

# Semantic Segmentation using Vision Transformers: A survey

Hans Thisanke<sup>a</sup>, Chamli Deshan<sup>a</sup>, Kavindu Chamith<sup>a</sup>, Sachith Seneviratne<sup>b,c</sup>, Rajith Vidanaarachchi<sup>b,c</sup>, Damayanthi Herath<sup>a,\*</sup>

<sup>a</sup>*Department of Computer Engineering, University of Peradeniya, Peradeniya, 20400, Sri Lanka*

<sup>b</sup>*Melbourne School of Design, University of Melbourne, Parkville, VIC 3010, Australia*

<sup>c</sup>*Faculty of Engineering and IT, University of Melbourne, Parkville, VIC 3010, Australia*

---

## Abstract

Semantic segmentation has a broad range of applications in a variety of domains including land coverage analysis, autonomous driving, and medical image analysis. Convolutional neural networks (CNN) and Vision Transformers (ViTs) provide the architecture models for semantic segmentation. Even though ViTs have proven success in image classification, they cannot be directly applied to dense prediction tasks such as image segmentation and object detection since ViT is not a general purpose backbone due to its patch partitioning scheme. In this survey, we discuss some of the different ViT architectures that can be used for semantic segmentation and how their evolution managed the above-stated challenge. The rise of ViT and its performance with a high success rate motivated the community to slowly replace the traditional convolutional neural networks in various computer vision tasks. This survey aims to review and compare the performances of ViT architectures designed for semantic segmentation using benchmarking datasets. This will be worthwhile for the community to yield knowledge regarding the implementations carried out in semantic segmentation and to discover more efficient methodologies using ViTs.

*Keywords:* vision transformer, semantic segmentation, review, survey, convolution neural networks, self-supervised learning, deep learning

---

\*Corresponding author

Email addresses: e16368@eng.pdn.ac.lk (Hans Thisanke), e16076@eng.pdn.ac.lk (Chamli Deshan), e16057@eng.pdn.ac.lk (Kavindu Chamith), sachith.seneviratne@unimelb.edu.au (Sachith Seneviratne), rajith.vidanaarachchi@unimelb.edu.au (Rajith Vidanaarachchi), damayanthiherath@eng.pdn.ac.lk (Damayanthi Herath)

---

## 1. Introduction

Transformers became the new state-of-the-art in natural language processing (NLP) [1] after the tremendous success it achieved. This led to the development of ViT [2] which was later adapted into the computer vision tasks such as image classification [2, 3], semantic segmentation [4, 5] and object detection [6, 7]. A typical Transformer encoder consists of a multi-head self-attention (MSA) layer, a multi-layer perceptron (MLP), and a layer norm (LN). The main driving force behind the ViT is the multi-head self-attention mechanism. It helps ViT to capture long-range dependencies with less inductive bias [8]. When trained on a sufficient amount of data, ViT shows remarkable performance, beating the performance of state-of-the-art CNNs [2]. However, ViTs still have some drawbacks compared to CNNs such as the need for very large datasets. Strategies such as self-supervised based approaches can be used to alleviate some of these drawbacks and further enhance ViTs [9].

Semantic segmentation is the process of assigning a class label to each and every pixel of an image. This requires accurate predictions at the pixel level. For segmentation, there exist both CNN-based models and Transformer based models. However, plain ViT models cannot be directly used for segmentation tasks because they do not consist of segmentation heads [10]. Instead SETR [5] and Swin Transformer [4] based architectures can be utilized for segmentation tasks. Unlike image classification, dense prediction tasks such as semantic segmentation and object detection come with a few difficulties due to the rich intra-class variation, context variation, occlusion ambiguities, and low image resolution [11]. There have been many improvements in the ViT domain in the last few years to overcome these challenges while further developments are still in progress to make them efficient.

The review focuses specifically on semantic segmentation using Vision Transformers. The comparison of the ViT models specialized for semantic segmentation is discussed with architecture-wise and tabulated specific sets of model variants that can be compared with the same set of benchmark datasets. The current surveys performed on ViTs have been structured with a detailed historical evolution from NLP to the Vision Transformer domain. [12] focuses on self-attention and its varieties with advantages and limitations with existing methods for segmentation, object detection, classification, and action recognition. The comparison follows between CNN and ViT backbones on the ImageNet dataset. The survey done by [13] is also considering various vision tasks and surpasses CNN-based models with experimental re-

sults on benchmark datasets. Even though several surveys have been done [12, 13, 14], a comparison between segmentation models with several benchmark datasets to identify the best-performing model has not been performed. In our survey, we provide a set of segmentation models, for each of which we define the best variant in each benchmark dataset category. This is useful in the sense of identifying the most optimal parameters such as patch size, iterations count for each variant of the model. By providing mIoU (%) of model performance results over several semantic segmentation-related benchmark datasets, overall evaluation and highest-performing model variants for each dataset can be identified.

In Section 2 we discuss the applications of semantic segmentation, ViTs, their challenges, and loss functions. Section 3 describes benchmark datasets used in semantic segmentation. Section 4 describes the existing work done in semantic segmentation using ViTs and presents a quantitative analysis. Finally, Section 5 provides the discussions and Section 6 concludes the paper with future directions.

## 2. Semantic Segmentation using Vision Transformers

This section aims to provide an in-depth analysis of the applications in semantic segmentation, with a focus on recent advancements in ViTs. We begin by exploring the principles and architecture of ViTs and their potential for improving semantic segmentation performance. We then delve into various application domains of semantic segmentation. We also devote a section to practical approaches for overcoming the data limitations that often arise in ViT models. Finally, we discuss various loss functions used in semantic segmentation and their effectiveness in different scenarios.

### 2.1. *Vision Transformers*

Automatic segmentation techniques have been evolving and improving throughout the years with the advancements of deep learning approaches and the application of semantic segmentation in practical usage. For semantic segmentation, the requirement is to locally identify the different classes in the image with spatial location. For that, the fully connected layers in the conventional CNN architecture were replaced with fully convolutional layers combined with feature extraction. This was introduced as Fully Convolutional Networks (FCN) [15] to identify high-level semantic features from images. These networks have shown to be faster compared to previous CNN-based techniques and are also capable of generating segmentation maps for images of any resolution. Some of the commonly known architectures are

U-Net (state-of-the-art FCN) and more improved architectures with higher accuracy and efficiency are developed by [16, 17, 18].

