Transformers for Computer Vison: A critical review

Ruthwik Ganesh (230702930)
ec23759@qmul.ac.uk

***Abstract*—The remarkable achievements of Transformer models in natural language tasks have prompted the vision community to investigate their applicability to computer vision challenges. The objective of this survey is to furnish a comprehensive overview of Transformer models within the realm of computer vision, elucidating their impact and potential applications in this discipline.**

I. Introduction

The advancements achieved by Transformer networks in the field of Natural Language Processing (NLP) have generated considerable enthusiasm within the computer vision community, prompting a keen interest in the adaptation of these models for applications in vision and multi-modal learning tasks. The distinctive characteristics inherent in visual data, of spatial and temporal coherence, necessitate innovative network architectures and training methodologies. Consequently, Transformer models and their derivatives have demonstrated notable success across various applications in the realm of computer vision. These applications encompass but are not limited to image recognition [1], [2], object detection [3], [4], segmentation [5], image super-resolution [6], video understanding [7], [8], image generation [9], text-image synthesis [10], and visual question answering [11], [12]. This survey endeavours to encompass the latest and noteworthy research within the computer vision domain, offering a point of reference for readers who are keenly interested in the subject matter, we'll understand Vision transformer and its variants for several application. Subsequently, a deep dive into Object Detection and Medical Image segmentation using the same. Furthermore, this discourse will outline the current challenges using vision transformers and discuss methods to mitigate them.

II. Vision Transformer

Vision Transformer (ViT) [1] is a pure transformer directly applies to the sequences of image patches for image classification task. It follows transformer's original design as much as possible. To address 2D images, the representation of the image, denoted as $X \in R^{h \times w \times c}$ is reconfigured into a series of flattened 2D patches $Xp \in R^{n \times (p\^2 \times c)}$ where c represents the number of channels. Here, *(h,w)* signifies the resolution of the original image, while *(p,p)* denotes the resolution of each image patch. The resultant sequence length for the transformer is articulated as $n=hw/p^2$ *(Fig 3)*. Given that the transformer maintains constant widths across its layers, a trainable linear projection is employed to map each vectorized patch to the model dimension d, and the output is denoted as patch embeddings.

A trainable embedding is employed on the sequence of embedding patches, with the state of this embedding serving as the representation of the image *(Fig 4)*. Throughout both the pre-training and fine-tuning stages, the classification heads are affixed to the same dimensionality. Additionally, 1D position embeddings are incorporated into the patch embeddings to preserve positional information. It is noteworthy that the Vision Transformer (ViT) exclusively employs the standard transformer's encoder, except for the layer normalization placement, and the output of this encoder is subsequently processed by a Multi-Layer Perceptron head.

III. Applications of Vision Transfomer

Usually, Vision Transformer (ViT) undergoes pre-training on extensive datasets and subsequently undergoes fine-tuning for specific downstream tasks using smaller datasets. ViT demonstrates moderate performance outcomes when trained on intermediate-sized datasets, such as ImageNet, achieving accuracies slightly below those of ResNets with comparable dimensions *(Fig 12)*. This is attributed to the inherent absence of certain inductive biases in transformers, such as translation equivariance and locality, leading to challenges in generalization when trained with limited data volumes.

The initial vision transformer excels in capturing long-range dependencies among patches but tends to overlook local feature extraction, as the 2D patch is transformed into a vector using a straightforward linear layer. Recent attention from researchers has been directed towards enhancing the modeling capacity for local information [13], [14], [15]. Notably, TNT [13] takes a step further by subdividing the patch into multiple sub-patches and introducing a novel transformer-in-transformer architecture. This design incorporates an inner transformer block to characterize the relationships among sub-patches and an outer transformer block for information exchange at the patch level *(Fig 7)*. Swin Transformers *(Fig 8)*, as delineated in [14] and [16], implement localized attention within a defined window and introduce a novel shifted window partitioning *(Fig 9)* approach to facilitate cross-window connections. In a similar vein, the Shuffle Transformer, in [17] and [18], leverages the spatial shuffle operation as an alternative to shifted window partitioning, thereby enabling efficient cross-window connections. KVT [19] introduces a k-NN attention mechanism to leverage the locality of image patches. This approach selectively computes attentions only with the top-k similar tokens, thereby effectively disregarding noisy tokens in the process. The optimization of future methodologies should particularly focus on enhancing the computational efficiency and attention precision of the self-attention mechanism.

