# Comparison of Vision Transformers and a Convolutional Neural Network Solving the Binary Data Problem

1st Tuomas Ylönen
*Information and Communications Technology*,
*Turku University of Applied Sciences*,
tuomas.ylonen@edu.turkuamk.fi

2nd Yoko Nakajima
*Department of the Creative Engineering*,
*National Institute of Technology*,
*Kushiro College*,
yoko@kushiro-ct.ac.jp

3rd Hirotoshi Honma
*Department of the Creative Engineering*,
*National Institute of Technology*,
*Kushiro College*,
honma@kushiro-ct.ac.jp

***Abstract*—While the introduction of Vision transformer type of AI models has been a growing trend, the Vision Transformers' ability to solve binary data problems has not been as visible. This research compares the solving ability of convolutional neural network type of AI model with three different vision transformers and their ability of solving the same binary data problem using images. All models in the research are using same dataset as the base for learning and executing the problem. The differences will come from the data augmentation used by the models and the model constructs. The priority in this research lies in getting proof for Vision transformers' capability to solve the problem compared to more traditional convolutional neural network. Secondary objective lies in the accuracy of the used different vision transformers' results and whether they could be useful in such problem solving.**

## I. Introduction

Since the introduction of Vision Transformer usage in computer vision problem solving [1] there have been various types of vision transformers invented. For example, Pyramid Vision Transformer [2], Cross-Attention Multi-Scale Vision Transformer [3] or MobileViT [4]. Since the large commonly used datasets for research (e.g., ImageNet, CIFAR-10, CIFAR-100) have a several number of classes, the datasets encourage to use softmax as an activation function. In this paper we use Vision Transformer models with sigmoid activation function to produce binary outcomes. These outcomes are then compared to a CNN model which has solved the same problem to give insight whether Vision Transformers could also perform reliably with binary data problems that require a small dataset.

The results suggests comfortably that Vision Transformers without any convolutions will be able to solve binary problems with smaller datasets. Vision transformers that use elements from CNN models and more data augmentation, produced more accurate results. Careful learning rate decay planning and hyperparameter adjustment produced more accurate results as well. This also suggests that for the Vision transformers to work with smaller datasets, solving a binary data problem would require resources for more advanced data augmentation and hyperparameter adjustment.

### A. The data problem

The data problem for the models was an image classification problem with only two possible results thus resulting the problem to be binary. Dataset consists of cabbage pictures taken from a sky and the cabbage conditioned is labeled to 'good' as in healthy cabbage or 'bad' as in unhealthy, rotten, dried, or having problems of such nature so the cabbage would be defined inedible or in need of an attention from the farmer. AI models will then predict the cabbage in the test data to be healthy or unhealthy and classify them either good or bad. Model's prediction results are compared to correct results to get the respective accuracy on the test data. This accuracy was finally compared between models to find out differences in the used models' performance.

### B. Models

First, we created an accurate CNN model for this dataset and its problem. We used Keras and a Sequential model structure to create a custom and simple CNN model, which using this dataset, achieved over 99% accuracy for the test data. Using a CNN model as the base comparator for this research was decided because of CNN model type of AI models have performed very well at handling classification problems using images. For example, a flower classification problem solved with CNN model [5] is a proper example of such excellent results. To then compare Vision Transformer type of an AI model performance for the same problem, three different Vision Transformer models were used.