One of the limitations identified with the FCN architecture is the low resolution of the final output segmentation image of the feature map due to going through several convolutional and pooling layers. Furthermore, the locality property of the FCN-based methods caused limitations to the capture of long-range dependencies of the feature maps. To solve this, researchers also looked into attention mechanisms to merge or replace these models. This has led to trying out Transformer architectures in the computer vision domain which were successful in NLP.

Self-attention-based architectures have taken priority in NLP by avoiding the drawbacks such as vanishing gradients in sequence modeling and transduction tasks. Specially designed for sequence modeling and transduction tasks, Transformers with attention were able to model long-range sequences of data. When training a NLP model, one of the best ways is to pre-train on a large text corpus and then fine-tune on a small set of data which is for the related task. But with deep neural networks, this was a challenging task. As Transformers have high computational efficiency and scalability, it was easier to train on a large set of data [19].

With the success of using self-attention to enhance the input-output interaction in NLP, works have been proposed to combine convolutional architectures with self-attention, especially in object detection and semantic segmentation where input-output interaction is highly needed [20]. But applying attention to convolutional architectures demands high computation power, even though they are theoretically efficient [1].

Considering images, calculating self-attention is quadratic to the image size as each pixel attends to every other pixel therefore it is a quadratic cost of the pixel count [2]. Thus [2] proposed to divide the image into a sequence of patches and treat them as tokens as it was done in NLP. Instead of pixel-wise attention, patch-wise attention was used in the architecture which helped to reduce the computational complexity compared to applying self-attention to convolutional architecture.

This architecture showed promising results by surpassing all the state-of-the-art convolution-based methods by reaching an accuracy of 88.55% on ImageNet, 90.72% on ImageNet-Real, and 94.55% on CIFAR-100 datasets [2]. A major characteristic of the ViT is that it needs more data for model training. Experiments carried out by [2] ensure that with increasing data size, ViT performs well.

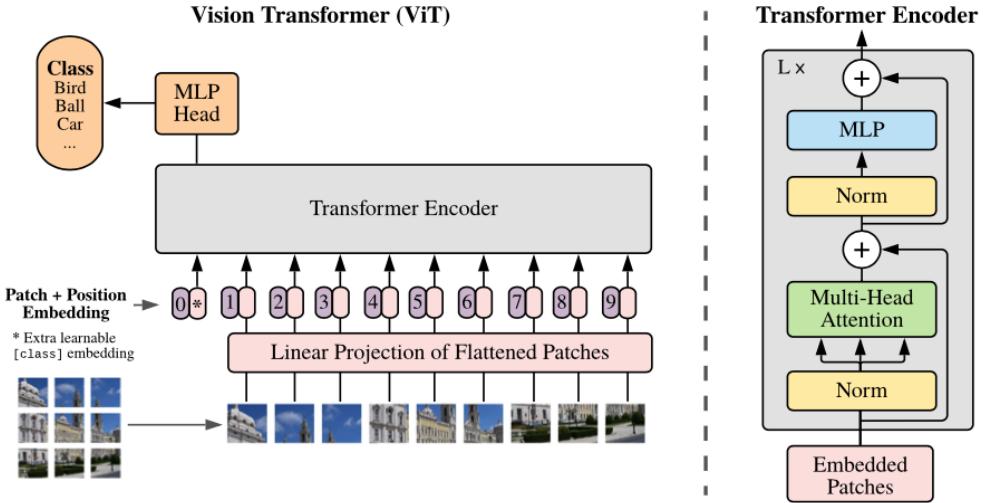


Figure 1: Architecture of the Vision Transformer. The model splits an image into a number of fixed-size patches and linearly embeds them with position embeddings (left). Then the result is fed into a standard transformer encoder (right). Adapted from [2].

## 2.2. Applications of Semantic Segmentation

In this section, we discuss various application domains of semantic segmentation, including remote sensing, medical imaging, and video processing. For each of these domains, we highlight the unique challenges and opportunities that arise, as well as the current state-of-the-art methods and techniques.

### 2.2.1. Semantic Segmentation of Remote Sensing Images

Remote sensing is the process of getting information and monitoring the characteristics of an area without having any physical contact. The two main types of remote sensing techniques are the use of active sensors such as RADAR, LiDAR and the use of passive sensors such as satellite imagery [21]. These high-resolution earth surface images provide a wide range of use cases such as world mapping updates [22], forest degradation analysis [23], monitoring changes to the surface [24], etc.

Remote sensing imagery is widely used in combination with computer vision and Artificial Intelligence (AI) for analyzing and processing the earth's surface over large areas with complex feature distributions. The images collected by satellites or unmanned aerial vehicles (UAV) provide a wide range of information for applications such as urban planning, disaster management, traffic management, climate change, wildlife conservation, crop monitoring, etc. The use of datasets containing these high-resolution images and their respective segmented masks [25] have provided a base for remote sensing

image analysis using computer vision and AI. The use of neural networks provides the ability to process large amounts of image data for object detection, semantic segmentation, and change detection tasks. The evolution in the remote sensing domain has further improved satellite sensors and the introduction of drone technology for aerial imagery has been vital to getting finer details on the earth's surface. This has resulted in precise and accurate data for processing using AI techniques [26].

Remote sensing images of the earth's surface provide land cover areas that can be categorized into different segmented classes. Each of these classes is assigned a label for each pixel while preserving the spatial resolution of the image. Many datasets containing these remote sensing images and their segmented masks are available [25, 27, 28] to use for different applications such as change detection, land cover segmentation, and classification. Examples of common land cover classes covered by the pixel-level classification are forests, crops, buildings, water resources, grasslands, roads, etc. Research has been conducted using ViT architecture models by adding layers and attention mechanisms efficiently and improvements in performance to process high-resolution remote sensing images for semantic segmentation such as Efficient Transformer [10] and Wide-Context Transformer [29].

Manual segmentation of these different environmental areas from a complex satellite or aerial images is a difficult task which is time-consuming, error-prone, and requires expertise in the remote sensing domain.