There persist discernible performance gaps between transformers and well-established CNN architecture. A primary contributing factor may stem from the inherent limitation in extracting local information. Despite enhancements introduced in certain variants of ViT to address this issue, a more straightforward avenue involves using transformer with convolution to seamlessly incorporate locality into the conventional transformer structure.

Object Detection: Object detection methods utilizing Transformer architectures can be broadly classified into two primary categories: transformer-based set prediction methods [3, 4, 20, 21, 22] and transformer-based backbone methods [23, 24] *(Fig 5)*. DETR (detection transformer), a simple and fully end-to-end object detector, treats the object detection

task as an intuitive set prediction problem, it starts with a CNN backbone to extract features from the input image, fixed positional encodings are added to the flattened features before the features are fed into the encoder-decoder transformer *(Fig 6)*. The decoder consumes the embeddings from the encoder along with N learned positional encodings (object queries) and produces N (predefined parameter where, N > no. of objects) output embeddings. Simple feed-forward networks (FFNs) are used to compute the final predictions, which include the bounding box coordinates and class labels. Unlike the original transformer, which computes predictions sequentially, DETR decodes N objects in parallel. DETR employs Hungarian loss and a bipartite matching algorithm to assign the predicted and ground-truth objects. DETR shows impressive performance on object detection, delivering comparable accuracy and speed with the popular and well-established Faster R-CNN [25] baseline on COCO benchmark. DETR poses several challenges, specifically, longer training schedule and poor performance for small objects.

Deformable DETR, presented in [33], seeks to address the challenges of slow convergence and limited spatial resolution faced by DETR due to Transformer attention modules. This approach introduces a deformable attention module, mitigating the slow convergence and high complexity of DETR without relying on Feature Pyramid Networks (FPN). Unlike DETR's use of multi-scale features, which increases complexity and training time, Deformable DETR selectively focuses attention on positions with more local information for each query. This strategy, facilitated by discriminative networks, reduces computational demands. Moreover, the Deformable Attention Module extends to multi-scale feature maps, effectively handling issues related to small objects. In contrast to DETR, Deformable DETR adopts a refined approach wherein, for each query, attention is selectively directed towards positions deemed to encompass more pertinent local information. This strategic focus is achieved through the incorporation of more discriminative networks, alleviating the computational burden posed by expansive feature graphs. Furthermore, the Deformable Attention Module is expansively applied to multi-scale feature maps, effectively addressing challenges associated with small objects.

Up-DETR, introduced by Dai et al. [34], leverages the success of pre-training transformers in natural language processing for object detection. The method entails a pretext task known as "random query patch detection," where patches randomly cropped from an image serve as queries for the decoder. The model is pre-trained to detect these query patches within the original image. UP-DETR demonstrates effective transferability, excelling in one-shot detection and panoptic segmentation. Ablation studies underscore the importance of freezing the pre-training Convolutional Neural Network (CNN) backbone to preserve feature discrimination. In contrast to DETR, UP-DETR adopts a unique approach, maintaining the frozen CNN backbone while introducing a patch feature reconstruction branch. This branch is jointly optimized with patch detection, offering a distinctive strategy to balance classification and localization preferences in the pretext task.

Medical Image Segmentation: Cao et al. [26] introduced a Transformer-based architecture, akin to Unet, designed explicitly for medical image segmentation. The methodology involves tokenizing image patches and subsequently inputting them into a U-shaped Encoder-Decoder structure *(Fig 10)* based on the Transformer architecture. This model incorporates skip-connections to facilitate local-global semantic feature learning. In a similar vein, Valanarasu et al. [27] delved into transformer-based approaches for medical image segmentation tasks. They examined the feasibility of employing transformer-based network architectures and proposed a Gated Axial-Attention model *(Fig 11)*, extending existing architectures by introducing an additional control mechanism within the self-attention module. Additionally, Cell-DETR [28], derived from the DETR panoptic segmentation model, represents an endeavour to utilize transformers for cell instance segmentation. This approach integrates skip connections to establish links between features in the backbone Convolutional Neural Network (CNN) and the CNN decoder in the segmentation head, aiming to enhance feature fusion. Remarkably, Cell-DETR attains state-of-the-art performance in cell instance segmentation from microscopy imagery.

