Learning to Pay Multiple Attention with Fully Convolutional Transformers

Samuel Ofosu Mensah[1,2][0000−0002−9290−1206], Bubacarr Bah[1,2][0000−0003−3318−6668], and Willie Brink[2][0000−0002−4081−8232]

[1] African Institute for Mathematical Sciences, Muizenberg, South Africa
{samuelmensah, bubacarr}@aims.ac.za
[2] Division of Applied Mathematics, Stellenbosch University, Stellenbosch, South Africa
wbrink@sun.ac.za

Abstract. Until recently, convolutional neural networks have been the de facto method for computer vision tasks. On the other hand, Transformers have gained popularity in several domains including computer vision. They are known mainly for desired properties including dynamic attention and improved generalisation. Transformers have excellent global representation capabilities but lack the locality inherent to convolutional neural networks. Besides, the global properties in Transformers are desired in convolutional-based tasks. In this study, we use separate fully convolutional Transformers (FCT) modules which take ResNet-50 feature maps as input. A combination of intermediate and final ResNet-50 model feature maps is used to learn global dependencies of the inputs for image classification. In detail, FCT is a modified Transformer which consists of convolutional layers in place of linear layers. As a result, we observe improved results over the baseline model when trained on the CIFAR-10 dataset.

Keywords: CNN · Visual Transformer · Visual Attention.

1 Introduction

Over the years, extensive studies have been done on convolutional neural networks (CNNs) [5]. Overall, they have demonstrated good performance and have dominated the area of computer vision [9,27,25]. With this impressive performance, other domains including natural language processing (NLP) [16,33,38] and speech recognition [7,15,20] have either adapted CNNs to form hybrid models or created novel models consisting entirely of convolutional operators. CNNs are mainly characterised by local connectivity and a shift-invariance property [3]. Even though CNNs have dominated the computer vision space, they lack the ability to learn long-range dependencies due to their poor scaling properties with respect to large receptive fields [21].

Other models without convolutional building blocks have been introduced in the space of computer vision [21,32,28]. A recent algorithm is the Visual Transformer (ViT) model which has attained impressive results for computer vision tasks [4]. Transformers were originally introduced by Vaswani et al. [31], and have become the go-to model for NLP-related tasks [4]. It has since been adapted to computer vision and is now gaining popularity [29]. Some studies have reported the possibility of Transformers replacing CNNs entirely [3]. This is largely due to their dynamic attention properties [36], scalability [4], improved generalisation and long-range capacities [29]. Unfortunately, ViT is computationally heavy as operations grow quadratically according to the number of pixels in an input image [3].

New studies have explored hybrid models which combine convolutions and Transformers for the best of both worlds and resolve previous challenges [36,30,6,37]. To this end, Tragakis et al. [30] introduced the Fully Convolutional Transformer (FCT) module, a modified Transformer model which replaces linear projections with convolutional operators. It works by first extracting long-range dependencies and finally capturing global hierarchical attributes. FCT is characterised by convolutional attention and a wide-focus module. It is also trainable and has demonstrated improved performance by large margins.

Inspired by Jetley et al. [12], we create a concurrent/hybrid model which attaches the FCT module to a ResNet-50 [9]. More precisely, we feed FCT with intermediate representations at different layers together with the feature map of the final convolution block of a ResNet-50 model to classify the CIFAR-10 dataset. We train the model in an end-to-end (without pre-training) approach and observe improved performance. In summary, our main contribution is that we build a hybrid model for image classification.

2 Related Work

Natural Language Processing has enjoyed recent success in deep learning, and this can be mainly attributed to the introduction of Transformers [31]. Transformers have the capability of capturing long-range dependencies with the help of self-attention mechanisms [31,4,36]. Self-attentions are non-local [35,37] in nature and have been successful in several domains including computer vision [31]. In computer vision, some studies have introduced architectures that are only made up of self-attention [21,35,24]. Others have augmented convolutions with self-attention [2].

When a self-attention mechanism is applied to a computer vision task, each pixel of the image attends to every other pixel making self-attention unable to scale for large input sizes [4]. For this reason, Dosovitskiv et al. [4] introduced Vision Transformer (ViT) to efficiently scale realistic input sizes. The success achieved with ViT has sparked significant interest in applying Transformers in computer vision [34]. Since then, some studies have created special cases of ViT [36] and others have created hybrid models by mixing ViT with state-of-the-art CNN backbones [37]. Also, other studies have refined the ViT model by creating a robust version [17] and some have used the design structure of ViT but with multi-layer perceptron (MLP) architecture as the backbone [28].