### *2.2.2. Semantic Segmentation of Medical Images*

Medical image analysis has developed and incorporated scanning and visualization techniques. Segmentation techniques have been vital as it has the ability to identify and segment medical imagery to assist in further diagnosis and interventions. By identifying each region of interest (ROI) highlighted, various important diagnoses are happening such as brain tumor boundary detection from MRI images, pneumonia affections in X-rays, cancer detection from biopsy sample images, etc. The demand for this type of analysis through image segmentation has emerged in the recent past with much research being done in the scope to develop more precise, efficient models and algorithms. These medical images that are used in image segmentation tasks can be grouped based on modalities such as MRI, CT scan, X-ray, ultrasound, microscopy, dermoscopy, etc. Each of these categories contains datasets that were collected under medical supervision and some are made publicly available.

Since there exist several modalities as mentioned above, the technological systems that are used for medical imagery differ. Medical imagery system development vendors built them as per the doctor's requirements. There-

fore, the images generated are bound to the limitations of the technology available and require medical personal intervention to examine them [30]. Therefore the segmentation of these images in different biological domains requires experts in each field to cope with these systems and spend a vast amount of time examining them. To overcome these difficulties, the capability of automatic feature extraction has been introduced with deep learning based techniques, which have been valuable in the sense of medical imagery. With the advancements in segmentation analysis, better-performing models have been introduced with the use of medical images by many researchers. One such famous architecture is the U-Net [31] which was initially introduced for medical image analysis. Based on this, several improved versions have been followed up using medical imagery datasets from heart, lesion, and liver segmentation [32, 33, 18]. This proves how beneficial the improvement of segmentation has been in the medical environment. In recent years, the emerging new architectures of ViTs have also been applied to the medical domain with TransUNet [34] and Swin-Unet [35]. They are hybrid Transformer architectures with the advantages of the U-Net. They performed with better accuracy in cardiac and multi-organ segmentation applications.

Some limitations of medical images are the relatively less number of images available compared to natural image datasets (landscapes, people, animals, and automobiles) with millions of images. In the medical domain, there are several image modalities. For annotating medical images, expertise in each medical field is a must. Among them, MRI and microscopy images are quite difficult to annotate [36]. Typically, these datasets contain fewer images compared to ultrasound, X-ray, and lesion datasets which are obtained with the existing scanning systems and are easier to annotate with less complex structures and fine boundaries. But still, limitations exist due to restrictions on privacy and other medical policies to obtain these images in large quantities. To overcome these limitations with some datasets, several image segmentation challenge competitions are taking place every year which provide publicly available well-annotated medical image datasets. Most of the improvements made through research in semantic segmentation models have been based on these challenge datasets and most are taken as benchmark datasets for segmentation [37, 38, 39].

### 2.2.3. Video Semantic Segmentation

Human-Machine interaction [40], augmented reality [41], autonomous vehicles [42], image search engines [43] are some applications in complete scene understanding and for these type of applications, semantic segmentation contributes more on complete scene understanding on videos. Usually, the idea is to apply semantic segmentation on frames of a high-resolution video where

the video is considered as a set of uncorrelated fixed images [44]. The common challenge with this type of semantic segmentation is the computational complexity of scaling the spatial dimension of the video using the temporal frame rate. Removal of temporal features and only focusing on spatial frame-by-frame features doesn't make sense in video segmentation. Since there is a combined flow among frames of a video, considering the temporal context of a video is an essential factor in video semantic segmentation, even though it is computationally expensive.

Research has been conducted to reduce this high computation cost on videos. Feature reuse and feature warping [45] have been proposed as a solution. Cityscapes [46] and CamVid [47], are some largest video segmentation datasets available for frame-by-frame approach of video segmentation [48]. Recent papers have proposed segmentation methods such as selective re-execution of feature extraction layers [49], optical flow-based feature warping [50], and LSTM-based, fixed-budget keyframe selection policies [51]. The main key problem in these approaches is that they have less attention to the temporal context of a video. Researchers have shown that to satisfy both spatial and temporal contexts, using an optical flow of video as temporal information to speed up uncertainty estimation makes good sense [52]. VisTR [53], TeViT [54] and SeqFormer [55] are some of the Transformer models that are used for video segmentation tasks.

### *2.3. Practical approaches to overcome the data limitation*

Deep neural networks have performed well with supervised learning in computer vision and NLP. But when it comes to the real world, supervised learning faces a bottleneck in training a neural network as it needs lots of labeled data. Collecting labeled data or manual labeling is difficult in every aspect. Training a network from scratch is a somewhat costly task; as a remedy for this, transfer learning comes into play. But when considering specified downstream tasks such as satellite imagery semantic segmentation, using pre-trained datasets is difficult as most of the architectures have been trained on benchmark datasets where the data domain is different. Therefore, getting good accuracy has been tricky.

Specially when considering Transformer architectures, self-supervised learning plays a great role as a remedy for data-hungry problems in deep learning. In human vision, humans are fed with different things in the environment and then are able to distinguish those things from other objects in the environment. There are no labeling mechanisms for these scenarios. Therefore, this is the technique used in SSL which actually trains a neural network using an unlabeled dataset where the labels are automatically provided through the dataset itself. As the first step, the network is set to solve a pretext

task as described in Figure 2. A pretext task is a pre-designed task from which the network can learn features and then using those trained weights for different features, the network can be applied to solve some downstream tasks. A downstream task is a specified task. Common downstream tasks in computer vision are semantic segmentation, object detection, etc.

