Semantic image segmentation using Vision Transformer (ViT)

Pravin G. Gawande[1], Y. H. Dandawate[2],
Chandana Lole[3], Dnyaneshwari Limbhore[4], Rutuja Thore[5]

Department of Electronics and Telecommunication Engineering,
Vishwakarma Institute of Information Technology Pune,
Maharashtra, INDIA.

Abstract— In the era of computer vision, where image processing unveils intricate details for comprehensive visual understanding, semantic segmentation plays a pivotal role. Unlike image classification, which assigns a single label to an entire image, semantic segmentation delves into assigning distinct labels to each pixel, delineating precise boundaries of objects. This paper navigates through the evolution of segmentation techniques, from influential CNN-based models like AlexNet and VGG-16 to the transformative Vision Transformers (ViT) designed for language tasks. Despite ViT's success, limitations in resolution and computational costs on larger images sparked the emergence of models like SegFormer. This state-of-the-art Transformer framework revolutionizes both encoder and decoder components, introducing a novel positional-encoding-free and hierarchical Transformer encoder. The lightweight All-MLP decoder efficiently combines local and global attention, setting a new benchmark in efficiency, accuracy, and robustness across public datasets. The research explores the rich history of CNN-based segmentation, the transformative impact of Vision Transformers, and the groundbreaking SegFormer model. The proposed SegFormer methodology, featuring a hierarchical encoder and lightweight MLP decoder, is implemented and fine-tuned on the scene_parse_150 dataset. The results demonstrate significant improvements in segmentation accuracy, offering promising avenues for future research in transformer-based model segmentation.

Index Terms- —Semantic Segmentation, Transformers, ViT, Mean IoU, PyTorch.

I. Introduction

These days, computer vision is at its peak and plays a crucial role in many innovations. Image processing helping computers understand visual information by showing small details in pictures and make changes to them. Even though there is significant improvement in image classification [1], [2] [3], which assign a single label to an entire image, the challenge lies in differentiate multiple objects within the same image. This issue is overcome by segmentation technique. Semantic segmentation [4] [5], is one type of image segmentation, goes one step ahead of image classification by labeling individual pixel and giving accurate outline of each object which helps a detailed understanding of an image. In semantic segmentation machines trace objects outlines, helping to understand images in more detail through different categories. Distinguishing itself from instance segmentation [6] , semantic segmentation classifies pixels into specific categories without distinguishing between different objects within the same class. Convolutional Neural Networks (CNNs) initially work effectively in segmentation[7], tasks as in classification. Researchers experimented with number of combinations of convolutional and pooling layers to enhance segmentation model accuracy. Various developments in CNN-based models introduce us to encoder-decoder models [7], where the encoder extracts feature from the image, and the decoder further uses these features to make decisions. For further improvements in model transformer-based models introduced as explained in the paper 'Attention is All You Need' [8][3], Earlier models mainly focused on nearby information whereas transformers consider the entire picture which bring new way understanding things. Also, transformer model come with solution to the tendency of deep neural networks to "forget" certain features during training[9]. These prevent granular information loss during backpropagation and increase model robustness[8]. To solve vision-related tasks and increase their efficiency, a transformer-based model, known as the vision transformer [10], has been introduced. Despite the great performance of ViT, it faces limitations such as generating single-scale, low-resolution features and leads to high computational costs with larger images [11]. In this paper, we discuss the ViT-based model SegFormer [11], which is designed for semantic segmentation tasks. SegFormer represents a Transformer-based architecture that is complete reconstruction of both the encoder and decoder components. Due to the great performance of SegFormer on various semantic segmentation datasets, it is now widely used in the field of semantic segmentation.

II. Related work

A. Cnn-based Segmentation

The evolution of Convolutional Neural Network (CNN)-based methods for semantic segmentation has been instrumental in advancing computer vision capabilities. Beginning with AlexNet's [12], breakthrough in 2012, CNN architectures have continually pushed the boundaries of accuracy and efficiency in semantic segmentation tasks. AlexNet's success in the ImageNet competition demonstrated the potential of deep learning models, achieving a test accuracy of 84.6% with its innovative design featuring convolutional layers, max-pooling, ReLU activations, and dropout [12].