In our case, we create a hybrid model by passing intermediate feature maps from a ResNet-50 model to Fully Convolutional Transformer (FCT) modules. By using this approach, we encourage early layers of the model to learn similar features of the global image descriptor.

3 Methodology

Our aim is to build a hybrid model that is able to learn long-range dependencies and capture the global features of an image.

3.1 Preliminaries

We consider a dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$, where $\mathcal{X}$ represents the input images and $\mathcal{Y}$ represents the target values. For each image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$, H and W are dimensions of the image and C represents the number of channels of the image. We seek to learn a model that best approximates the true target values. For this study, we achieve our goal by modifying a ResNet-50 model to have auxiliary layers at different layers of the network. Specifically, the auxiliary layers are Fully Convolutional Transformer (FCT) modules, which take in input feature maps from certain layers of the network. By doing so, we expect the model to focus on discriminating regions of the input while paying less attention to regions of less importance.

First, we present background on the various components used in the model. These include ResNet-50, MHSA (Multi-Head Self-Attention), FL (Focus Layer), BN (Batch Normalisation) and LN (Layer Normalisation).

ResNet-50. ResNet-50 inherits its name from residual network with 50 layers. It is characterised by several convolutional layers stacked together as convolutional blocks with skip connections. For an input $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$, a skip connection is defined as

$$\mathbf{x}' = \mathcal{F}(\mathbf{x}) + \mathbf{x}, \tag{1}$$

where $\mathcal{F}(\mathbf{x})$ is the output of a convolutional block and $\mathbf{x}'$ is the output of the skip connection. Also, the number of layers changes depending on the variant of the ResNet model while maintaining the number of blocks at 4. For example, ResNet-50 has 3 convolutional layers in the first convolutional block, 4 convolutional layers in the second convolutional block, 6 convolutional layers in the third convolutional block and another 3 convolutional layers in the final convolutional block.

MHSA. In this study, we linearly transform input $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ to three representations namely a query (Q), key (K) and value (V), using three matrices W_Q, W_K and W_V. It should be noted that these matrices are learnable. The representations Q, K and V are computed as

$$Q = \mathbf{x}W_Q \quad K = \mathbf{x}W_K, \quad V = \mathbf{x}W_V. \tag{2}$$

Next, we define self-attention as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V, \tag{3}$$

where d is a scaling factor. A single self-attention is known as a head. Multi-Head Self-Attention computes several heads (h) in a parallel approach and concatenates the outputs. MHSA is given as

$$\text{MHSA}(Q, K, V) = \textsc{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_n), \tag{4}$$

where n represents the number of heads and $\textsc{Concat}(\cdot)$ is a concatenating function.

FL. The Focus Layer is a feature aggregation layer which applies convolutions to extract fine-grained information from the output of the MHSA output. For input $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$, the focus layer is defined as

$$\text{FL}(\mathbf{x}) = \sigma(\textsc{Conv}(\mathbf{x})), \tag{5}$$

where $\textsc{Conv}(\cdot)$ is a convolutional operator and $\sigma(\cdot)$ is an activation function, which in our case is GELU [10].

BN. & LN. Normalisation as the name suggests is a technique used to normalise the mini-batch or layers of a model to zero mean and unit variance. It is batch normalisation (BN) [11] if applied on a mini-batch and layer normalisation (LN) [1] otherwise. For a sample $x \in \mathbb{R}^d$, normalisation is defined as

$$\text{N}(x) = \frac{x - \mu}{\sigma} \circ \gamma + \beta, \tag{6}$$

where $\mu \in \mathbb{R}$ is the mean and $\sigma \in \mathbb{R}$ is the standard deviation of the feature maps, $\circ$ is an element-wise multiplication, and $\gamma \in \mathbb{R}^d$, $\beta \in \mathbb{R}^d$ are learnable parameters.