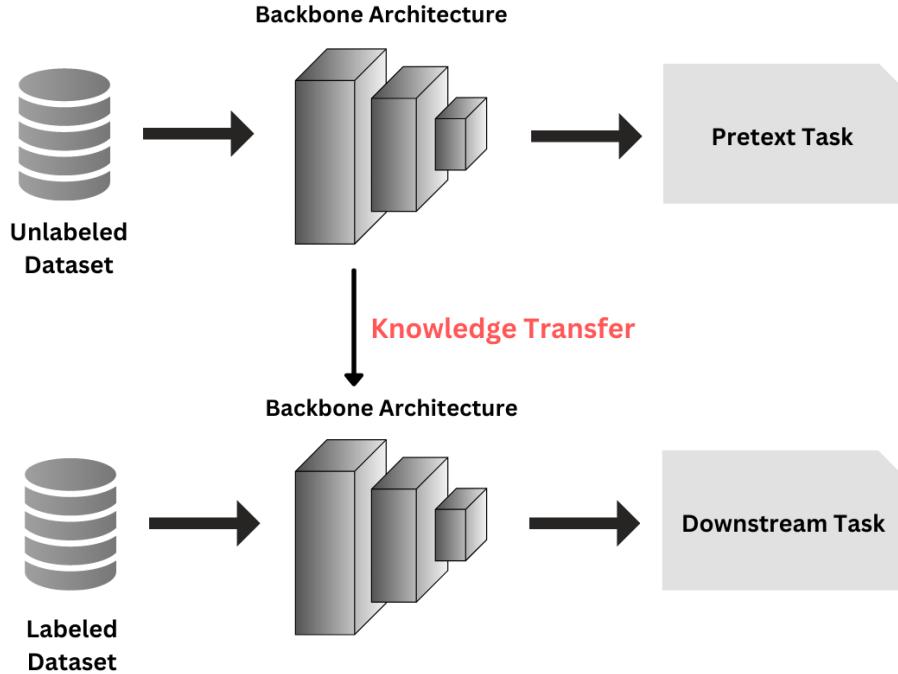


Figure 2: The general pipeline of self-supervised learning. The trained weights from solving a pretext task are applied to solve some downstream tasks.

Rotating an image by a given angle and predicting the rotation, solving jigsaw puzzles, filling a cut patch on an image, predicting the relative position of a patch of an image, and separating images belonging to different clusters can be considered as some of the pretext tasks in SSL [56]. By using these methods, the network can learn different features in the dataset under the given scope. No labels are used here and automatic labeling is achieved via the image itself.

SSL has three general categories based on how the training happens.

- Generative: Train the encoder to encode the given input and using the decoder get the input back
- Contrastive: Train the encoder to encode the given input and find the similarities

- Generative-Contrastive (Adversarial): Train encoder to encode the given input and create fake outputs and compare the features of the input and output [57]

Semantic segmentation is one of the major downstream tasks that can be performed using SSL. Pixel-wise labeling is essential in semantic segmentation. If there are no properly annotated datasets, SSL is the best way to train semantic segmentation architectures.

#### *2.4. Loss functions in semantic segmentation*

For segmentation, classification, and object detection models accuracy improvement not only depends on the model architectures but also on the loss functions used. The loss function calculates the overall error while training batches and adjust the weights through back propagation. Numerous loss functions have been created to cope with various domains, and some of them are derived from existing loss functions. Additionally, these loss functions take into account the imbalances in the dataset too.

In the case of semantic segmentation, the default choice and most commonly used is the cross-entropy loss which is applied pixel-wise. The loss function independently evaluates the class predictions for each pixel and averages over all the pixels.

$$CE_{loss}(p, q) = - \sum_{i=1}^n p_i \log(q_i) \quad (1)$$

The equation 1 above computes the average loss for each pixel in an image. Here in the equation  $p_i$  is the true probability of the  $i^{th}$  class and  $q_i$  is the predicted probability of the same class. This supports the model to generate probability maps that closely resemble the actual segmentation masks while penalizing inaccurate predictions more heavily. By minimizing the cross-entropy loss function during training, the model becomes better at precise image segmentation.

Even though the above method is widely used it can be biased with dataset imbalance as the majority class will be dominant. To overcome this when the dataset is skewed, a weighted cross entropy loss is introduced in [31].

$$WCE_{loss}(p, q) = - \sum_{i=1}^n p_i w_i \log(q_i) \quad (2)$$

Here as in equation 2, a weight factor as  $w_i$  for the  $i^{th}$  class is inserted to the typical equation 1. But the issue was not significantly solved as the cross

entropy calculates the average per-pixel loss without considering the adjacent pixels which can be boundaries.

As a further improvement for the cross-entropy loss, the focal loss technique [58] was introduced. This is implemented by altering the structure of cross-entropy loss. When focal loss is applied to samples with accurate classifications, the scaling factor value is down-weighted. This ensures the more harder samples are emphasized, therefore high class imbalance won't bias toward the overall calculations.

$$F_{loss}(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t) \quad (3)$$

In the equation 3,  $p_t$  is the predicted probability of the true class,  $\alpha_t$  is a scaling factor that gives higher weight to the positive class, and  $\gamma$  is a focusing parameter that controls how much the loss is focused on hard examples.

The cross-entropy loss is scaled in this loss function, with the scaling factors decreasing to zero as the confidence in the well-classified classes rises. Therefore more attention is given to the pixel classes which are difficult to predict.

Another set of loss calculation techniques is the overlapping between prediction and actual segmentations. The models are trained to minimize the loss such that the model outputs segmentations with higher overlaps.

Dice loss is one such widely used popular measure in computer vision tasks to calculate the similarity between two images. It is based on the dice coefficient which was later developed as the dice loss function in the segmentation domain. This loss was first used in the computer vision domain by [59] in medical image segmentation tasks.

$$D_{loss}(g, p) = 1 - \frac{2 \sum_{i=1}^n g_i p_i}{\sum_{i=1}^n g_i + \sum_{i=1}^n p_i + \epsilon} \quad (4)$$

Here, in equation 4  $g$  and  $p$  describes the ground truth and prediction segmentations. The sum is calculated over the  $n$  number of pixels with  $\epsilon$  small constant added to avoid division by zero. The dice coefficient measures the overlap between the samples (ground truth and prediction) and provides a score ranging from 0 to 1, 1 means perfect overlap. Since this method considered pixels in both global and local contexts, the accuracy is higher than cross-entropy loss calculations.