Following AlexNet, VGG-16 [13], emerged from Oxford's Visual Geometry Group, showcasing deeper architectures with 16 layers and smaller receptive fields. This model's victory in the ImageNet competition in 2013 with a test accuracy of 92.7% underscored the importance of depth in CNNs for semantic segmentation tasks. Similarly, GoogLeNet's[14], introduction of inception modules demonstrated novel ways to optimize computational resources while maintaining accuracy, achieving 93.3% accuracy in ImageNet.

However, one of the most significant advancements came with Microsoft's ResNet [9], which addressed the challenge of training very deep networks using residual blocks and identify skip connections. ResNet's 152-layer variant achieved a remarkable accuracy of 96.4% in ImageNet, setting a new standard for depth and performance in CNN architectures.

Semantic segmentation models typically adopt an encoder-decoder architecture, where the encoder extracts hierarchical features from input images, and the decoder refines these features for detailed predictions[15]. Various combinations of convolutional layers and pooling operations have been explored to enhance accuracy and efficiency in segmentation tasks[16].

Looking ahead, the Vision Transformer (ViT) [8], architecture introduces a paradigm shift by replacing convolutional layers with self-attention mechanisms, enabling better capturing of long-range dependencies in images[17]. This innovation holds promise for improving segmentation efficiency and accuracy, paving the way for future research directions in transformer-based models for semantic segmentation. Potential areas of exploration include experimenting with different transformer architectures, adapting self-attention mechanisms for segmentation tasks, and integrating transformer-based models with existing CNN architectures for enhanced performance and versatility in computer vision applications[17].

B. Vision Transformer

The rise of transformers, originally designed for language-based tasks, has significantly impacted the field of natural language processing (NLP), improving efficiency and performance in various applications. However, the success of transformers in NLP has sparked interest in exploring their potential for vision tasks as well [8].

In 2020, the Vision Transformer (ViT) [8], was introduced as a pioneering effort to extend the transformer architecture to handle visual tasks. Initially developed for image classification, ViT assigns a single label to the entire image, differing from segmentation tasks that require pixel-wise labeling. Despite this difference, the success of ViT in image classification paved the way for the exploration of transformer-based models in segmentation tasks [11].

Several models have emerged that utilize ViT architecture as a backbone for segmentation tasks. Notable examples include SETR, Swin Transformer [18], and ReSTR [19], which aim to adapt ViT for pixel-wise labeling challenges inherent in segmentation. These models modify the ViT architecture to improve segmentation accuracy, leveraging pre-training and fine-tuning on diverse datasets to observe performance changes[11].

SegFormer [11], represents another advancement in this domain, demonstrating the adaptability of transformer-based models for segmentation tasks. These models leverage the inherent strengths of transformers, such as capturing long-range dependencies and semantic information, to enhance segmentation efficiency and accuracy.

The Vision Transformer architecture offers unique advantages for segmentation tasks, such as its ability to handle global context and complex relationships within images[10]. This opens up new avenues for research in segmentation using transformer-based models. Future studies may explore various aspects, including architecture modifications, novel training strategies, and dataset augmentation techniques, to further improve the performance and versatility of transformer-based segmentation models[20]. Ultimately, transformer-based approaches hold immense potential to advance the state-of-the-art in computer vision, offering innovative solutions to complex segmentation challenges[21].

III. Methodology

In this section, we introduce the methodology followed in the paper to achieve semantic segmentation using a transformer. Initially, we needed to consider a vision transformer model modified for segmentation as ViT had a classification head [11]. Further, we fine-tune the pre-trained transformer-based model on custom dataset while considering required parameters to enhance segmentation alongside comparing the same with CNN based segmentation.

A. Model justification

With increased demand of Transformers [8], for NLP tasks Dosovitskiy et al. [10], first introduced a Vision Transformer (ViT). The ViT model was specifically created for image classification. To alter the ViT model for segmentation various other models have been proposed one of them is a SegFormer proposed by Enze Xie et al. [11]. It comprises of two layers i.e. a) modified Transformer encoder to produce low-resolution fine features and high-resolution coarse features important for segmentation; b) a lightweight Multiple Layer Perceptron (MLP) decoder to aggregate local and global attention from previous layers [11]. Due to its positional-encoding free Transformer encoder structure it avoids interpolating of different resolution images during inference. It implements a comparatively simple MLP decoder architecture which combines the local as well as global features from an image [11]. The SegFormer architecture redefines the field of image segmentation with its innovative Hierarchical Transformer Encoder and Lightweight All-MLP Decoder[11]. By effectively dividing the input image into patches and utilizing advanced self-attention techniques, SegFormer achieves high efficiency and accuracy in prediction.

