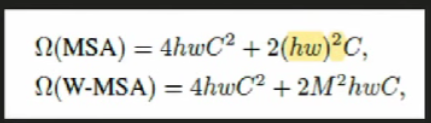
**Literature Review: A ConvNet for the 2020s**

The paper “A ConvNet for the 2020s” has been written with the goal of modernizing our Convolutional Networks as an attempt to rival the state of the art performance of vanilla ViTs as well as Hierarchical Transformers such as the Swin. CNNs were popularly the gold standard of vision related tasks until the 2020s. The paper references the ResNet50/ResNet200 models as main examples of the dominance of CNNs in vision related tasks. The 2020s however, brought the introduction of the Vision Transformer(ViT). This performed much better on image classification but failed to do as well on object detection as well as segmentation tasks. We then move on to the Swin Transformers, which introduced some ConvNet priors, making it more versatile in vision related tasks. This was attributed to the sliding window strategy, which was reintroduced as it is intrinsic to visual processing, especially high-resolution images.



Quadratic complexity of ViT with respect to hw vs Linear complexity of Swin with respect to hw (quadratic complexity with respect to M, which is generally a 7x7 window)

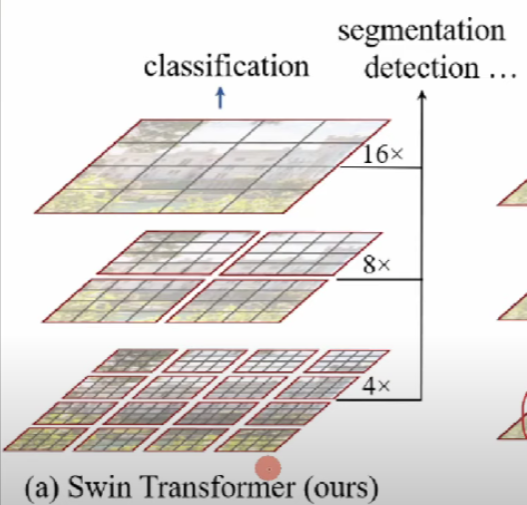
The paper expresses the need of a model which can prove to be the face of all CV related tasks. Although the Swin transformers showed revolutionary results in most computer vision tasks, the computational power the transformer requires makes scalability of such a model an issue. The writers of the paper draw inspiration from the Swin transformers, due to the incorporation of their sliding window strategy. The main problem the paper tries to tackle is the lack of comparable performances of pure ResNets with respect to the ViT or Swin. It tries to bridge the gap between pre-ViT and post-ViT eras for CNNs. By proposing a family of pure CNNs(ConvNeXts) that achieve comparable performance with Swin, it can take over as the face of all CV related tasks, while maintaining the efficiency of CNNs.

Experimentation with newer versions of the ResNet have been tried in the past. The paper mentions in its Conclusion that the model itself is not completely new, as many of its design choices have been examined separately over the last decade. A good example of this would be the link provided by [76], the paper: “ResNet strikes back”, which incorporates the use of the AdamW optimizer, RRC, Mixup, Cutmix and RandAugment as data augmentation techniques, Label Smoothing and weight decay as regularization techniques. The use of depthwise convolution was also inspired by its popular use in the AlexNet model.

The approach taken by this paper is significantly different from most other existing literature on the matter. Unlike other papers where experiments are performed and then based on the results, inferences are drawn about features of a model, here we simply follow a procedural approach to make our ResNet50 model similar to the Swin-T architecture. The paper claims and backs up the statement that transformers are to be adopted as a generic backbone for vision related tasks to achieve state of the art performance. It also believes that the essence of convolutions, which was captured by the Swin model still remains much desired in the filed of CV. Keeping these factors in consideration, the ResNet50 model is modified in its training approaches to its micro design features. It is important to note that the following approach is one of its kind in the way that techniques used by transformers are incorporated into the more efficient ConvNets. The writers claim that we can get the best of both worlds as both ConvNets and hierarchical transformers have similar inductive biases. This means that both the models generalise beyond just the training data it sees in a similar manner.

The paper performs a bunch of experiments to prove their central claim that design decisions used in Transformers can positively impact our CNNs performance. Each stage of experimentation is backed up by statistical data including the improvement of the model’s performance across various tasks in CV. The new CNNs, now called ConvNeXts, were evaluated over the following tasks:- ImageNet classification, object detection/segmentation on COCO and semantic segmentation on ADE20K. In order to modernize our ResNet, changes were made in the way we trained our model as well as in the Macro and Micro design.