Another similar method used to evaluate the metric of models is the IoU (Intersection over Union) loss also known as the Jaccard index. It is quite similar to the dice metric and measures the overlapping of the positive instances between the considered samples. This method as shown in equation

$\text{IoU}_{loss}(g, p) = 1 - \frac{\sum_{i=1}^n g_i p_i}{\sum_{i=1}^n g_i + \sum_{i=1}^n p_i - \sum_{i=1}^n g_i p_i + \epsilon}$

For multi-class segmentation, the mean IoU is considered by taking the average of each individual class IoU. This is widely used for performance comparison and evaluation of dense prediction models [60].

### 3. Datasets

In this section, the common datasets used for the training and testing of semantic segmentation models are considered. Factors affecting the creation of real datasets are lighting conditions, weather, and season. Based on these factors, datasets can be classified into different groups. When data is collected under normal daytime environmental conditions, those data are categorized under no cross-domain datasets. If data is collected under some deviated environmental conditions including rainy, cloudy, nighttime, snowy, etc then such data are categorized under cross-domain datasets. Another category is synthetic data, where the data is artificially created and collected for training purposes. These synthetic datasets are mostly created as a cost-effective supplement for training purposes. Following are some of the benchmark datasets specially made for semantic segmentation tasks, with a summary presented in Table 1.

**PASCAL-Context** [61] This dataset was created by manually labeling every pixel of PASCAL-VOC 2010 [62] dataset with semantic categories. The domain of this dataset is not limited and its data contains different objects. The semantic categories of this dataset can be divided into three main classes. (i) objects, (ii) stuff, and (iii) hybrids. Objects have defined categories such as cups, keyboards, etc. Stuff has classes without a specific shape and has regions such as sky, water, etc. Hybrid contains intermediate objects such as roads where roads have a clear boundary but shape cannot be predicted correctly.

**ADE20K** [63] Annotations of this dataset are done on scenes, objects, parts of objects. Many of the objects in the dataset are annotated with their parts. Annotations in this dataset are made continuously. Therefore, this is a growing dataset.

**KITTI** [64] This dataset contains both 2D and 3D images which have been collected from urban and rural expressway incidents and traffic scenarios. It is useful for robotics and autonomous driving. This dataset has

different variants namely KITTI-2012, KITTI-2015 and they have some differences in the ground truth.

**Cityscapes** [46] This contains large-scale pixel-level and instance-level semantic segmentation annotations recorded from a set of stereo video sequences. Compared to other datasets, quality, data size, and annotations in this dataset have a good rank and data have been collected from 50 different cities in Germany and neighboring countries.

**IDD** [65] This is specially designed for road scene understanding and data have been collected from 182 Indian road scenes. As these are taken from Indian roads, there are some variations in the weather and lighting conditions because of dust and air quality on roads. One key feature of this dataset is, this contains some special classes such as auto-rickshaws and animals on the roads.

**Virtual KITTI** [66] Except for different weather and imaging conditions, most of the virtual vision datasets such as Virtual KITTI are similar to the real vision datasets. Therefore virtual datasets are useful for pre-training purposes. This dataset is created from 5 different urban scene videos from the real-world KITTI dataset. Data have been automatically labeled and can be used for object detection, semantic segmentation, instance segmentation, etc.

**IDDA** [67] This contains 1 million frames generated from simulator CARLA oriented on different 7 city models. This dataset can be used to do semantic segmentation for more than 100 different visual domains and is specially designed for autonomous driving models.

Dataset	Classes	Size	Train	Validation	Test	Resolution (pixels)	Category
PASCAL-Context	540	19740	4998	5105	9637	387 × 470	No cross-domain
ADE20K	150	25210	20210	2000	3000	-	No cross-domain
KITTI	5	252	140	-	112	1392 × 512	No cross-domain
Cityscapes	30	5K fine, 20K coarse	2975	500	1525	1024 × 2048	Cross-domain
IDD	34	10004	7003	1000	2001	1678 × 968	Cross-domain
Virtual KITTI	14	21260	-	-	-	1242 × 375	Synthetic
IDDA	24	1M	-	-	-	1920 × 1080	Synthetic

Table 1: **Summary of the datasets**

Note: Both cross-domain and no-cross domain falls into the non-synthetic category

#### 4. Meta - analysis

In this section, we discuss some of the ViT models specialized for the task of semantic segmentation. The models are selected upon considering the datasets that they benchmarked (ADE20K, Cityscapes, PASCAL-Context).

The intuition behind that is to compare all the models on a common basis. The benchmark results are summarized in Table 2.

#### 4.1. *SEgmentation TRansformer (SETR)*

SETR [5] proposes semantic segmentation as a sequence-to-sequence prediction task. They adopt a pure Transformer as the encoder part of their segmentation model without utilizing any convolution layers. In this model, they replace the prevalent stacked convolution layer based encoder with a pure Transformer which gradually reduces the spatial resolution.

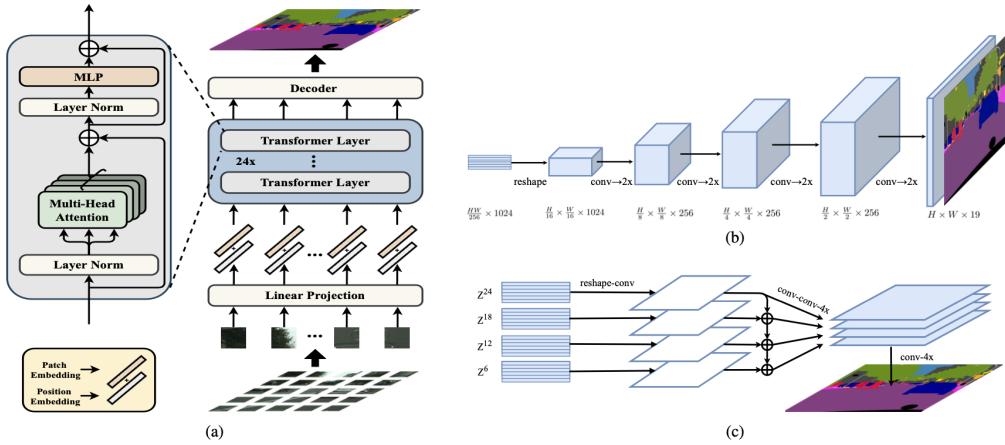


Figure 3: SETR architecture and its variants adapted from [5]. (a) SETR consists of a standard Transformer. (b) *SETR-PUP* with a progressive up-sampling design. (c) *SETR-MLA* with a multi-level feature aggregation.