Architecture:

The altered SegFormer architecture [11], for image segmentation mainly comprises two parts: a hierarchical encoder architecture and a Lightweight All-MLP Decoder. An image of size H × W × 3 is divided into patches of size 4 × 4 for an efficient dense prediction task. These patches are given as an input to the first layer of the model i.e. a transformer encoder to obtain multi-level features at {1/4, 1/8, 1/16, 1/32} resolutions of the original image.

Hierarchical Transformer Encoder [11]:

As evident from the name it consists of four hierarchical transformer encoder layers which produces corresponding hierarchical feature maps F_i with a resolution of $\frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}} \times C_{i+1}$ as demonstrated in Fig. 1 [11]. This layer enhances the segmentation process by employing optimized self-attention using sequence reduction approach and mix FFN technique for positional information of patches. The result of this process is a feature map formulated in eq. 1[11],

$$X_{out} = MLP(GRLU(CONC3X3(MLP(X_{in}))))+ X_{in}$$

then patch merging is used to obtain hierarchical feature map F_i corresponding to each hierarchical transformer encoder representing different abstraction levels.

Lightweight All-MLP Decoder [11].

The hierarchical feature map F_i generated from the transformer encoder is given as an input to the MLP decoder. Initially, the decoder processes the different channels of multi-level features through MLP layers. Following with the upsampling to 1/4th of the original resolution and fusion of concatenated features using another MLP layer. Hence generating a predicted mask (M) with dimensions H/4 × W/4 × N_{cls}.

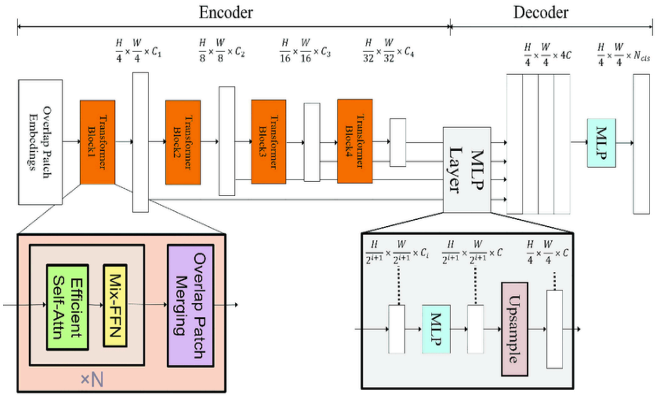
Encoder
Decoder
$\frac{H}{4} \times \frac{W}{4} \times C_1$
$\frac{H}{8} \times \frac{W}{8} \times C_2$
$\frac{H}{16} \times \frac{W}{16} \times C_3$
$\frac{H}{32} \times \frac{W}{32} \times C_4$
$\frac{H}{4} \times \frac{W}{4} \times 4C$
$\frac{H}{4} \times \frac{W}{4} \times N_{cls}$
Overlap Patch Embeddings
Transformer Block1
Transformer Block2
Transformer Block3
Transformer Block4
MLP Layer
MLP
Efficient Self-Attn
Mix-FFN
Overlap Patch Merging
×N
$\frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}} \times C_i$
$\frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}} \times C$
$\frac{H}{4} \times \frac{W}{4} \times C$
MLP
Upsample

Fig. 1. SegFormer Framework

This model has been pretrained on three segmentation datasets: ADE20K, Cityscapes, and COCO-Stuff. On Cityscapes dataset, SegFormer-B0 the lightweight model yielded 71.9% mean IoU(Mean Intersection over Union). The largest SegFormer model i.e. SegFormer-B5 yielded 84.0% mean IoU which had a significant improvement compared to previous Transformer based models such as SETR [11].

In this paper, we have considered the above transformer-based model for image segmentation due to its highly effective architecture and highly efficient obtained results.