The ResNeXt is trained using a strategy close to that of DeiT and Swin. Training is done for 300 epochs instead of 90 and various data augmentation techniques like Mixup, Cutmix, RandAugment amd Random erasing are used. The AdamW optimizer is introduced and Stochastic Depth and Label Smoothing are used as regularization techniques. This gives the originally 76.1% accurate ResNet50 a 2.7% boost.

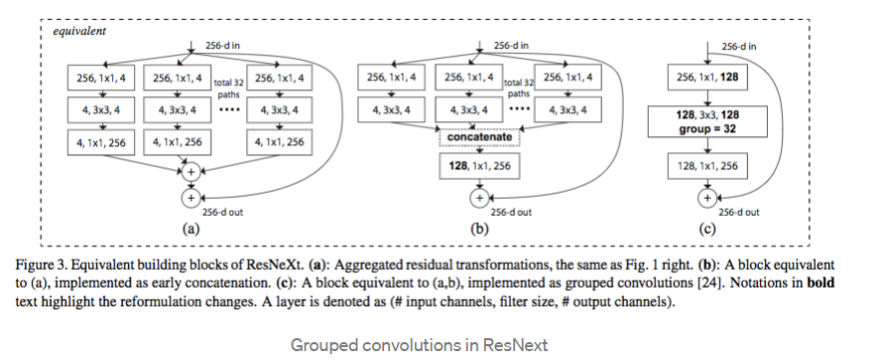
Swin-T followed a compute ratio of 1:1:3:1. To imitate this, the number of blocks in each stage of ResNet50 is changed from (3,4,6,3) to (3,3,9,3). This compute ratio gives us a 0.6% boost. The standard stem cell in ResNet50 containing 7x7 kernels with stride 2 are replaced by a non-overlapping convolution technique, which is again very similar to the Shifted WINdows we find in SWIN as shown in the figure below:-

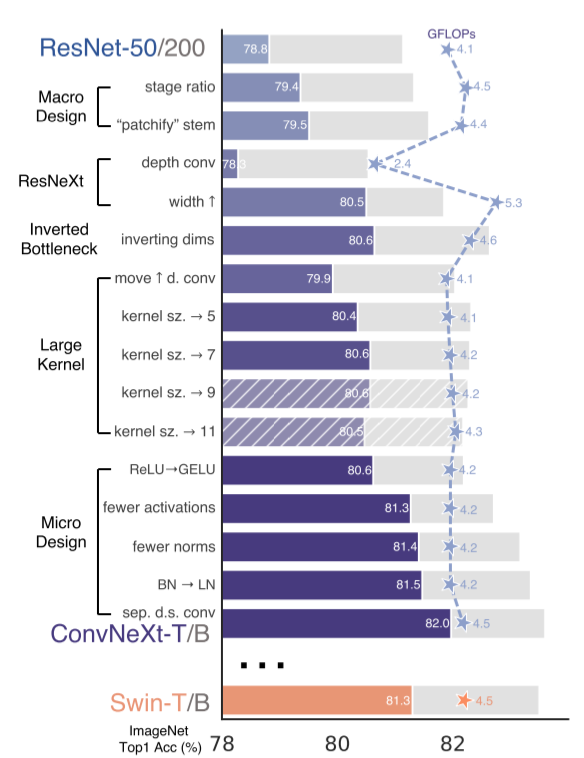
In order to localise our self-attention with the goal of reducing the quadratic complexity, 16x16 windows are broken into 16 4x4 local windows

This pattern is closely replicated by our ResNeXt, by using 4x4 kernels with a stride of 4(‘patchify system’). This gives us a 0.1% boost.

Next, the network width is increased to 96(same no. of channels as the Swin-T) with the use of depthwise convolutions, popularized by the AlexNet. This is done by using multiple diverging kernels and running parallel backpropagations. They are concatenated at the end of the block. The inverted bottleneck design is also incorporated, with a ratio of 4 and the use of 1x1 kernels. Larger 7x7 kernels depthwise convolutions are used in each block, which are similar to those used in Swin.

These combined get the accuracy up to 80.6%.

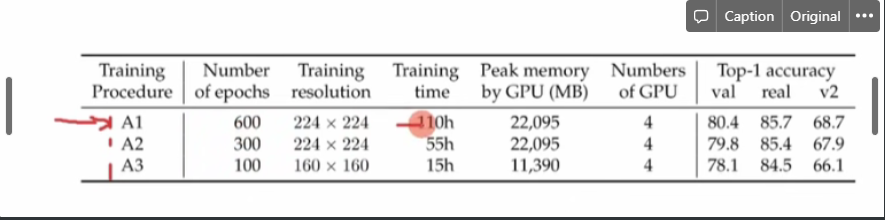
Grouped convolutions enable efficient model parallelism, so much so that AlexNet was trained on GPUs with only 3GB RAM.