3.2 Paying Multiple Attention

Our proposed model is illustrated in Figure 1. In detail, we extract feature maps denoted by $\hat{\mathbf{z}}_l$, where $l \in \{1, \dots, \ell\}$ represents a convolutional layer. Assuming equal dimensions, we add $\hat{\mathbf{z}}_l$ to a global image descriptor $\boldsymbol{g}$ and pass the output to an FCT for attention. A global image descriptor in this case is the output of the penultimate layer of a ResNet-50 model. Finally, the feature maps from the FCT modules are concatenated into a single vector for classification purposes. We use this approach of learning to force earlier layers in the model to learn similar mappings of the global image descriptor of the vanilla model (without attention). We achieve this by using $\hat{\mathbf{z}}_l$ to contribute directly to the classification step [12].

3.3 Fully Convolutional Transformers

The Fully Convolutional Transformer (FCT) module is a special case of the Transformer module (Fig. 2). In FCT, transformations are done using convolutional functions instead of position-wise linear projection for the attention operation inherent in Transformers [36]. The motivation behind using convolutions is to keep local relations between pixels/features while simultaneously maintaining the Transformer structure.

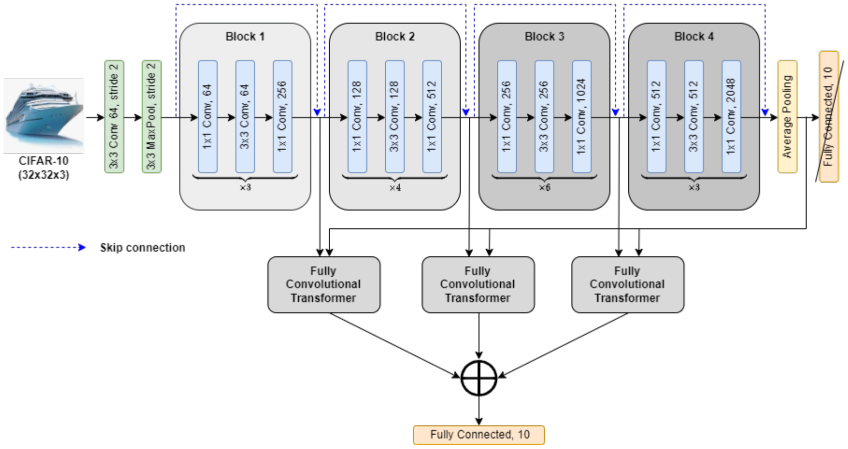


Fig. 1. Illustrating the overall model. Instead of feeding linear classification layers with feature maps of the final block of a backbone model, we first feed intermediate feature maps to FCT modules to capture long-range dependencies, then concatenate the output and later classify.

In our model, the input to the FCT module is a feature map extracted from intermediate layers of the ResNet-50 model. First, we convert the feature maps into overlapping patches using convolution. The generated patches are analogous to tokens in NLP [36]. Next, we feed the generated patches to a depth-wise convolution to generate Q, K, and V. We normalise the outputs and apply MHSA to generate attention. Finally, we fuse the outputs with the patches and feed a normalised resultant to the focus layer which aggregates features using convolution. We summarise FCT mathematically as follows:

$$\mathbf{z}_{l-1} = \textsc{PatchEmbed}(\hat{\mathbf{z}}_{l-1} + \boldsymbol{g}), \tag{7}$$

$$\mathbf{z}_{l} = \mathrm{MHSA}(\mathrm{N}(\textsc{ConvProj}(\mathbf{z}_{l-1}))) + \mathbf{z}_{l-1}, \tag{8}$$

$$\mathbf{z}_{l+1} = \mathrm{FL}(\mathrm{N}(\mathbf{z}_{l})) + \mathbf{z}_{l}, \tag{9}$$

where $\hat{\mathbf{z}}_{l-1}$ is a feature map from an intermediate representation of the network and $\boldsymbol{g}$ is a global image descriptor. $\textsc{PatchEmbed}(\cdot)$ is a convolutional operator used to create patch embeddings. The patch embeddings are used to generate Q, K, and V for MHSA (see Eqn. 4) using $\textsc{ConvProj}$ which is a depth-wise convolution. Before that, Q, K, and V are normalised using either batch normalisation or layer normalisation (see Eqn. 6).