B. Fine-Tuning of SegFormer

The section advances the fine-tuning process of the SegFormer model on a custom dataset to predict segmented masks. The SegFormer model has pretrained on larger datasets such as Imagenet-1K making it learn many valuable features in advance[11]. Thus, to initialise the fine-tuning process, we need to adapt the model in terms of modifying the final layers and decreasing the learning rate [22]. The fine-tuning procedure is implemented in the python code leveraging the modules of PyTorch explained in detail using key elements listed below.

Initially, the dataset has been preprocessed to prepare images and respective annotations according to the model's accepted input specifications. Notable data augmentation techniques [23], have been applied to enrich the training set with variations in the data. The model is optimised using the AdamW optimizer with a learning rate of 0.00006 on a custom dataset.

The model iterates over the training dataset for 200 epochs. The AdamW optimizer utilises backpropagation to update the model's parameters. The loss is computed by comparing the model's predictions with ground truth labels. The training also measures parameters such as pixel-wise accuracy and mIoU [24]. Based on the current model predictions and ground truth labels, the code calculates and prints the loss, mIoU, and mean accuracy after every 100 batches. Evaluation metrics, particularly Mean Intersection over Union (IoU) [24], are loaded for model assessment.

Additionally, bilinear interpolation[25], is utilised in the evaluation step to adjust the model's logits so that they match with the size of the ground truth labels. The model's efficacy during training can be evaluated by comparing the model's predictions to the ground truth and printing the calculated metrics. Overall, this section includes fine-tuning [22], a SegFormer model for semantic segmentation, covering data processing, model loading, training, evaluation, and visualization.

IV. Experiments

A. Dataset

The "scene_parse_150" dataset was meticulously crafted by selecting the top 150 objects based on their total pixel ratios from the ADE20K dataset. To maintain simplicity, large-sized images in the original ADE20K dataset were rescaled, ensuring a minimum height or width of 512 pixels. Among the chosen 150 objects, 35 belong to stuff classes (e.g., wall, sky, road), while 115 are discrete objects (e.g., car, person, table). The annotated pixels of these 150 objects collectively occupy 92.75% of all pixels in the dataset, with stuff classes and discrete objects accounting for 60.92% and 31.83%, respectively.

The dataset is partitioned into training, test, and validation sets. The training set comprises 20,210 images, the test set contains 3,352 images, and the validation set includes 2,000 images. Annotation masks within the dataset assign labels ranging from 0 to 150, where 0 corresponds to "other objects." Pixels labeled as 0 are not considered in the official evaluation.

Further details about the labels of the 150 semantic categories, including indices, pixel ratios, and names, can be found in a specific file provided with the dataset. This succinct overview provides essential information about the dataset's composition, object categories, pixel ratios, and dataset splits, setting the stage for its potential applications in semantic segmentation research.

B. Evaluation Metrics

The segmentation accuracy is measured by the Intersection over Union (IoU) and the mean IoU between the predicted segmentation and the ground truth. IoU is calculated by dividing the area of overlap between the predicted segmentation and the ground truth by the total number of pixels present across both masks [24].

$$IoU = \frac{(target \cap prediction)}{target \cup prediction}$$

The mean IoU is the average of the IoU for each class. Additionally, we calculated per-category IoU, indicates how well the model can distinguish objects of a particular category from their backgrounds in an image.

We are calculating the per category accuracy means evaluating a binary mask. A true positive (TP) represents a pixel that is correctly predicted to belong to the given class as in ground truth and a true negative (TN) represents a pixel that is correctly identified as not belonging to the given class. A false positive (FP) occurs when the segmentation model predicts the presence of a certain object or class in an image that is not actually present, and a false negative (FN) occurs when the model fails to predict the presence of a certain object or class that is present in the image. the accuracy can be represented as given below equation,

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

Similarly, along with per-category accuracy, we calculate the overall accuracy for each step. Mean IoU and overall accuracy provide us with information about the model's fit.

C. Results

Segmentation using Transformer based model:

After fine-tuning the SegFormer model[22], on a scene_parse_150 dataset there is need to evaluate the performance of the model to predict accurate segmented image. Hence, after training the model to learn features from the dataset we inferenced the model to showcase its ability to predict output from new given inputs. The parameters to decide the competency of the model for inference need to be well justified. The essential evaluation metrics we considered to check the success of transformer-based segmentation model on a custom dataset are Mean Intersection over Union(mIoU) [24], overall accuracy and per category wise accuracy and mIoU.

