object detection. For object detection tasks, methods that re-learn all layers, such as fine-tuning, are widely used. However, instead of re-learning all the layers, the performance could be improved by setting each layer as a layer to be re-learned and a layer to be fixed. In addition, deep learning-based object detection models often have many more layers than image recognition models. Therefore, the setting of re-learning

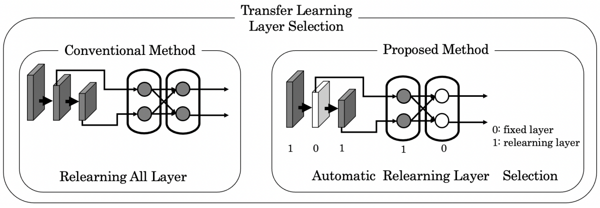

Fig. 1. Comparison of conventional and proposed method

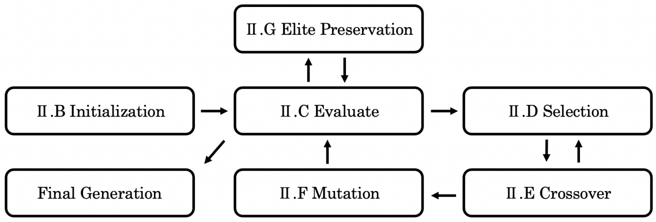

Fig. 2. Overview of proposed method

and fixing of layers in deep learning-based object detection models should be selected automatically instead of manually.

In this paper, we propose a transfer learning method with automatic selection of re-learning layers for object detection models based on genetic algorithms. Figure 1 shows the conventional and proposed methods in transfer learning. Since the proposed method does not change the structure of the network or the number of parameters, it can improve the performance by selecting suitable layers for re-learning without increasing the time for inference or the size of the network. This is useful when the users want to improve the performance of a model when there are limitations on time taken for inference or the size of the model.

In the experiments, we performed transfer learning of the EfficientDet-D0[14] model pre-trained on the COCO dataset[8] to the GWHD dataset[2] using a proposed method. The results show that the genetic algorithm provides a selection of re-learning layers that leads to improved performance.

2. PROPOSED METHOD

2.1. Overview

The proposed method is based on genetic algorithms. A genetic algorithm is a stochastic search algorithm based on mimic the mechanisms of evolution in the biological world. The proposed method uses chromosomes to represent the re-learning layer and the fixed layer during transfer learning. Then, we train on the training data and keep the chromosomes that are highly evaluated on the validation data for the next generation. The proposed method is executed using each operation of initialization, evaluation, selection, crossover, mutation, and elite preservation. Finally, among the chromosomes generated, the chromosome with the highest degree of fitness is obtained. The overview of the proposed method is shown in Fig. 2. We also describe each process of the proposed method in Algorithm 1.

2.2. Initialization

We defined chromosomes that indicate which layer of the pre-trained model used for transfer learning is the re-learning layer and which layer is the fixed layer. Chromosomes are represented as binary, with 1 and 0 representing the re-learning and fixed layers, respectively. For example, in the chromosome [0,1,1,0,1] in the Fig. 3, layers 2, 3, and 5 are the re-learning layers, and layers 1 and 4 are the fixed layers. Based on this definition of a chromosome, N chromosomes are generated. Each chromosome generated using a random number represents whether the layer of the entire model is a re-learning layer or a fixed layer. In the experiments, two methods are used during initial population generation. The two methods are to select a re-learning layer with a constant probability on the generated chromosomes in the population or to select a re-learning layer with a non-constant probability.

Algorithm 1 Algorithm of the proposed method

Generate N chromosomes
Generation $\Leftarrow$ 0
while Generation $<$ Final Generation **do**
  Evaluate each chromosome
  for $i = 0$ to $N - E$ **do**
    Select two parental chromosomes
    Crossover parental chromosomes and generate child chromosome
    Mutate a child chromosome
  end for
  Elite preservation of E chromosomes.
  Generation $\Leftarrow$ Generation + 1
end while
return The best chromosome among the generated chromosomes