IV. Challenges

The generalization and robustness of transformers in the context of computer vision pose notable challenges. In contrast to Convolutional Neural Networks (CNNs), pure transformer models exhibit a lack in certain inductive biases and heavily depend on extensive datasets for large-scale training [1]. Therefore, the efficacy of transformers in terms of generalization and robustness is markedly influenced by the quality of the training data. While Vision Transformer (ViT) demonstrates notable effectiveness in downstream image classification tasks like CIFAR [29] and VTAB [30], its direct application as a backbone for object detection has yielded results inferior to those achieved by Convolutional Neural Networks (CNNs) [29]. Considerable progress is required to enhance the generalization capabilities of pre-trained transformers for a broader spectrum of visual tasks. Position embeddings are introduced into image patches to preserve crucial positional information, a significant consideration in computer vision tasks. Drawing inspiration from the extensive utilization of parameters in transformers, the concept of over-parameterization [31], [32] emerges as a potential avenue for enhancing the interpretability.

V. Conclusion

One area of focus for exploration pertains to assessing the effectiveness and efficiency of transformers in the realm of computer vision. The objective is to develop vision transformers that are both highly effective and resource-efficient, encompassing high performance and low resource costs. The performance aspect is indicative of the model's applicability in real-world scenarios, while the resource cost directly impacts deployment on devices [35], [36]. Given the inherent correlation between effectiveness and efficiency, exploring methods to achieve a more optimal balance between these factors emerges as a meaningful avenue for further investigation.

REFERENCES

[1] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.C.

[2] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. J´egou, "Training data-efficient image transformers & distillation through attention," arXiv preprint arXiv:2012.12877, 2020.

[3] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," arXiv preprint arXiv:2005.12872, 2020.

[4] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection,"arXiv preprint arXiv:2010.04159, 2020.

[5] L. Ye, M. Rochan, Z. Liu, and Y. Wang, "Cross-modal selfattention network for referring image segmentation," in CVPR, 2019.

[6] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in CVPR, 2020.

[7] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A joint model for video and language representation learning," in ICCV, 2019.

[8] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in CVPR, 2019.

[9] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, and W. Gao, "Pre-trained image processing transformer," arXiv preprint arXiv:2012.00364, 2020.

[10] A. Ramesh, M. Pavlov, G. Goh, and S. Gray, "DALL·E: Creating images from text," tech. rep., OpenAI, 2021.

[11] H. Tan and M. Bansal, "LXMERT: Learning cross-modality encoder representations from transformers," in EMNLP-IJCNLP, 2019.

[12] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "VL-BERT:Pre-training of generic visual-linguistic representations," arXiv preprint arXiv:1908.08530, 2019.

[13] K. Han et al., "Transformer in transformer," in Proc. Conf. Neural Informat. Process. Syst., 2021. https://proceedings.neurips.cc/

[14] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in Proc. Int. Conf. Comput. Vis., 2021, pp. 10012–10022.

[15] C.-F. Chen et al., "Regional-to-local attention for vision transformers," 2021, arXiv:2106.02689.

[16] X. Dong et al., "CSWin transformer: A general vision transformer backbone with cross-shaped windows," 2021, arXiv:2107.00652.

[17] Z. Huang et al., "Shuffle transformer: Rethinking spatial shuffle for vision transformer," 2021, arXiv:2106.03650.

[18] J. Fang et al., "MSG-transformer: Exchanging local spatial information by manipulating messenger tokens," 2021, arXiv:2105.15168.

[19] P. Wang et al., "KVT: K-NN attention for boosting vision transformers," 2021, arXiv:2106.00515.

[20] Z. Sun et al., "Rethinking transformer-based set prediction for object detection," in Proc. Int. Conf. Comput. Vis., 2021, pp. 3611–3620.

[21] M. Zheng et al., "End-to-end object detection with adaptive clustering ransformer," in Proc. Brit. Mach. Vis. Assoc., 2021.

[22] T. Ma et al., "Oriented object detection with transformer," 2021, arXiv:2106.03146.

[23] J. Beal et al., "Toward transformer-based object detection," 2020, arXiv:2012.09958.

[24] X. Pan et al., "3D object detection with pointformer," in Proc. Conf. Comput. Vis. Pattern Recognit., 2021, pp. 7463–7472.