Mean Intersection over Union: This metric evaluates the model for inference based on the ground truth. It compares the ground truth and predicted segmentation mask on merits of similarity between the two by mathematically computing the area of overlap and union[24], and formulates the above to compute the performance of the model.

Overall accuracy: The accuracy of the model is an evaluation metric represented as a percentage. This metric provides information about the insights learnt by the model during training hence computing accuracy of the model to predict corresponding multi class segmented image. It considers the overall accuracy of predicted mask by combining per class accuracy.

Per-category metrics: The scene_parse_150 dataset contains total 150 categories of objects present in the images. The background class corresponds to 0 class index. We trained the model on class indexes ranging from 0 to 149 by reducing each class label by one and assigning background class to 255 to be ignored by loss function of our model. We need to compute correctness of the model to predict individual class segments we achieve this by considering per category metrics such as per-category accuracy and per-category mean IoU for individual image.

During training, the fitting of the model can be represented in the terms of training loss and validation loss. Fig. 2 represents the loss over number of epochs.

As evident from the Fig. 2 graph the model fits optimally on training and validation data.

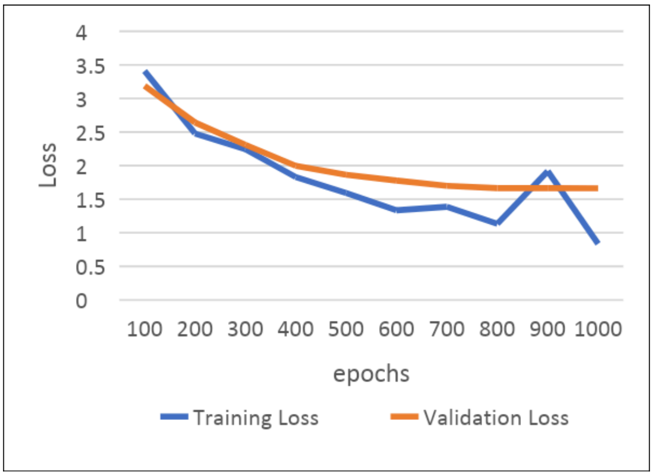

Fig. 2. Training and Validation Loss over no. of epochs

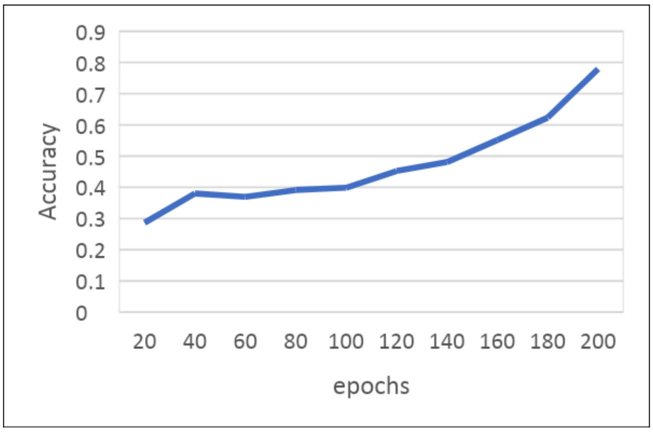

Fig. 3. accuracy over no. of epochs

Evident from fig. 3 the accuracy of the SegFormer model exhibits a noteworthy progression throughout the fine-tuning process [22], on the Scene-Parse-150 dataset. Starting at 0.28657 in the initial epochs, the model experiences fluctuations before demonstrating a steady improvement from epoch 80 to 160, indicating the assimilation of discriminative features. The most significant advancement occurs in the later epochs, with accuracy reaching 0.7793 at epoch 200, showcasing the model's convergence and high proficiency in semantic segmentation [15]. This substantial increase suggests successful adaptation to the custom dataset, reflecting the SegFormer model's ability to capture intricate patterns and details [11], ultimately achieving a commendable level of accuracy on the training data.

Segmentation using CNN based model:

As observed in the Fig.4, the loss per epoch demonstrates a consistent decrease for both training and validation sets. This behavior indicates that the model is effectively learning and fitting the training data. The diminishing training loss signifies that the model is optimizing its parameters to minimize errors on the training set. Simultaneously, the validation loss, which measures performance on unseen data, also shows a decreasing trend.