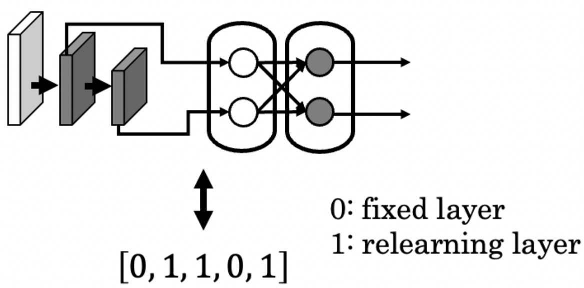

Fig. 3. Representation of chromosome used in the proposed method

2.3. Evaluation

Each generated chromosome in the population is evaluated, and the degree of fitness is calculated. The evaluation is performed by learning an object detection model reflecting the selection of a re-learning layer expressed by each chromosome. Training data and validation data are prepared from a target dataset, and learning is performed by using the training data by setting at a pre-determined learning setting. Then, the object detection model is evaluated by a pre-determined evaluation index using the learned model and the validation dataset. In the experiments, we used two degrees of fitness: the reciprocal of the loss function on the validation data and the mean Average Precision (mAP, for short) on the validation data.

2.4. Selection

In the selection process, parental chromosomes are selected based on their degree of fitness. As the selection method, a roulette selection method, a tournament selection method, an elite selection method, and the like can be used. In our experiments, we used the tournament selection method. The tournament selection method is a selection method in which a fixed number of chromosomes are randomly taken from a population, and the one with the highest degree of fitness is selected.

2.5. Crossover

In the crossover, child chromosomes are generated based on the parent chromosomes selected in the selection process. In our experiments, we used a single-point crossover. Single-point crossover is a crossover method in which the genes of two selected parents are interchanged by cutting at a crossover point. An example of a single point crossover is shown in Fig. 4.

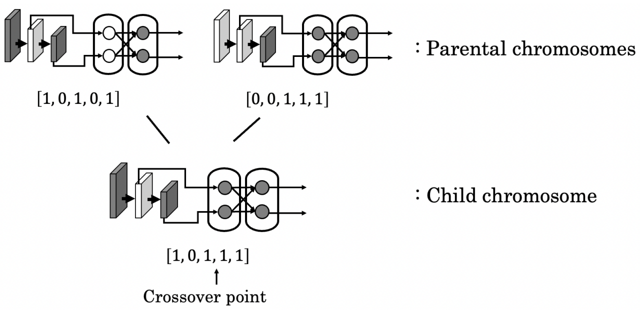

Fig. 4. Example of crossover

The process of selection and crossover generates N-E child chromosomes.

2.6. Mutation

The mutation that inverts the value of each gene occurs at a certain probability p% in the offspring produced by crossover. This mutation operation is performed on the N-E child chromosomes generated during the selection and crossover process.

2.7. Elite Preservation

Elite preservation is used so that chromosomes which have high degree of fitness are not lost during the process of crossover or mutation. Elite preservation leaves some chromosomes which have high degree of fitness for the next generation.

3. EXPERIMENTS

The experiments are performed by transfer training the EfficientDet-D0 model, which has been previously trained on the COCO dataset, to the Global Wheat Detection Dataset using fine-tuning and the proposed method.

3.1. Model Architecture

EfficientDet is an object detection model that uses the classification model EfficientNet[13] for feature extraction. Like EfficientNet, it introduces a parameter to scale the capacity of the network to achieve a balance between FLOPs and accuracy. The architecture of EfficientDet is shown in Fig. 5. EfficientDet consists of four networks: backbone network, fpn network, class prediction net, and box prediction net. The capacity of EfficientDet changes as the number of layers of each of these networks and other factors change. The models vary in capacity from EfficientDet-D0 to EfficientDet-D7. In this experiment, EfficientDet-D0, the lightest model, is used, and the COCO dataset is trained in advance.

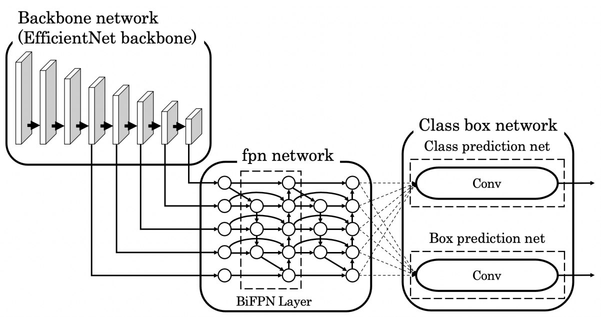

Fig. 5. Architecture of EfficientDet

3.2. Dataset and Data Augmentation

The GWHD dataset was used for the transfer learning experiments. The GWHD dataset is a large dataset of labeled wheat images built for the purpose of developing and benchmarking wheat head detection methods. The dataset contains 4,700 high-resolution RGB images and 190,000 labeled wheat heads from all over the world. We used 80% of the training data from this dataset as training data and 20% as validation data to train our model. Figure 6 shows a few sample images of the GWHD dataset.