[25] S. Ren et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," in Proc. Conf. Neural Informat. Process. Syst., 2015, pp. 91–99.

[26] H. Cao et al., "Swin-Unet: Unet-like pure transformer for medical image segmentation" 2021, arXiv:2105.05537.

[27] J. M. J. Valanarasu et al., "Medical transformer: Gated axial-attention for medical image segmentation," in Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv., 2021, pp. 36–46.

[28] T. Prangemeier et al., "Attention-based transformers for instance segmentation of cells in microstructures," in Proc. Int. Conf. Bioinf. Biomed., 2020, pp. 700–707.

[29] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Master's thesis, Univ. Tront, 2009.

[30] X. Zhai et al., "A large-scale study of representation learning with the visual task adaptation benchmark," 2019, arXiv:1910.04867.

[31] R. Livni et al., "On the computational efficiency of training neural networks," in Proc. Conf. Neural Informat. Process. Syst., 2014, pp. 855–863.

[32] B. Neyshabur et al., "Towards understanding the role of overparametrization in generalization of neural networks," in Proc. Int. Conf. Learn. Representations, 2019.

[33] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, Jifeng Dai, DEFORMABLE DETR: DEFORMABLE TRANSFORMERS FOREND-TO-END OBJECT DETECTION In arXiv preprint arXiv:2010.04159,2010.

[34] Zhigang Dai1, Bolun Cai, Yugeng Lin, Junying Chen, UP-DETR: Unsupervised Pre-training for Object Detection with Transformers In arXiv preprint arXiv: 2011.09094,2011.

[35] T. Chen et al., "DianNao: A small-footprint high-throughput accelerator for ubiquitous machine-learning," ACM SIGARCH Comput. Architect. News, vol. 42, pp. 269–284, 2014.

[36] H. Liao et al., "DaVinci: A scalable architecture for neural network computing," in Proc. IEEE Hot Chips Symp., 2019, pp. 1–44.

Appendix

Fig 1. Transformer architecture (Vaswani et al., 2017)

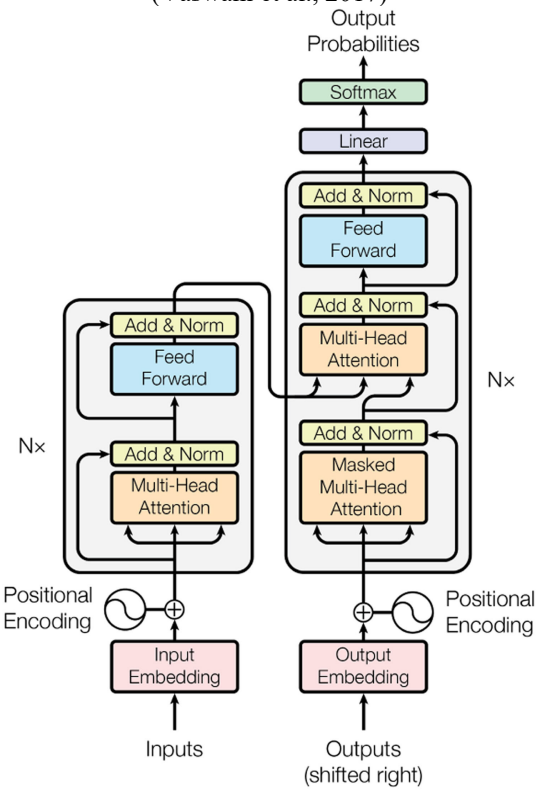

Fig 2. Multi-head attention & scaled dot product attention (Vaswani et al., 2017)

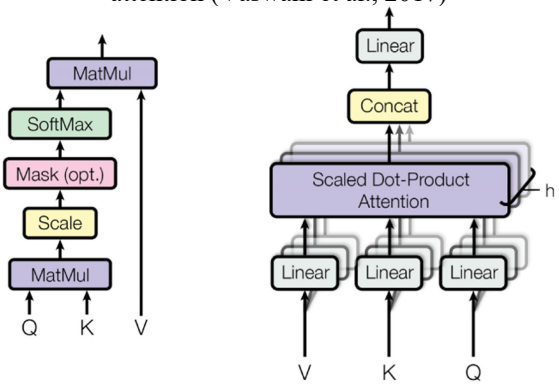

Fig 3. Vison Transformer architecture (A. Dosovitskiy et al., 2021)