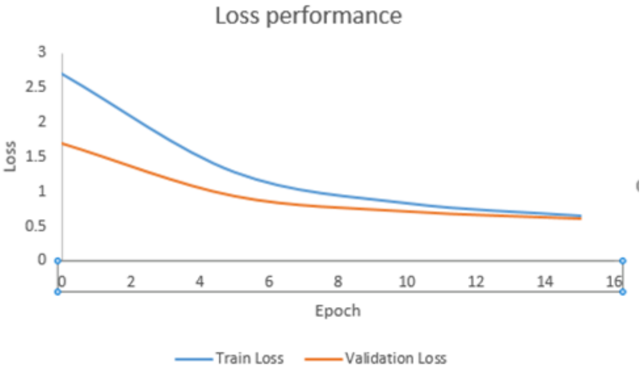

Fig. 4. Training and Validation Loss over no. of epochs

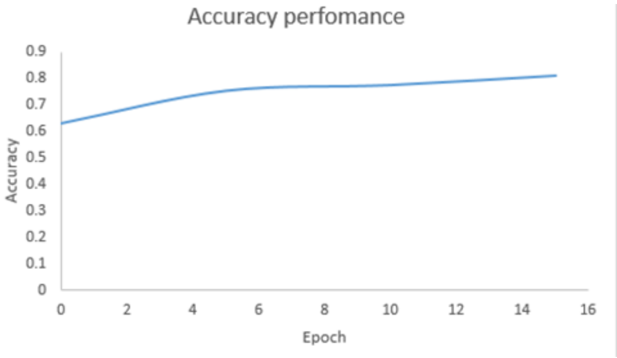

Fig. 5. Accuracy over no. of epochs

As depicted in the Fig. 5, there is a discernible upward trend, indicating a consistent improvement in the model's accuracy over successive epochs. This positive trajectory suggests that the model is learning and making more accurate predictions as training progresses. The rise in accuracy signifies an enhancement in the model's performance, showcasing its capacity to achieve higher precision in predicting the accurate segmentation of images [4].

Comparison drawn from the results:

After conducting the experiment, it became evident that SegFormer surpassed U-Net [26], in the realm of image segmentation. The accuracy achieved by SegFormer was notably higher, coupled with the advantage of requiring less computational resources compared to U-Net. The transformer-based architecture of SegFormer [11], played a pivotal role in enhancing parallelization, thereby contributing to its superior computational efficiency. Beyond efficiency gains, SegFormer demonstrated exceptional proficiency in generating highly accurate masks, especially in scenarios involving intricate details or expansive object dimensions. These results underscore SegFormer as a compelling choice for image segmentation, offering a compelling combination of heightened accuracy and computational efficiency.

References

[1] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” 2014, doi: 10.48550/ARXIV.1409.0575.

[2] P. G. Gawande and A. M. Sapkal, “Quality-dependent fusion system using no-reference image quality metrics for multimodal biometrics,” *Period. Eng. Nat. Sci. PEN*, vol. 6, no. 1, p. 260, Jun. 2018, doi: 10.21533/pen.v6i1.282.

[3] Dr. Shailesh V. Kulkarni, Dr. Pravin G. Gawande, Dr. Rajendra S. Talware, Dr. K. J. Raut, and Dr. Anup W. Ingle, “CAR IDENTIFICATION FOR BRAKE LIGHT DETECTION USING HAAR CASCADE APPROACH,” *Eur Chem Bull 2023*, vol. 12 (S3), no. S3, pp. 2961–2971, 2023, doi: doi: 10.31838/ecb/2023.12.s3.3642023.18/05/2023.

[4] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, Jun. 2015, pp. 3431–3440. doi: 10.1109/CVPR.2015.7298965.

[5] P. G. Gawande, “Enhancing Robustness and Generalization in Deep Learning Models for Image Processing,” *Power Syst. Technol.*, vol. Vol. 47 No. 4 (2023), no. 4, p. 12, Dec. 2023.

[6] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice: IEEE, Oct. 2017, pp. 2980–2988. doi: 10.1109/ICCV.2017.322.

[7] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3059968.