We resized the images in this GWHD dataset to 512×512 to fit the input image size of the EfficientDet-D0

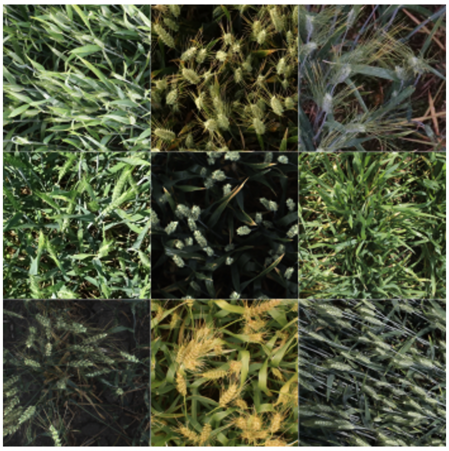

Fig. 6. Sample images from the GWHD dataset

model after applying several transformations. The following table shows the data expansion methods used in the experiments for each method.

- Random Size Crop
- Hue Saturation Value
- Random Brightness Contrast
- To Gray
- Horizontal, Vertical Flip
- Cutout[3]
- CutMix[16]

3.3. Evaluation

Experiments were conducted using the three proposed methods (a), (b), and (c). In setting (a), the probability of re-learning layer selection is fixed at 50% for all chromosomes in the initial population generation, and the reciprocal of the validation loss is used as the degree of fitness. In setting (b), the probability of re-learning layer selection is fixed at 50% for all chromosomes in the initial population generation, and mAP is used as the degree of fitness. In setting (c), it is to make the probability of re-learning layer selection variable in the initial population generation. Specifically, the probability of re-learning layer selection was changed in the range of 10% to 90% for each chromosome. The mAP is used as the degree of fitness.

By changing the degree of fitness, the remaining chromosomes are changed by the genetic algorithm. The degree of fitness is a very important item in the genetic algorithm, and we will confirm which is better in this experimental setting, the reciprocal of the validation loss or mAP. By weighting the probability of re-learning layer selection, the search for a solution is considered to be broad in terms of the ratio of re-learning layers. However, it is a sparse search around the same ratio. In the setting of this experiment, it is confirmed whether the fixed probability or the variable probability gives a better result.

Five experiments were conducted for fine-tuning as a conventional method and for each of the three proposed methods. The number of chromosomes in the population was set to n=15, and the number of generations was set to Final Generation=3. In the fine-tuning method, the pre-training model is trained for 30 epochs. In the proposed method, we train a model that reflects each chromosome for 15 epochs to find the optimal re-learning layer setting among the generated chromosomes. Then, we compare the proposed method with the fine-tuning method by training 30 epochs of the pre-trained model with the optimal re-learning layer settings among the generated chromosomes. For all methods, the optimization method used was AdamW[9], the batch size was set to 4, and it was run using a GeForce GTX 1080Ti graphics card. We compare the accuracy of each method on the test data of the GWHD dataset. Although the test data is not public, the accuracy of the test data was calculated by using the Global Wheat Detection competition on Kaggle's website[1]. Since both public and private scores are calculated in the competition, the average of these two scores is used as the accuracy.

Table 1. Experimental result of each method

Re-Learning Network	Validation Loss	mAP
Backbone, fpn network	9.514	0.072
Class, box prediction network	0.839	0.074

3.4. Degree of Fitness

In the experiment, the reciprocal of the validation loss and mAP is used as the degree of fitness. As shown in Figure 5, EfficientDet consists of four networks: backbone network, fpn network, class prediction net, and box prediction net. Fig. 7 shows the inference results for the EfficientDet-D0 model when only the backbone network and the fpn network are re-learned. Fig. 8 also shows the inference results when only the class prediction network and the box prediction network of the EfficientDet-D0 model are re-learned. Table 1 also shows the loss and mAP for the validation data in each re-learning situation. In Fig. 7, the tip of the ear is detected more often than in Fig. 8, but there are many extra detections. However, verification loss is lower in Fig. 8. There is no significant difference in mAP. Since the number of wheat ears that we are able to detect is increasing, but there are many unfavorable detections, we thought that a less favorable indicator such as mAP might be suitable as the degree of fitness, so we used two indicators as the degree of fitness.