In terms of Micro changes, the model notably replaces ReLU with GELU(Gaussian Error Linear Unit). Fewer activation functions and fewer normalization layers are used, following the footsteps of Transformers. A single GELU is used in each block. Batch Normalization is replaced by Layer Normalization. Furthermore, with the use of separate spatial downsampling layers , we can improve the accuracy of the model to 82.0%, which beats Swin-T's 81.3%.

Summed up: Modernization of ResNet50

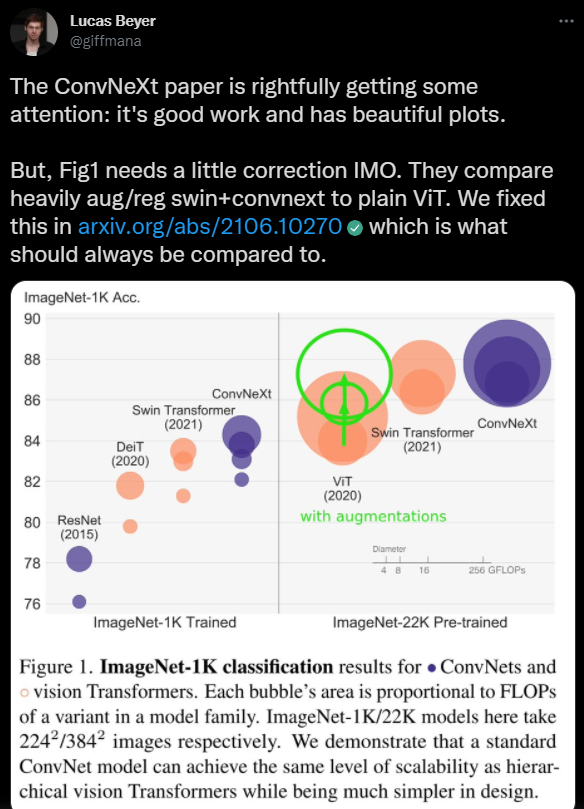
The evaluation of various ConvNeXt variants over the ImageNet, COCO and ADE20K whilst maintaining better results approximately same FLOPS than the Swin served as the perfect experiments to prove their central claim that pure ConvNets can maintain their efficiency while matching the performance of state of the art Hierarchical Transformers such as Swin across all major bechmarks.

The paper makes a strong argument in favour of ConvNets still being a relevant force in the field of CV with its results. However, the paper falls short when it comes to flexibility of approach. There are a few such instances of this in the paper. The paper credits another paper [76] “ResNet strikes back”, and mentions how its modern training techniques significantly enhance the performance of ResNet50, particularly its A1 strategy:-

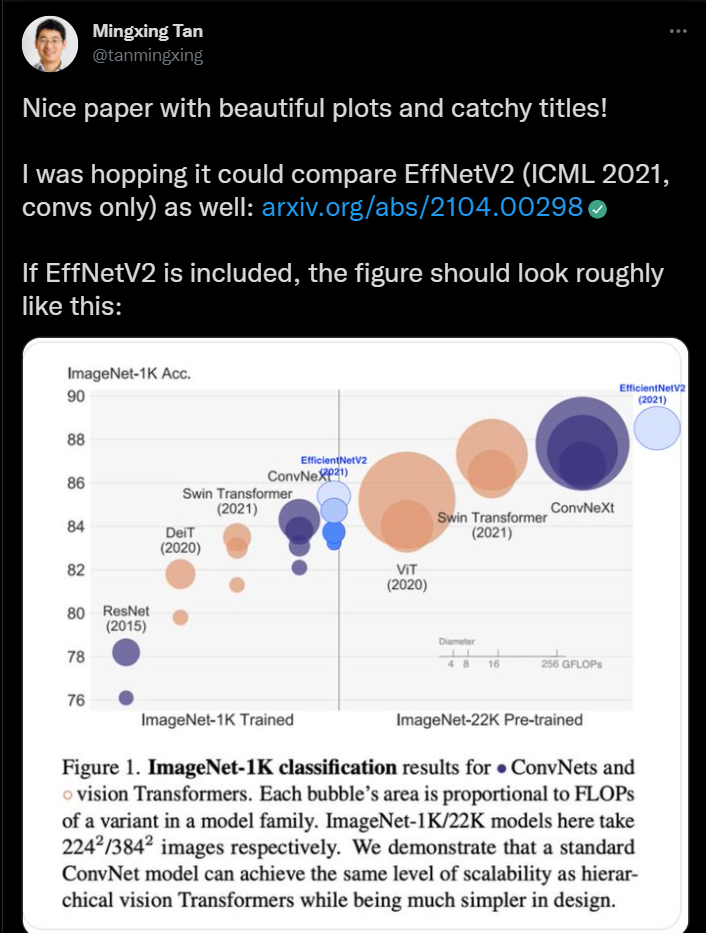
A small screenshot taken from the paper: “ResNet strikes back”