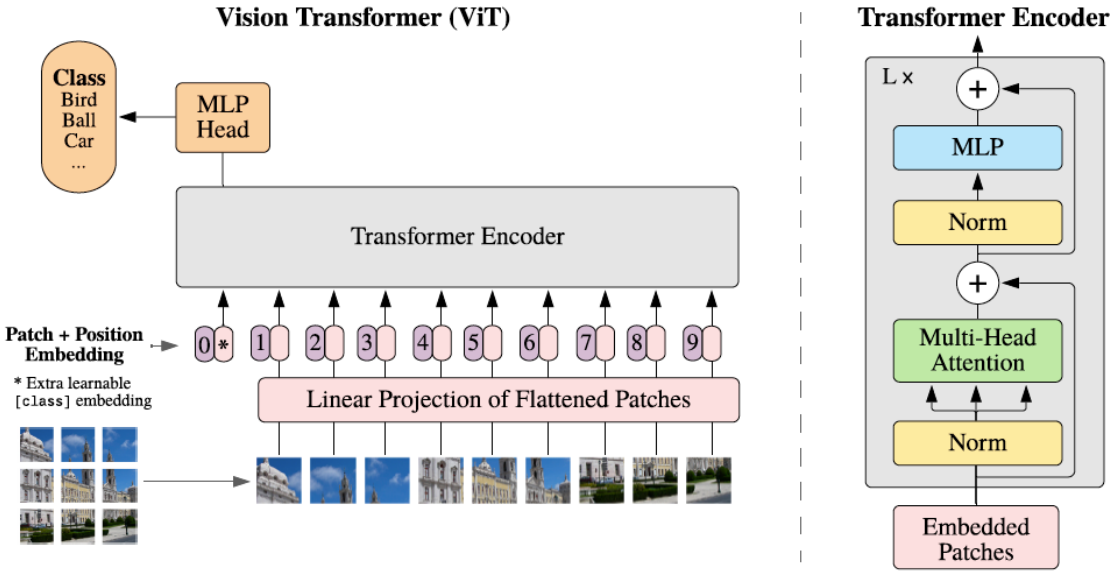

Fig 4. Self-attention block used in vision tasks.

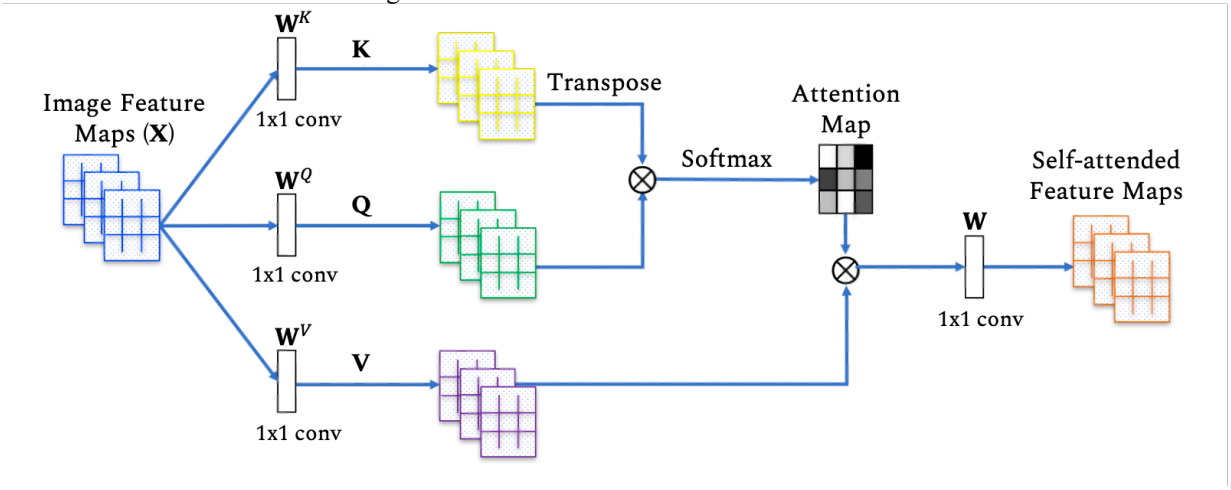

Fig 5. General framework of transformer-based object detection.