4 Experiments and Results

Experimental Setup. We implement the description of the model in Section 3 and experiment using the CIFAR-10 image dataset [14]. First, we augment the data by random cropping, padding, or flipping the images either horizontally or vertically. With

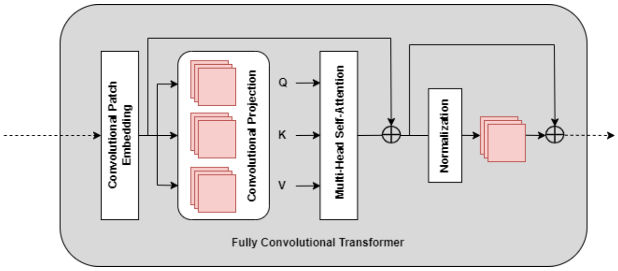

Convolutional Patch Embedding
Convolutional Projection
Q
K
V
Multi-Head Self-Attention
Normalization
Fully Convolutional Transformer

Fig. 2. Details of the Fully Convolutional Transformer (FCT) module. It takes in feature maps from intermediate layers of the backbone network, creates patch embeddings, projects to Q, K, and V for the MHSA mechanism, and feeds to another convolution layer for classification.

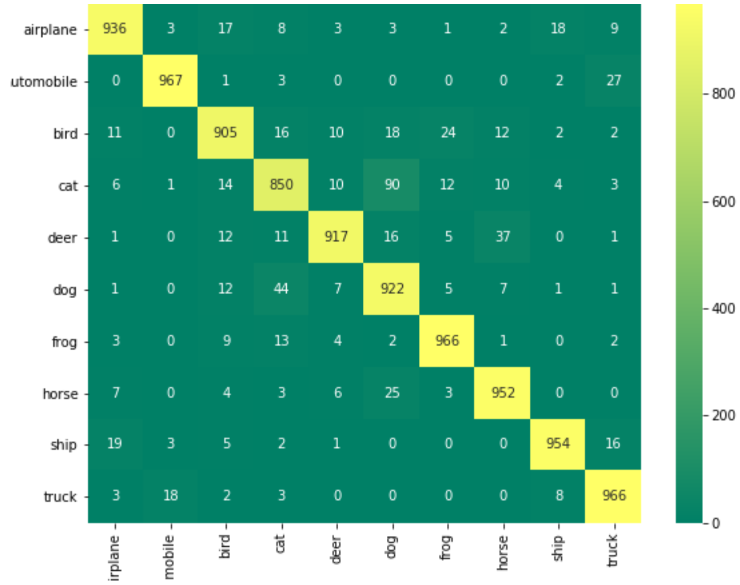
Table 1. Top-1 validation classification accuracy on CIFAR-10 dataset.

Model	Top-1 Acc. (%)
vanilla (ResNet-50)	92.87
ours (with LN)	93.04
ours (with BN)	93.35
ours (pre-trained)	95.72

a batch size of 128, we train the model using the Adam optimizer [13] at a learning rate of 0.01, a weight decay of 1×10^{-4} and cyclical learning rates [26]. We also clip all gradients at global norm 1 [4]. Moreover, we initialise the model using the Kaiming normal initialiser [8] and train end-to-end for 200 epochs. Finally, we use cross-entropy loss as our cost function, and top-1 accuracy to measure performance for the various experiments.

Results. We modified a ResNet-50 model to predict the classes of CIFAR-10 image dataset. In detail, we feed intermediate layers from the ResNet-50 model to a Fully Convolutional Transformer (FCT) module. As a baseline, we train a ResNet-50 with no modifications and achieve 92.87% validation accuracy. Motivated by Wu et al. [36], we experiment with two normalisation techniques: batch normalisation (BN) and layer normalisation (LN). We observe that our model trained with BN performs slightly better than the LN version. To compare, we also train our model initialised with pre-trained weights from ImageNet [22]. We observe a relatively close performance between our model trained from scratch and our model trained with pre-trained weights (see Table 1). Additionally, we predict and use Grad-CAM [23] to generate localisation maps on the input images (see Fig. 5 in the Appendix).

We also generate a confusion matrix (see Fig. 3) from our best-performing model (that is, the model trained from scratch with batch normalisation). We see in Figure 3 that the model struggles to correctly classify the cat category, misclassifying 150 images and mostly classifying them as dogs. This can be partly explained by the visual



	irplane	mobile	bird	cat	deer	dog	frog	horse	ship	truck
airplane	936	3	17	8	3	3	1	2	18	9
utomobile	0	967	1	3	0	0	0	0	2	27
bird	11	0	905	16	10	18	24	12	2	2
cat	6	1	14	850	10	90	12	10	4	3
deer	1	0	12	11	917	16	5	37	0	1
dog	1	0	12	44	7	922	5	7	1	1
frog	3	0	9	13	4	2	966	1	0	2
horse	7	0	4	3	6	25	3	952	0	0
ship	19	3	5	2	1	0	0	0	954	16
truck	3	18	2	3	0	0	0	0	8	966

Fig. 3. The confusion matrix of our model on the CIFAR-10 validation set.

similarity that exists between cats and dogs. As support to this claim, we see that several dogs are misclassified as cats.