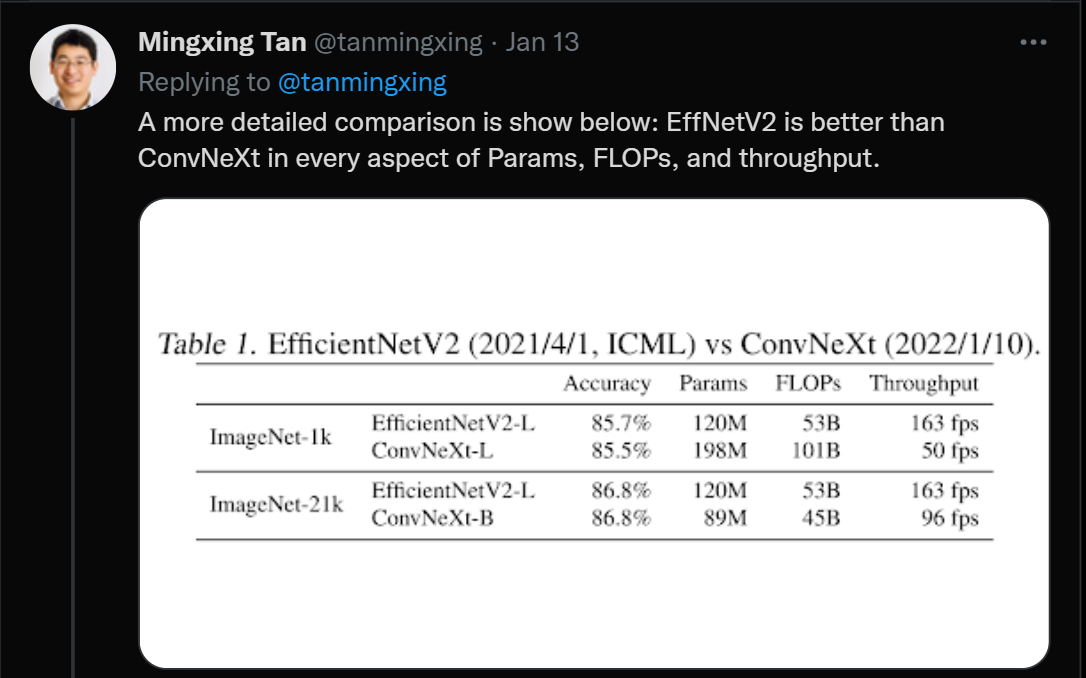
Here we can clearly see that the A1 training strategy yields a top-1 accuracy of 80.4% on ImageNet. Despite this being a extremely significant improvement from its original 76.1%, the paper still follows a training recipe that is close to DeiT’s as well as Swin’s, which only yields an accuracy of 78.8%. Thus, to maintain its uniformity of following the priors of Transformers, it is giving up a possibility of an even better final accuracy after the incorporation of Macro and Micro changes in the model. This is one way I would definitely build on the paper’s rigid approach.

Another problem faced by the paper is its slightly inaccurate or outdated representation of data. The following is a tweet by Lucas Beyer, the creator of ViT, in which he states that vanilla ViT was compared unfairly against heavily augmented and regularized Swin and ConvNeXt models. He provides a rectified representation of the ViT:-



From this graph, we can also see that the ResNet has been represented with its 2015 version, which has since undergone major improvements. It also doesn’t mention the EffNetV2, which is better than the ConvNeXt in every aspect of parameters, FLOPs and throughput. The following tweets by Mingxing Tan(creator of EffNetV2), put things into perspective:-





Thus, the paper’s failure to acknowledge better, improved approaches of models lead to loss of possibilities of better outcomes. To build upon the paper’s approach, there is a need to draw inspiration from not only the Swin way of doing things, but also from currently existing superior approaches such as the EfficientNetV2, which outperforms ConvNeXt in most aspects. If we forego the rigidity of considering Swin as our generic vision backbone, it would lead us through a gate of many more interesting possibilities.