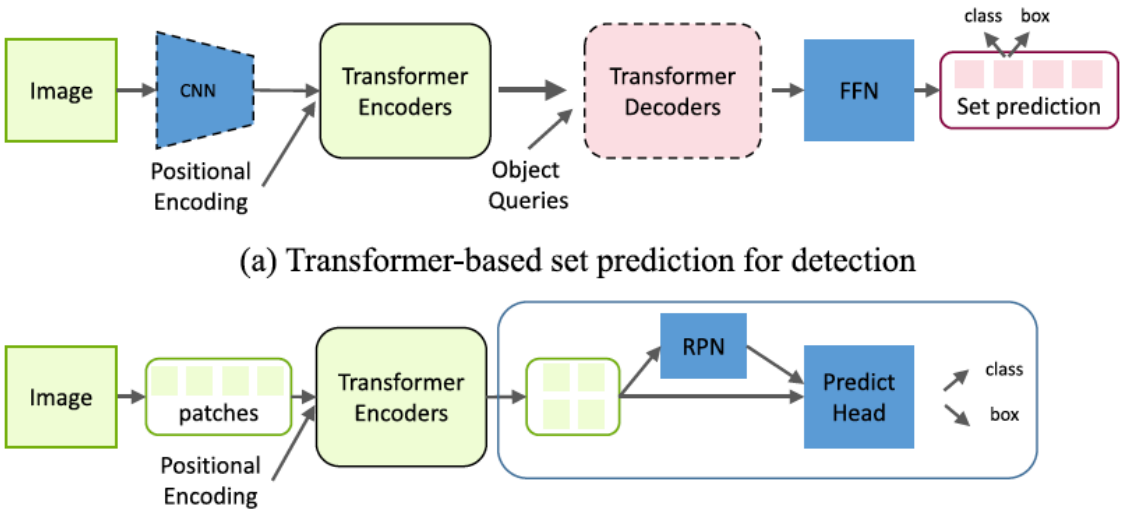

Fig 6. End to End Object detection with Transformers (DETR, Facebook AI).

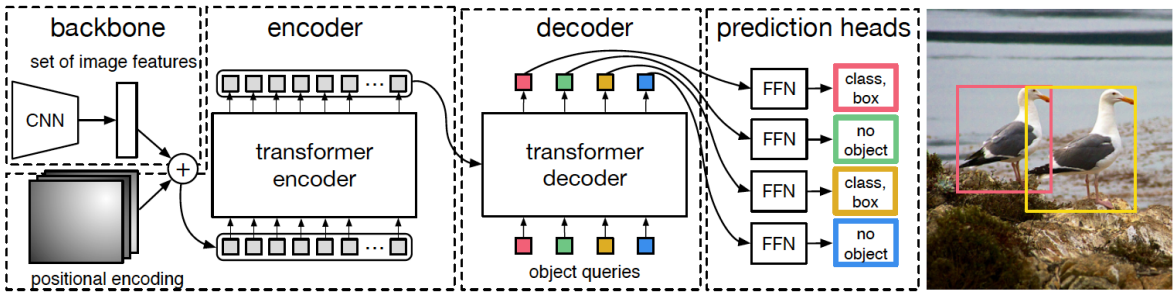

Fig 7. Transformer in Transformer (TNT)