Furthermore, we visualise attention maps of the various auxiliary layers used in our model. Since we add a global image descriptor to intermediate feature maps, we expect certain regions of the output to have high values if they contain similar or parts of dominating regions of the global image descriptor. We observe that the earliest layer (that is the first FCT) produces coarse localisations on the discriminative regions. We see that as we progress through the network, the model produces finer localisations and is more focused on the object of interest (Fig. 4).

5 Conclusion

In this study, we attempted to learn long-range dependencies and capture global features using a modified ResNet-50 which outputs feature maps from intermediate layers to Fully Convolutional Transformer (FCT) modules. We trained the model in an end-to-end fashion and observed superior performance over the vanilla model (ResNet-50 with no FCT modules) used for the study. Also, we observed the benefit of training the model with batch normalisation over layer normalisation. Finally, we saw that our model is able to highlight discriminating regions of the input image, generating coarse localisations to fine localisations as it progresses through the layers. Overall, we demonstrated the potential of the proposed model and thus have provided a new perspective for the future design of hybrid models containing convolutional blocks and Transformers.

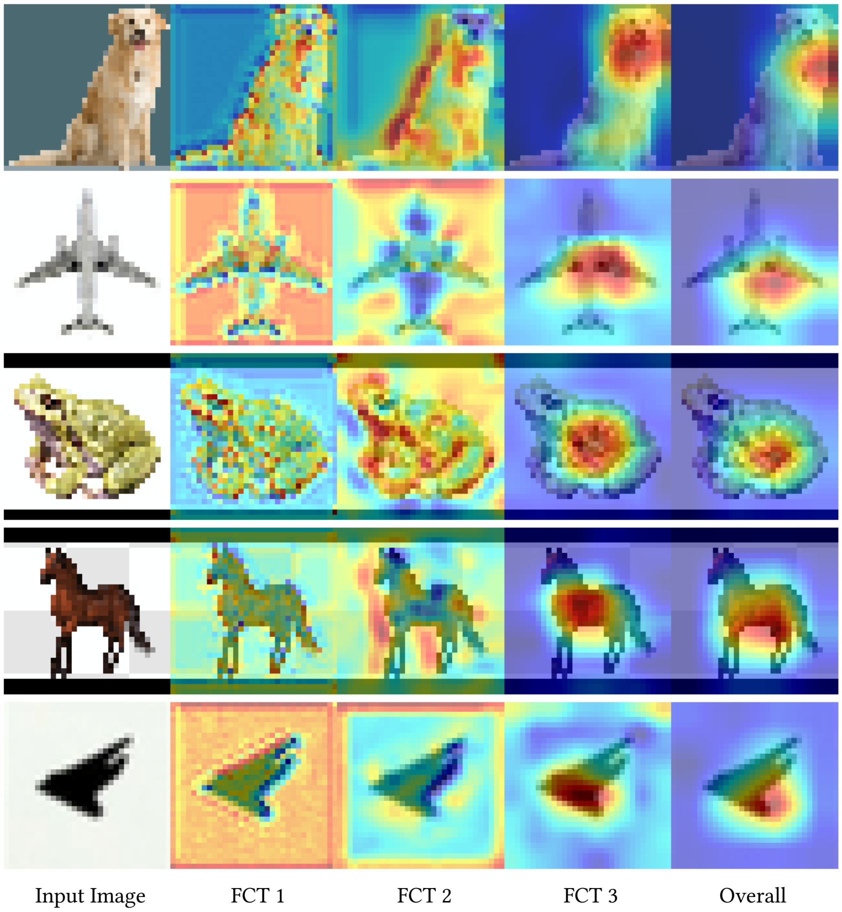


Fig. 4. Discriminating regions of randomly selected images using Grad-CAM.