The SETR encoder (Figure 3a) which is a standard Transformer treats an image as a sequence of patches followed by a linear projection. Then it embeds these projections with patch embedding + position embedding to feed them into a set of Transformer layers. SETR has no down-sampling in spatial resolution at each layer of the encoder transformer while it only provides global context modeling. They classify SETR into a few variants depending on the decoder part of the model; *SETR-PUP* (Figure 3b) which has a progressive up-sampling design and the *SETR-MLA* (Figure 3) which has a multi-level feature aggregation.

SETR achieved state-of-the-art semantic segmentation results on ADE20K, Pascal Context by the time of submission [5]. It has also been tested on the Cityscapes dataset and has shown impressive results.

#### 4.2. *Swin Transformer*

To address the issue of not having a general purpose Transformer backbone for computer vision tasks, [4] proposed Swin Transformer (Hierarchical

Vision Transformer using Shifted Windows) which can be served as a general purpose backbone for computer vision tasks such as image classification and dense prediction.

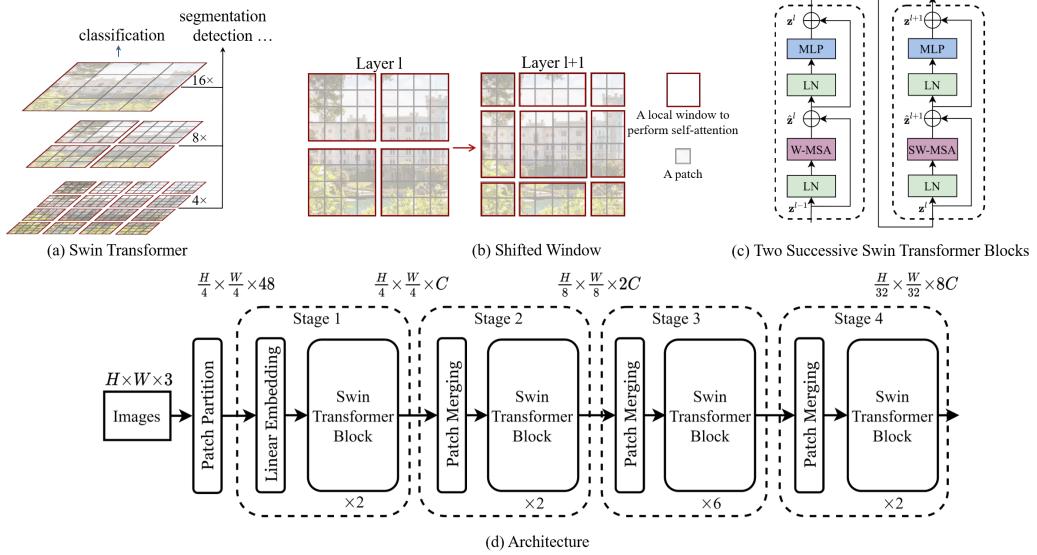


Figure 4: An overview of the Swin Transformer adapted from [4]. (a) Hierarchical feature maps for reducing computational complexity. (b) Shifted window approach which was used when calculating self-attention. (c) Two successive Swin Transformer Blocks which presented at each stage. (d) Core architecture of the Swin.

Swin Transformer was able to bring down the quadratic computational complexity of calculating self-attention in Transformers to linear complexity by constructing hierarchical feature maps (Figure 4a). Also, the shifted window approach illustrated in Figure 4b has much lower latency than the earlier sliding window based approaches which were used to calculate the self-attention. Swin Transformer showed great success over the previous state-of-the-art in image classification (87.3% top-1 accuracy on ImageNet-1K), semantic segmentation (53.5% mIoU on ADE20Kval) and object detection (58.7 box AP and 51.1 mask AP on COCO test-dev) [4].

According to the architecture of a Swin Transformer, in the beginning, it splits the given image into a sequence of non-overlapping patches (tokens) by using the patch partitioning module (Figure 4d). Then a linear embedding is applied to this sequence of patches to project them into an arbitrary dimension. It is followed by several Swin Transformer blocks to apply self-attention. The main responsibility of the patch merging module is to reduce the number of tokens in deeper layers. It is noteworthy that the feature map resolutions in the hierarchical stages are similar to those in typical convo-

lution architectures such as ResNet [68]. Therefore Swin Transformer can efficiently replace ResNet backbone networks in computer vision tasks.

#### 4.3. Segmenter

Segmenter [11] is a purely transformer-based approach for semantic segmentation which consist of a ViT backbone pre-trained on ImageNet and introduces a mask transformer as the decoder (Figure 5). Even though the model was built for segmentation tasks, they take advantage of the models made for image classification to pre-train and then fine-tune them on moderate-sized segmentation datasets.

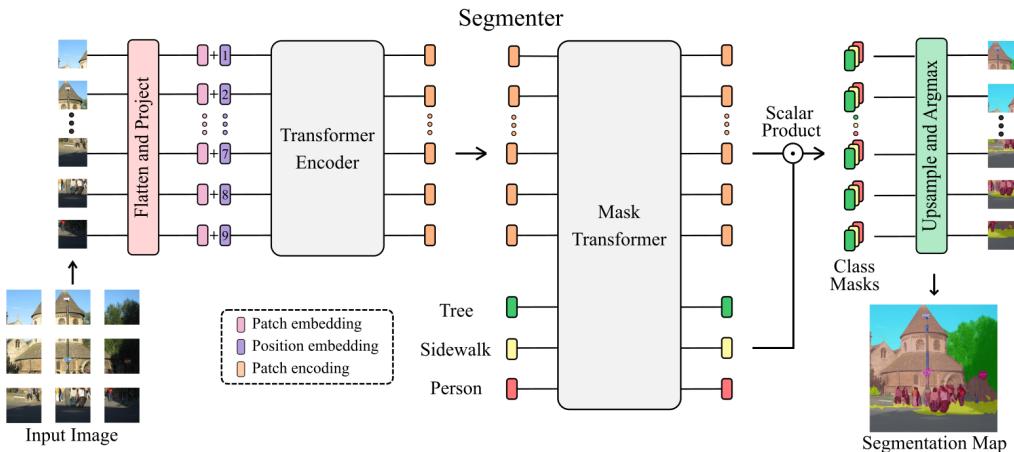


Figure 5: Segmenter architecture adapted from [11]. It basically has a ViT backbone with a mask transformer as the decoder.