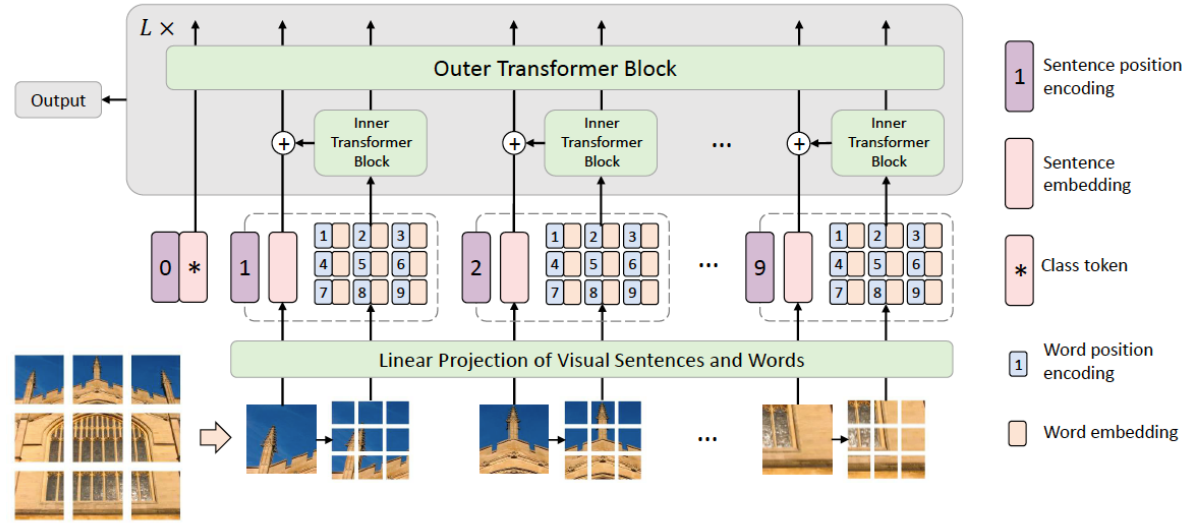

Fig 8. Swin Transformer architecture.

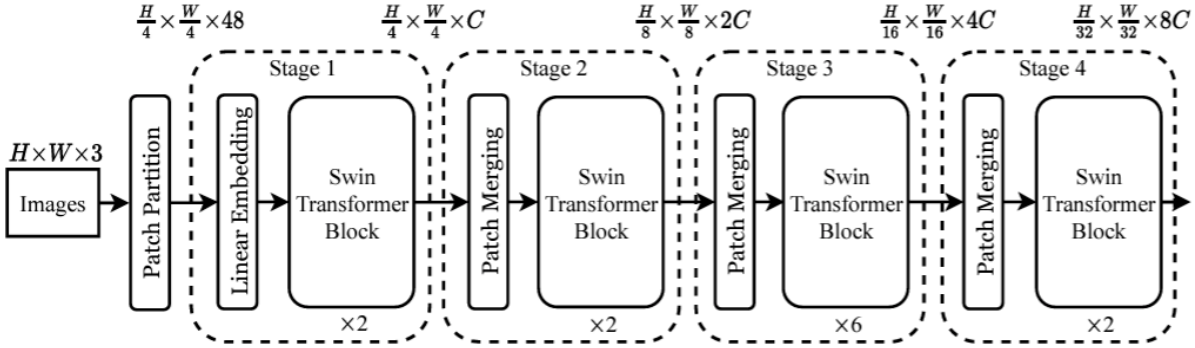

Fig 9. Swin Transformer: Hierarchical feature maps by merging image patches.