Acknowledgments. We thank the German Academic Exchange Service (DAAD) and the State Ministry of Research and the Arts for support of this research. We are also grateful to the Centre for High Performance Computing (CHPC) for providing us with computing resource for this research.

Appendix

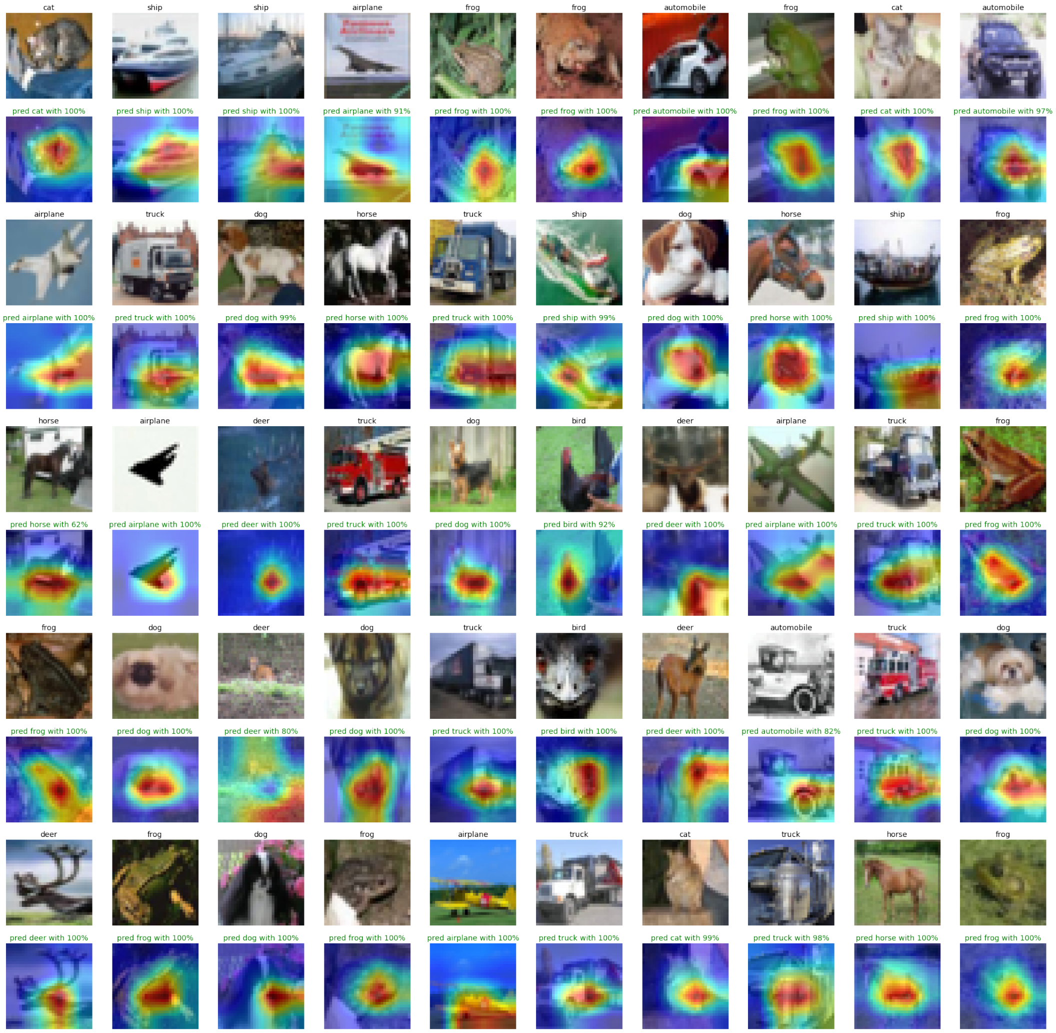


Fig. 5. We predict and highlight discriminating regions of a subset of the validation set of CIFAR-10. We observe high confidence the predictions. It should be noted that we rounded the confidence values to the nearest integer percentages.