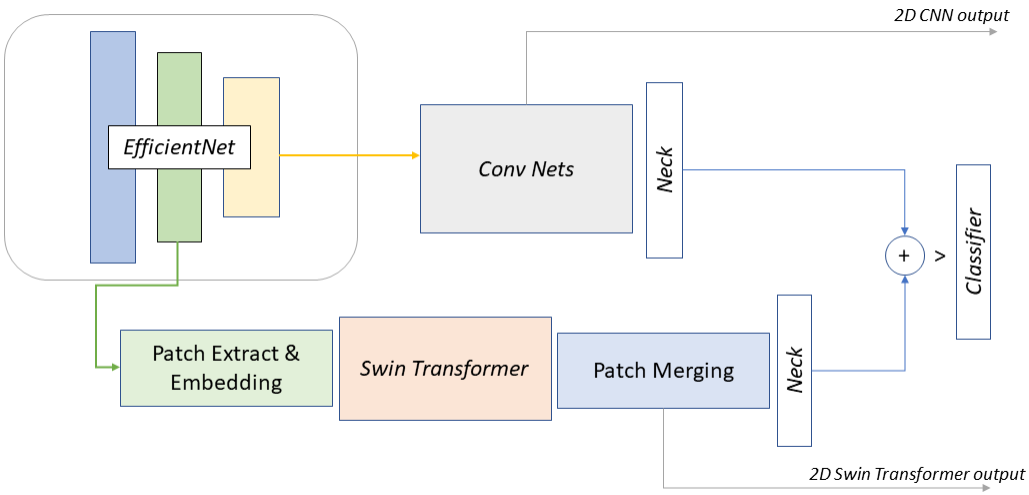


Fig. 1. Hybrid-EfficientNet-Swin Transformer from https://github.com/innat/HybridModel-GradCAM .

One of the models is identical to the original paper's Fine-tuned B16 [1] in the lone difference that the activation function was changed to sigmoid for the purpose of producing binary outcome. Two other models used combine properties of convolutions or convolutions as is with transformers. MobileViT architecture type of model was used as a lightweight counterpart for the B16 with significantly less parameters. MobileViT uses MobileViT blocks [4] which are light weight while using standard convolutions with transformers. The combined usage will lead for the MobileViT block to learn both local and global representations. MobileViT architecture is furthermore simple to use with Keras to implement it to a model, making it to be viable for this research's purposes.

Third model is a hybrid model using Swin-Transformer and part of pre-trained EfficientNet model structure along with more advanced data augmentation. This brings to a more oriented model suitable for this research's particular data problem because this hybrid model was originally built to recognize different flowers from each other[1]. The Swin-Transformer model is hereafter referred to as 'hybrid model' in this paper. Figure 1 will show the Hybrid Model's construct in a simplified way. EfficientNet B0 Model is used as an input for Swin-Transformer blocks. The input shape is set for 128 to correspond for the image size used in this research. Output from the EfficientNet B0 Model's layer *block6a_expand_activation* is then used as an input for the Swin-Transformer. Table 1 shows the parameter differences between the models used in this research.

TABLE I
PARAMETER AMOUNT OF MODELS USED IN THIS RESEARCH.

| Model | Parameters |
|---|---|
| CNN | 80121 |
| B16 | $85.697 \times 10^6$ |
| MViT | $1.307 \times 10^6$ |
| Hybrid Model | $4.292 \times 10^6$ |

## II. Related work

Since the original paper [1] introduced using a pure vision transformer solving image classification problems, there have been several VisionTtransformer models combining properties of the CNN models and Vision Transformers. Pyramid Vision Transformer [2] was tested on ImageNet 2021 dataset with 1000 classes and resulted 18.8% – 24.9% Top-1 Error percentage. CSWin Transformer [6] models were experimented on ImageNet-1K dataset. All of the CSWin Transformer models achieved over 82% with Top-1 percentage. Swin Transformer [7] models used ImageNet-1K and pre-trained models used ImageNet-22K in their experiments. Swin Transformers achieved over 81% Top-1 accuracies with ImageNet-1K models and the pretrained ImageNet-22K Swin Transformers resulted on over 85% Top-1 accuracies. The results promise successful variants of Transformers used in image classification. It is suggested that instability is a serious issue with Transformers [8] which can be improved with several methods. Another hypothesized finding for Vision Transformer inferior performance compared to similar-sized CNN counterparts is that Vision Transformers are unable to model image edges and lines like CNNs [9]. This results in Vision Transformers needing more training samples than CNNs to achieve similarity in performance.