Fig. 7. Inference results (re-learned backbone and fpn networks)

Fig. 8. Inference results (re-learned class, box prediction networks)

Table 2. Experimental result of each method

Number	Conventional Method	Proposed Method		
	Fine-tuning	(a)	(b)	(c)
1	0.5951	0.6040	0.6050	0.6045
2	0.5930	0.5996	0.6037	0.6031
3	0.5951	0.6035	0.6035	0.6032
4	0.5935	0.6017	0.6038	0.6041
5	0.5949	0.6032	0.6020	0.6030
Ave.	0.5943	0.6024	0.6036	0.6036
Std.	0.0010	0.0018	0.0011	0.0007

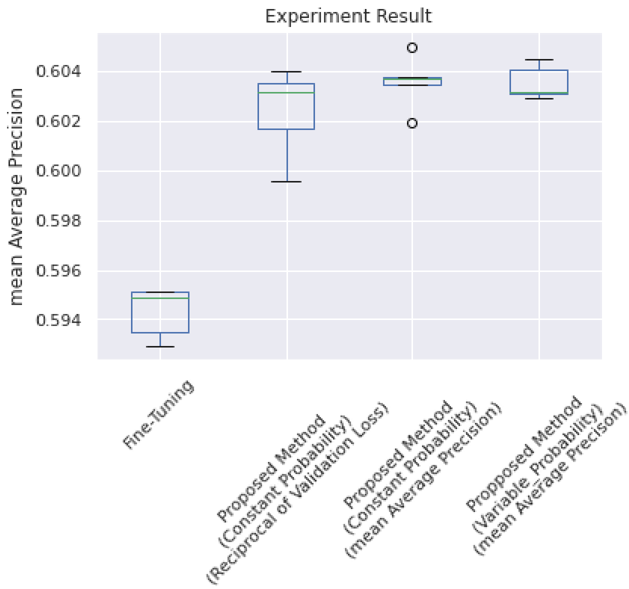

Fig. 9. Results of each method

4. RESULTS and DISCUSSION

4.1. Results of Each Method

Table 2 shows the experimental results of fine-tuning and the three proposed methods for transfer learning. Furthermore, Fig. 9 shows the boxplot of the experimental results for each method. For the probability that each layer becomes a re-learning layer for each chromosome in the initial population generation, the proposed method (a) and the proposed method (b) use a constant probability for each chromosome, while the proposed method (c) uses a different probability for each chromosome. For the degree of fitness, the inverse of the degree of fitness is used in the proposed method (a), and mAP is used in proposed methods (b) and (c). As shown in the table, while the average value of the conventional method, fine-tuning, is 0.59430, in the experiments of the three proposed methods, the average values are 0.60238, 0.60357, and 0.60356, respectively, indicating that the proposed method has better accuracy than the conventional method. This suggests that rather than retraining all the layers, a better selection of the layers to be re-learned will lead to improved accuracy. Furthermore, the standard deviation of results obtained by the proposed method is smaller than that of the conventional method, and it shows the stable learning process of the proposed method. The experimental results show that the proposed method is able to automatically select a good re-learning layer.

Among the three proposed methods, the two that used mAP for the degree of fitness had better accuracy, albeit by a small amount. In particular, the maximum accuracy of the experiment using mAP for the degree of fitness and constant probability for the initial population generation had the highest accuracy in all experiments. When mAP is used for the degree of fitness, the variation of accuracy is smaller, and when compared with the minimum value, the result is better than when the reciprocal of the verification loss is used. We believe that the mAP is a better indicator of whether the re-learning layer is better selected than the reciprocal of the validation loss.

On the other hand, if the probability of initial population generation was kept constant, the search would become localized, and if the probability of initial population generation was varied, the search would become broad,

Table 3. Ratio of re-learning layer

	Proposed Method		
Number	(a)	(b)	(c)
1	52.2%	51.7%	85.7%
2	50.0%	49.4%	75.2%
3	50.9%	52.4%	85.2%
4	51.5%	48.3%	83.9%
5	50.7%	49.8%	83.9%

and even if the probability of initial population generation was varied, there was not much improvement in accuracy.