References

1. Ba, J. L., Kiros, J. R., Hinton, G. E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
2. Bello, I., Zoph, B., Vaswani, A., Shlens, J., Le, Q. V.: Attention augmented convolutional networks. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 3286-3295) (2019)
3. Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., ..., Shen, C.: Twins: Revisiting the design of spatial attention in vision transformers. Advances in Neural Information Processing Systems, 34, 9355-936 (2021)
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... , Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
5. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ..., Chen, T.: Recent advances in convolutional neural networks. Pattern recognition, 77, 354-377 (2018)
6. Gulati, A., Qin, J., Chiu, C. C., Parmar, N., Zhang, Y., Yu, J., ..., Pang, R.: Conformer: Convolution-augmented transformer for speech recognition. arXiv preprint arXiv:2005.08100 (2020)
7. Han, W., Zhang, Z., Zhang, Y., Yu, J., Chiu, C. C., Qin, J., ..., Wu, Y.: Contextnet: Improving convolutional neural networks for automatic speech recognition with global context. arXiv preprint arXiv:2005.03191 (2020)
8. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision (pp. 1026-1034) (2015)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778) (2016)
10. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
11. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). PMLR (2015)
12. Jetley, S., Lord, N. A., Lee, N., Torr, P. H.: Learn to pay attention. arXiv preprint arXiv:1804.02391 (2018)
13. Kingma, D. P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
14. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. (2009)
15. Kubanek, M., Bobulski, J., Kulawik, J.: A method of speech coding for speech recognition using a convolutional neural network. Symmetry, 11(9), 1185 (2019)
16. Li, P., Mao, K.: Knowledge-oriented convolutional neural network for causal relation extraction from natural language texts. Expert Systems with Applications, 115, 512-523 (2019)
17. Mao, X., Qi, G., Chen, Y., Li, X., Duan, R., Ye, S., ... Xue, H.: Towards robust vision transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 12042-12051) (2022)
18. Nair, V., Hinton, G. E.: Rectified linear units improve restricted boltzmann machines. In ICML (2010)
19. Park, N., Kim, S.: How Do Vision Transformers Work?. arXiv preprint arXiv:2202.06709 (2022)
20. Passricha, V., Aggarwal, R. K.: Convolutional neural networks for raw speech recognition (pp. 21-40). IntechOpen (2018)

21. Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., Shlens, J.: Stand-alone self-attention in vision models. Advances in Neural Information Processing Systems, 32 (2019)
22. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., … , Fei-Fei, L.: Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3), 211-252 (2015)
23. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE international conference on computer vision (pp. 618-626) (2017)
24. Shen, Z., Bello, I., Vemulapalli, R., Jia, X., Chen, C. H.: Global self-attention networks for image recognition. arXiv preprint arXiv:2010.03019 (2020)
25. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
26. Smith, L. N., Topin, N.: Super-convergence: Very fast training of residual networks using large learning rates (2018)
27. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … , Rabinovich, A.: Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9) (2015)
28. Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., … , Dosovitskiy, A.: Mlp-mixer: An all-mlp architecture for vision. Advances in Neural Information Processing Systems, 34, 24261-24272 (2021)
29. Touvron, H., Cord, M., Jégou, H.: Deit iii: Revenge of the vit. arXiv preprint arXiv:2204.07118 (2022)
30. Tragakis, A., Kaul, C., Murray-Smith, R., Husmeier, D.: The Fully Convolutional Transformer for Medical Image Segmentation. arXiv preprint arXiv:2206.00566 (2022)
31. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … , Polosukhin, I.: Attention is all you need. Advances in neural information processing systems, 30 (2017)
32. Wang, H., Zhu, Y., Green, B., Adam, H., Yuille, A., Chen, L. C.: Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In European Conference on Computer Vision (pp. 108-126). Springer, Cham (2020, August)
33. Wang, W., Gang, J.: Application of convolutional neural network in natural language processing. In 2018 International Conference on Information Systems and Computer Aided Education (ICISCAE) IEEE (pp. 64-70) (2018)
34. Wang, W., Xie, E., Li, X., Fan, D. P., Song, K., Liang, D., … , Shao, L.: Pvt v2: Improved baselines with pyramid vision transformer. Computational Visual Media, 8(3), 415-424 (2022)
35. Wang, X., Girshick, R., Gupta, A., He, K: Non-local neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7794-7803) (2018)
36. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 22-31) (2021)
37. Yan, H., Li, Z., Li, W., Wang, C., Wu, M., Zhang, C.: ConTNet: Why not use convolution and transformer at the same time?. arXiv preprint arXiv:2104.13497 (2021)
38. Yin, W., Kann, K., Yu, M., Schütze, H.: Comparative study of CNN and RNN for natural language processing. arXiv preprint arXiv:1702.01923 (2017)