## III. Training details

All Vision Transformer models were constructed and trained 10 times with the same parameters, dataset, and augmentation settings they were set to. The CNN model was trained 33

[1]Code was implemented from an experimental model https://github.com/innat/HybridModel-GradCAM and modified further for producing binary outcomes without the visual interpretations of the original model.

epochs with a batch size of 32. CNN had fixed learning rate of $10^{-3}$ and Adam was used as the optimizer. B16 and MViT had learning rate change between $10^{-5}$ and $10^{-6}$. B16 and MViT used rectified Adam as optimizer. B16 had 30 of maximum epochs per training run and MViT had 50 epochs. The hybrid model was trained maximum of 15 epochs with a cosine learning rate scheduler starting from 0 and warming up to $5 \times 10^{-4}$. Training schedules include only a few number of epochs since this size of dataset tend to overfit rapidly [10].

Vision Transformer models hyperparameter updating was not optimized to the fullest, instead the purpose was to find out whether there were possibilities for the Vision Transformers to reliably solve the data problem. The reason for this was that the resources for the research were limited in several aspects.

### A. Dataset and augmentation

Same dataset of cabbage pictures was used for all ViT models and the CNN model. The differences came from data augmentation used by models and Keras. Test data of the dataset was never modified in any way. The same size of 400 pictures of test data was used for all models in this research in its original shape and size of 128 x 128 pixels. This differs from more widely used 224 x 224 pixels [1], [11] or 384 x 384 pixels [12].

Fig. 2. Synthetically created training data examples.

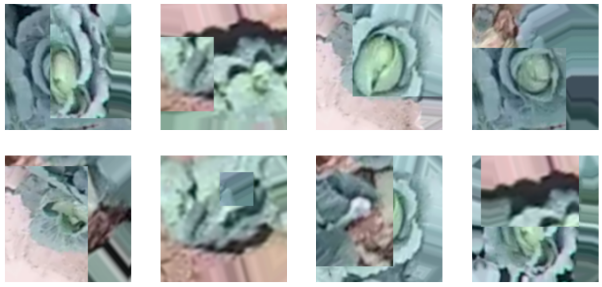

Fig. 3. Augmented training data examples used by the Hybrid model.

Original pictures have 500 pictures of healthy cabbage used for training and validation and 200 pictures of unhealthy cabbage for training and validation. In addition, the test data is 400 cabbage pictures separated from the rest of the data. Since the data originally used for the model was only 700 pictures for training and validation, we increased synthetically the data used for training and validation to 6300 pictures with random rotation, horizontal flipping, changing width and height, zooming in and shearing. The new base of 6300 pictures would remain the same until each of the model applied its own separate version and style of augmentation to the data. Figure 2 is a set of example pictures created to work as training data. CNN, B16 and MviT models in this research used only a simple data augmentation of randomized flipping or turning the image and zooming into it. Turning the image or zooming was set to randomize between 0 and 20%. The hybrid model used CutMix [13] and MixUp [14] for the data augmentation. CutMix and MixUp examples can be seen from Figure 3.

## IV. Results

CNN model constructed for this problem achieved 99.50% accuracy with the test data.

TABLE II
MEAN ACCURACIES, VARIANCE AND TRAINING HYPERPARAMETERS OF THE VIT MODELS USED IN THE RESEARCH.

| Model | B16 | MViT | Hybrid Model |
|---|---|---|---|
| Mean accuracy(%) | 93.15 | 92.45 | 99.13 |
| Variance | $1.90 \times 10^{-4}$ | $3.80 \times 10^{-4}$ | $9.06 \times 10^{-6}$ |
| Image size | $128 \times 128$ | $128 \times 128$ | $128 \times 128$ |
| Batch size | 32 | 32 | 8 |
| Learning rate | $10^{-5}$ to $10^{-6}$ | $10^{-5}$ to $10^{-6}$ | 0 to $5 \times 10^{-4}$ |
| Epochs | 30 | 50 | 15 |
| Patch dimensions | $8 \times 8$ | $4 \times 4$ | $2 \times 2$ |

Models utilizing convolutions in together with transformers handled better the smaller data size used for training and validation compared to the original Vision Transformer B16. The hybrid model with Swin-Transformer outperformed other Vision Transformers in this research. B16 with over 86M parameters gave reliable over 90% results during testing. MobileViT achieved similar results with B16 still being marginally more accurate. These results can be seen from Table 2. Both B16 and MobileViT had more variance compared to Hybrid Swin-Transformer model with the test data. B16 variance was lower than MViT variance and producing more reliability. B16 and MobileViT did not have advanced data augmentation when the Hybrid-Swin Transformer model did. This suggests that both B16 and MobileViT would do better if resources for hyperparameter adjustments and using advanced data augmentation are available.