CNN-based models are generally inefficient when processing global image context and ultimately result in a sub-optimal segmentation. The reason for the sub-optimal segmentation of the convolution-based approaches is that convolution is a local operation which poorly accesses the global information of the image. But the global information is crucial where the global image context usually influences the local patch labeling. But modeling of global interaction has a quadratic complexity to the image size because it needs to model the interaction between each and every raw pixel of the image. The architecture of the Segmenter especially captures the global context of images, unlike the traditional CNN-based approaches.

Other than the semantic segmentation tasks, this Segmenter model also can be applied to panoptic segmentation (semantic segmentation + instance segmentation) tasks by altering the model architecture. The class embeddings of the model need to be replaced by object embeddings in such a case.

#### 4.4. SegFormer

SegFormer [69] is an architecture for semantic segmentation which consist of a hierarchical Transformer encoder with a lightweight multilayer perceptron (MLP) decoder (Figure 6). The MLP decoder is used for predicting the final mask. To obtain a precise segmentation, it uses a patch size of  $4 \times 4$  in contrast to ViT which uses a patch size of  $16 \times 16$ . It has an overlapped patch merging process to maintain the local continuity around the patches.

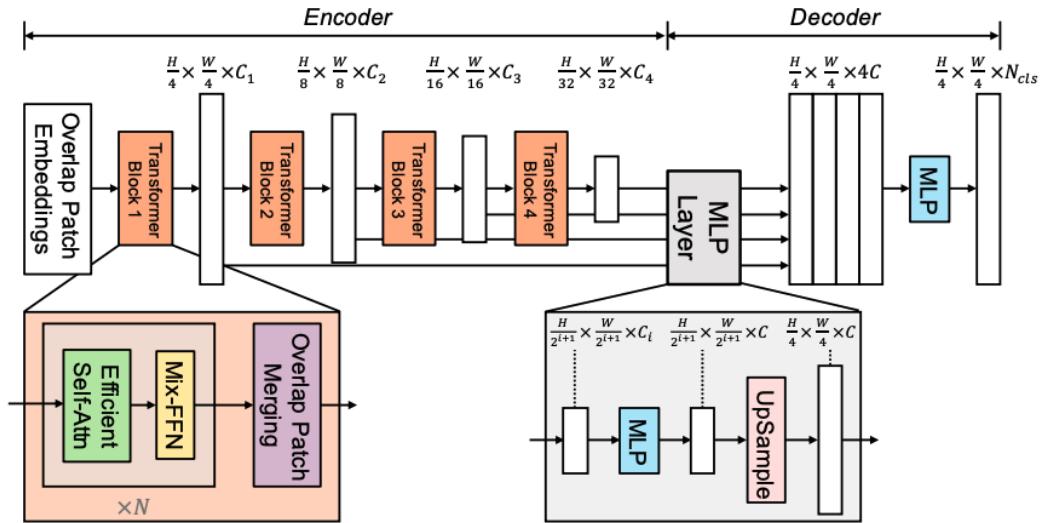


Figure 6: SegFormer architecture adapted from [69]. It has a hierarchical Transformer encoder for feature extraction and a lightweight MLP decoder for predicting the final mask.

Generally, ViT has a fixed resolution for positional encoding [70]. This leads to a drop in accuracy since it needs to interpolate the positional encoding of testing images when they have a different resolution than training images. Thus, SegFormer introduces a Positional-Encoding-Free design as a key feature.

Moreover, the authors claim their architecture is more robust against common corruptions and perturbations than current methods which make SegFormer appropriate for safety-critical applications. SegFormer achieved competitive results on ADE20K, Cityscapes, and COCO-Stuff datasets as shown in Table 2. SegFormer comes in several variants from SegFormer-B0 to SegFormer-B5, where the largest model is SegFormer-B5. This largest model surpasses the SETR [5] on the ADE20K dataset achieving the highest mIoU while being  $4\times$  faster than SETR. All of these SegFormer models have trade-offs between model size, accuracy, and runtime.

#### 4.5. Pyramid Vision Transformer (PVT)

ViT couldn't be directly applicable to dense prediction tasks because its output feature map is single scaled and it generally has a low resolution which comes at a higher computational cost. PVT [71] overcomes the aforementioned concerns by introducing a progressive shrinking pyramid backbone network to reduce the computational costs and simultaneously output more fine-grained segmentation. PVT comes in two variants. PVT v1 [71] is the first work by the authors and PVT v2 [72] comes with some additional improvements to the previous version.

##### 4.5.1. PVT v1

This initial version has some noteworthy changes compared to the ViT. It takes  $4 \times 4$  input patches in contrast to the  $16 \times 16$  patches in ViT. This improves the model's ability to learn high-resolution representations. It also reduces the computational demand of traditional ViT by using a progressive shrinking pyramid. This pyramid structure progressively shrinks the output resolution from high to low in the stages which are responsible for generating the scaled feature maps (Figure 7). Another major difference is that it replaces the multi-head attention layer (MHA) in ViT with a novel spatial reduction attention (SRA) layer which reduces the spatial scales before the attention operation. This further reduces the computational and memory demand because SRA has a low computational complexity than MHA.