[8] A. Vaswani *et al.*, “Attention is All you Need,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[10] A. Dosovitskiy *et al.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.” arXiv, Jun. 03, 2021. Accessed: Dec. 17, 2023. [Online]. Available: http://arxiv.org/abs/2010.11929

[11] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, “SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers.” arXiv, Oct. 28, 2021. Accessed: Dec. 17, 2023. [Online]. Available: http://arxiv.org/abs/2105.15203

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2012. Accessed: Apr. 05, 2024. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[13] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition.” arXiv, Apr. 10, 2015. doi: 10.48550/arXiv.1409.1556.

[14] C. Szegedy *et al.*, “Going deeper with convolutions,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[15] S. Zheng *et al.*, “Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers.” arXiv, Jul. 25, 2021. doi: 10.48550/arXiv.2012.15840.

[16] F. Sultana, A. Sufian, and P. Dutta, “Evolution of Image Segmentation using Deep Convolutional Neural Network: A Survey,” *Knowl.-Based Syst.*, vol. 201–202, p. 106062, Aug. 2020, doi: 10.1016/j.knosys.2020.106062.

[17] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, “Attention Augmented Convolutional Networks.” arXiv, Sep. 09, 2020. doi: 10.48550/arXiv.1904.09925.

[18] Z. Liu *et al.*, “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows.” arXiv, Aug. 17, 2021. doi: 10.48550/arXiv.2103.14030.

[19] N. Kim, D. Kim, C. Lan, W. Zeng, and S. Kwak, “ReSTR: Convolution-free Referring Image Segmentation Using Transformers.” arXiv, Mar. 30, 2022. doi: 10.48550/arXiv.2203.16768.

[20] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens, “Stand-Alone Self-Attention in Vision Models,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2019. Accessed: Apr. 05, 2024. [Online]. Available:

https://papers.nips.cc/paper_files/paper/2019/hash/3416a75f4cea9109507cacd8e2f2aefc-Abstract.html

[21] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, “Segmenter: Transformer for Semantic Segmentation,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE, Oct. 2021, pp. 7242–7252. doi: 10.1109/ICCV48922.2021.00717.

[22] X. Han *et al.*, “Pre-trained models: Past, present and future,” *AI Open*, vol. 2, pp. 225–250, Jan. 2021, doi: 10.1016/j.aiopen.2021.08.002.

[23] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, vol. 6, no. 1, p. 60, Jul. 2019, doi: 10.1186/s40537-019-0197-0.

[24] M. A. Rahman and Y. Wang, “Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation,” in *Advances in Visual Computing*, G. Bebis, R. Boyle, B. Parvin, D. Koracin, F. Porikli, S. Skaff, A. Entezari, J. Min, D. Iwai, A. Sadagic, C. Scheidegger, and T. Isenberg, Eds., Cham: Springer International Publishing, 2016, pp. 234–244. doi: 10.1007/978-3-319-50835-1_22.

[25] S. Fadnavis, “Image Interpolation Techniques in Digital Image Processing: An Overview,” vol. 4, no. 10, 2014.

[26] “U-Net: Convolutional Networks for Biomedical Image Segmentation | SpringerLink.” Accessed: Apr. 05, 2024. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-24574-4_28

AUTHORS

First Author – Dr. Pravin G. Gawande, PhD, Vishwakarma Institute of Information Technology Pune,

Maharashtra, INDIA, *pravin.gawande@viit.ac.in*

Second Author – Dr. Prof. Y. H. Dandawate, PhD, Vishwakarma Institute of Information Technology Pune,

Maharashtra, INDIA, *yogesh.dandawate@viit.ac.in*

Third Author – Chandana Lole, Student, Vishwakarma Institute of Information Technology Pune, *chandana.22011165@viit.ac.in*

Forth Author – Dnyaneshwari Limbhore, Student, Vishwakarma Institute of Information Technology *dnyaneshwari.22010671@viit.ac.in*

Fifth Author – Rutuja Thore, Student, Vishwakarma Institute of Information Technology Pune, *rutuja.22010589@viit.ac.in*

Correspondence Author – Dr. Pravin G. Gawande, PhD,

Vishwakarma Institute of Information Technology Pune,

Maharashtra, INDIA, *pravin.gawande@viit.ac.in****,*** *9850287681.*