With these results it can be said that Transformer will be able to solve binary data problems reliably. The results suggests that more advanced data augmentation brings more reliable results with the size of data set used in this research. Every model in this research achieved over 90% accuracy with the test data every single time models were trained. This occurrence is shown on Figure 4. The CNN model still outperformed using a simple data augmentation and significantly less parameters compared to the Vision Transformer models. The results support findings from the original paper [1] that for the B16 to get closer to the CNN model's accuracy, it will need a larger dataset available.

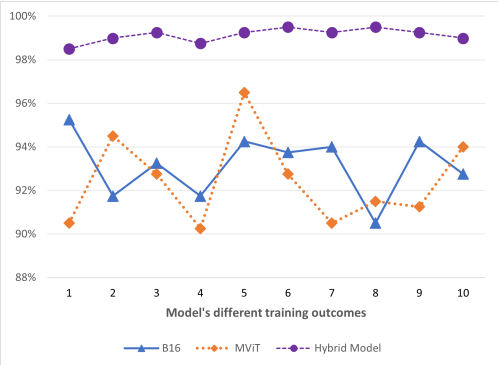


Fig. 4. Accuracies of ViT models after different training periods.

## V. Conclusion

CNN outperformed with a small size of dataset. Vision Transformer are able to solve binary image classification problems. The question rising from these results is: When is it resource efficient to use a Vision Transformer with a smaller dataset? Results suggesting from this research that to achieve excellent results with a Vision Transformer and a small dataset to a binary data problem, advanced data augmentation is important. Vision Transformers with convolutions or CNN elements have the ability of challenging a light CNN model's accuracy with a binary classification problem. Large scale datasets may not be needed in the near future as often anymore [10], [15].

## References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". arxiv, 2020.

[2] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, Ling Shao. "Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions". arxiv, 2021.

[3] Chun-Fu Chen, Quanfu Fan, Rameswar Panda. "CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification". arxiv, 2021.

[4] Sachin Mehta, Mohammad Rastegari. "MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer". arxiv, 2021.

[5] Hazem Hiary, Heba Saadeh, Maha Saadeh, Mohammad Yaqub. "Flower classification using deep convolutional neural networks". IET Computer Vision, 12(6):855-862, 2018. doi: 10.1049/iet-cvi.2017.0155. URL https://doi.org/10.1049/iet-cvi.2017.0155.

[6] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, Baining Guo. "CSWin Transformer: A General Vision Transformer Backbone with Cross-Shaped Windows". arxiv, 2021.

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, Baining Guo. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". arxiv, 2021.

[8] Xinlei Chen, Saining Xie, Kaiming He. "An Empirical Study of Training Self-Supervised Vision Transformers". arxiv, 2021.

[9] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, Shuicheng Yan. "Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet". arxiv, 2021.

[10] Alaaeldin El-Nouby, Gautier Izacard, Hugo Touvron, Ivan Laptev, Hervé Jegou, Edouard Grave. "Are Large-scale Datasets Necessary for Self-Supervised Pre-training?" arxiv, 2021.

[11] Hugo Touvron, Matthieu Cord, Hervé Jégou. "DeiT III: Revenge of the ViT". arxiv, 2022.

[12] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, Lucas Beyer. "Scaling Vision Transformers". arxiv,2021.

[13] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, Youngjoon Yoo. "CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features". arxiv, 2019.

[14] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, David Lopez-Paz. "mixup: Beyond Empirical Risk Minimization". arxiv, 2017.

[15] Zhiying Lu, Hongtao Xie, Chuanbin Liu, Yongdong Zhang. "Bridging the Gap Between Vision Transformers and Convolutional Neural Networks on Small Datasets". arxiv, 2022.