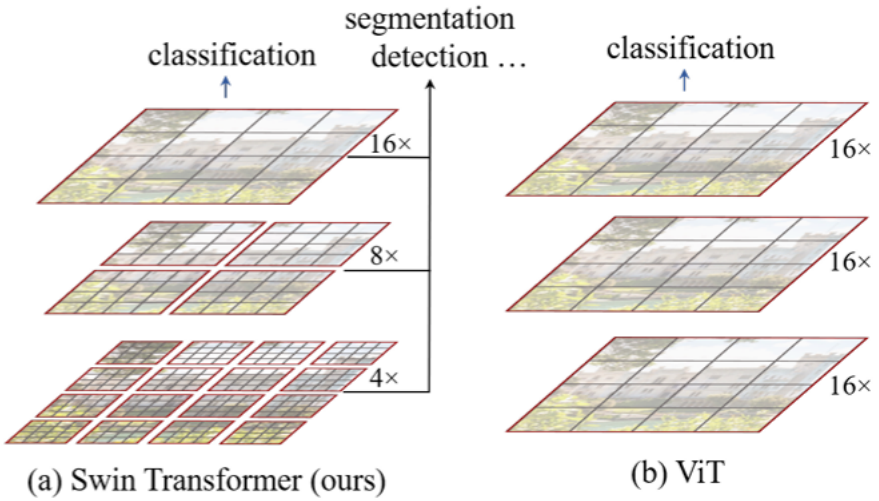

Fig 10. Swin-Unet Architecture

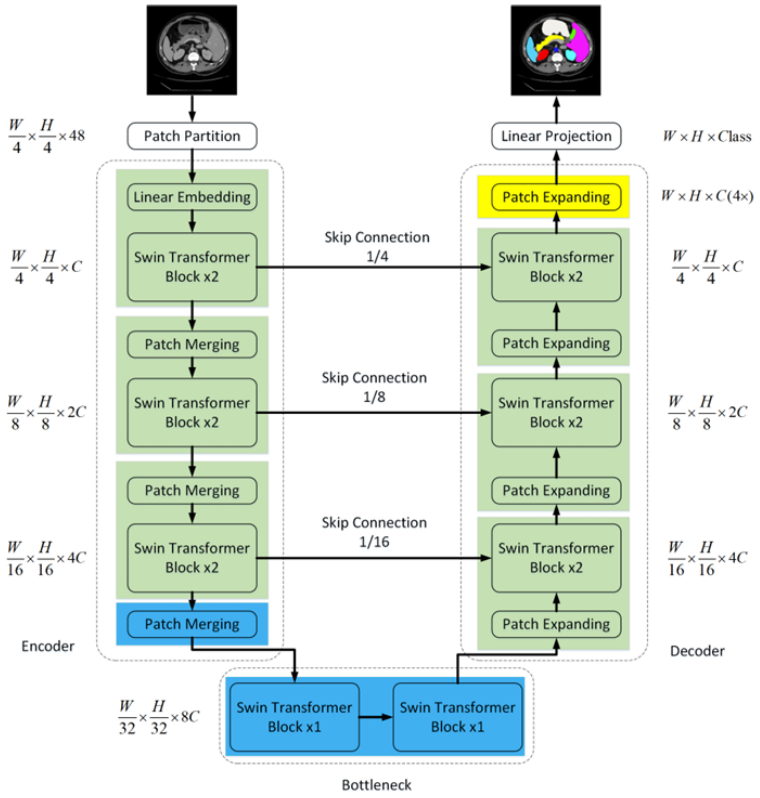

Fig 11. MedT Components

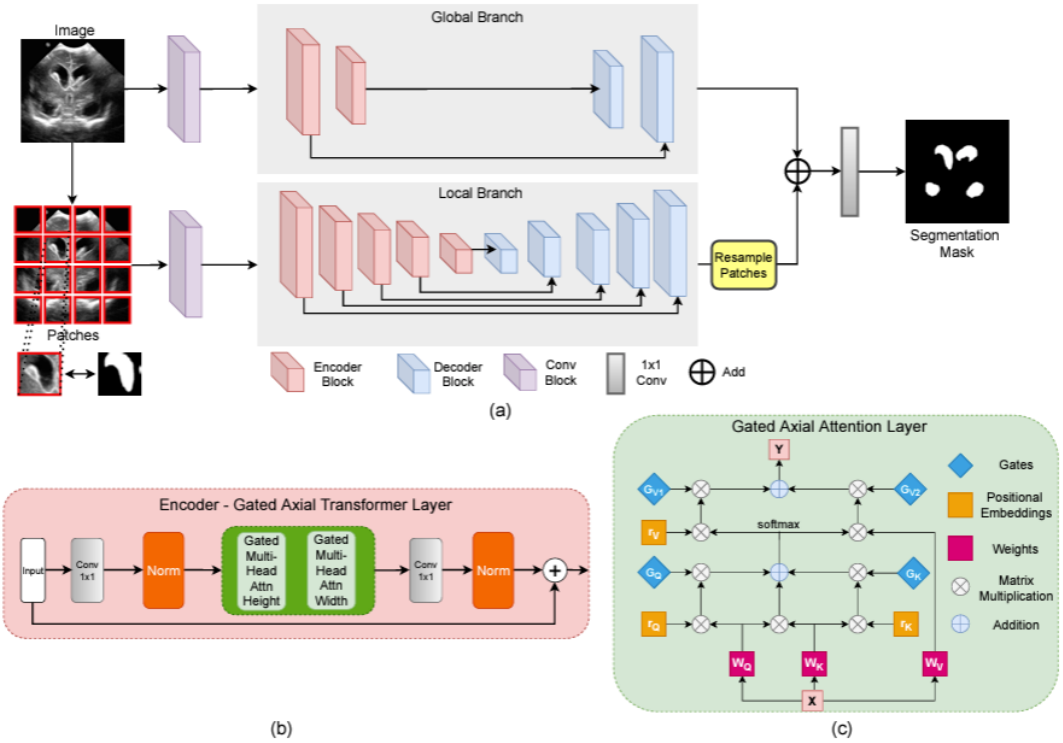

Fig 12. Performance outcomes ViT vs ResNet