4.2. Ratio of Re-learning Layer

Table 3 shows the percentage of re-learning layers represented by the chromosomes with the highest degree of fitness in the transfer learning experiments with the three proposed methods. When a constant probability is used to generate the initial population, the percentage of re-learning layers is still around 50%. When the probability is variable, the percentage of re-learning layers is around 85% and 75%. Even when there was no significant difference in accuracy, the ratio and arrangement of the re-learning layers were found to be different. This suggests that there are several arrangements and proportions of re-learning layers that are suitable for transfer learning. However, even if there are some suitable arrangements and proportions, it would be difficult to find them manually in a model with many layers. The proposed method is effective because it can automatically derive suitable re-learning layers for transfer learning.

5. CONCLUSION

In this paper, we proposed a method for automatic re-learning layer selection during transfer learning of object detection models based on genetic algorithms. We conducted transfer learning experiments using the COCO dataset with the pre-trained EfficientDet-D0 model and the GWHD dataset. In the experiment, we used fine-tuning as a conventional method and compared three proposed methods using genetic algorithms with different settings of the degree of fitness and initial population generation. The conventional method and the proposed method were compared in terms of mAP, which was measured using the Kaggle website. In the experimental results, the proposed method was higher than fine-tuning on average and showed stable accuracy.

This indicates that the accuracy after transfer learning is better when some of the layers are re-learning layers rather than all of the layers are re-learning layers. In addition, even though the accuracy was not significantly different, the placement and ratio of the re-learning layers were different, indicating that there are multiple placement and ratio of re-learning layers that are suitable for transfer learning. However, in a model with a large number of layers, it is difficult to select the re-learning layers manually, and the proposed method that automatically selects the re-learning layers is considered to be effective.

In this paper, we used the GWHD dataset for transfer learning experiments. In the future, we would like to conduct experiments using other datasets to demonstrate the effectiveness of transfer learning methods for the automatic selection of re-learning layers using genetic algorithms. Due to computational costs, the number of chromosomes and generations in the population was limited. There is a need for further improvement in terms of accuracy and computational cost by improving each process, such as initial population generation, crossover, selection, and mutation.

References:

[1] "Global Wheat Detection". https://www.kaggle.com/c/global-wheat-detection. Accessed: 2021-05-06.

[2] E. David, S. Madec, P. Sadeghi-Tehran, H. Aasen, B. Zheng, S. Liu, N. Kirchgessner, G. Ishikawa, K. Nagasawa, M. A. Badhon, et al., "Global Wheat Head Detection (GWHD) dataset: a large and diverse dataset of high-resolution RGB-labelled images to develop and benchmark wheat head detection methods", Plant Phenomics, 2020, 2020.

[3] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout", arXiv preprint arXiv:1708.04552, 2017.

[4] S. Imai and H. Nobuhara, "Stepwise PathNet: Transfer Learning Algorithm to Improve Network Structure Versatility", In 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 918–922, 2018.

[5] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A Survey of Deep Learning-Based Object Detection", IEEE Access, 7, 2019.

[6] A. Krizhevsky, G. Hinton, et al., "Learning multiple layers of features from tiny images", 2009.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 25, pp. 1097–1105, Curran Associates, Inc., 2012.

[8] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context", In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, pp. 740–755, Cham, 2014, Springer International Publishing.

[9] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization", In International Conference on Learning Representations, 2019.

[10] S. Nagae, S. Kawai, and H. Nobuhara, "Transfer Learning Layer Selection Using Genetic Algorithm", In 2020 IEEE Congress on Evolutionary Computation (CEC), pp. 1–6, 2020.

[11] S. J. Pan and Q. Yang, "A Survey on Transfer Learning", IEEE Transactions on Knowledge and Data Engineering, 22(10):1345–1359, 2010.

[12] Y. Sawada and K. Kozuka, "Whole layers transfer learning of deep neural networks for a small scale dataset", International Journal of Machine Learning and Computing, 6(1):27, 2016.

[13] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks", In International Conference on Machine Learning, pp. 6105–6114. PMLR, 2019.

[14] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection", In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10781–10790, 2020.

[15] A. Thengade and R. Dondal, "Genetic algorithm–survey paper", In MPGI National Multi Conference, pp. 7–8. Citeseer, 2012.

[16] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features", In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032, 2019.

[17] Z. Zou, Z. Shi, Y. Guo, and J. Ye. "Object Detection in 20 Years: A Survey", 2019.

[18] H. Zunair, N. Mohammed, and S. Momen, "Unconventional Wisdom: A New Transfer Learning Approach Applied to Bengali Numeral Classification", In 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), pp. 1–6, 2018.