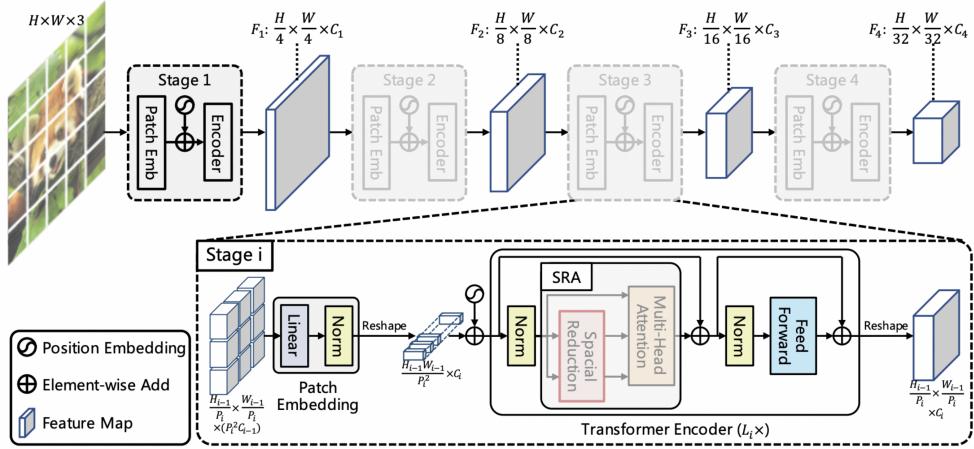


Figure 7: PVT v1 architecture adapted from [71]. The pyramid structure of the stages progressively shrinks the output resolution from high to low.

#### 4.5.2. PVT v2

The former version has a few drawbacks. The computational demand of the PVT v1 is relatively large when processing high-resolution images. It loses the local continuity of the images when processing the image as a sequence of non-overlapping patches. It cannot process variable-sized inputs because of the fixed-size position encoding. This new version has three major improvements which circumvent the previous design issues. First one is linear spatial reduction attention (LSRA) which reduces the spatial dimension of the image to a fixed size using average pooling (Figure 8). Unlike SRA in the PVT v1, LSRA benefits from linear complexity. Second one is the overlapping patch embedding (Figure 9a). This is done by zero-padding the border of the image and taking more enlarged patch windows which overlap with the adjacent windows. It helps to capture more local continuity of the images. The third one is the convolutional feed-forward network (Figure 9b) which helps to process different sizes of input resolutions. With these major improvements, PVT v2 was able to bring down the complexity of PVT v1 to linear complexity.

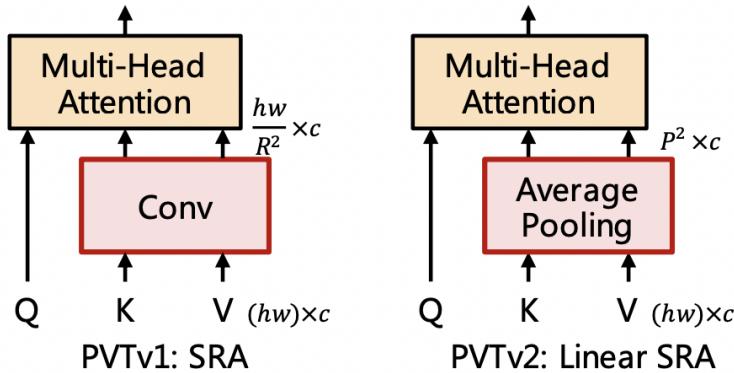


Figure 8: Comparison of spatial reduction attention (SRA) layers in PVT versions [72]

We can clearly see how the improvements of the PVT v2 contribute to higher gains in the benchmark comparison in Table 2.

#### 4.6. Twins

Twins [73] propose two modern Transformer designs for computer vision named Twins-PCPVT and Twins-SVT by revisiting the work on the PVT v1 [71] and Swin Transformer [4].

Twins-SVT uses a spatially separable self-attention (SSSA) mechanism based on the depth-wise separable convolutions in neural networks. This

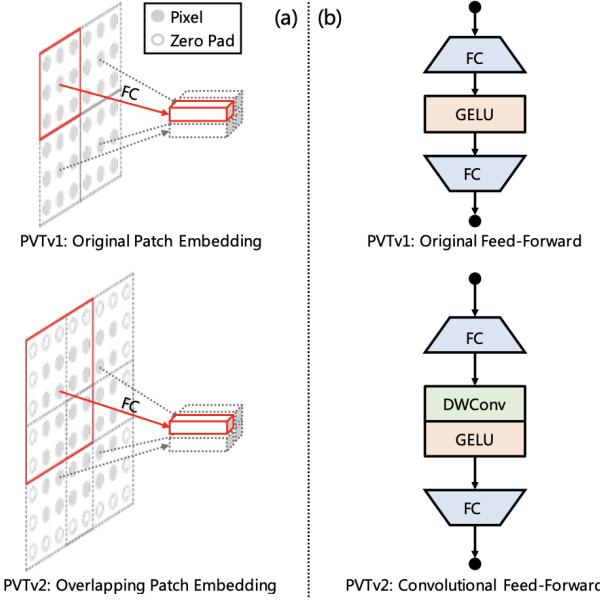


Figure 9: Improved patch embedding and feed-forward networks in PVT v2 [72]

SSSA has two underlying attention mechanisms which are capable of capturing local information as well as global information. Locally grouped self-attention (LSA) and global sub-sampled attention (GSA) are the above-mentioned attention mechanisms respectively. Those techniques greatly reduce the heavy computational demand in high-resolution image inputs while keeping a fine-grained segmentation.

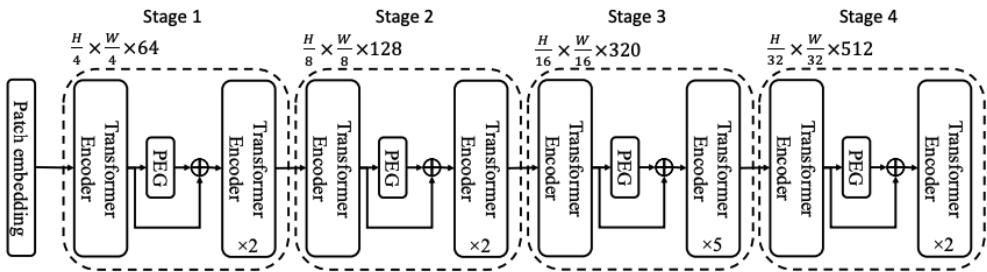


Figure 10: Twins-PCPVT architecture adapted from [73]. It uses conditional position encoding with a positional encoding generator (PEG) to overcome some of the drawbacks of fixed-positional encoding.

As we discussed in the Pyramid Vision Transformer section, PVT v1 can only process fixed-size image inputs due to its absolute positional en-

coding. This hinders the performance of PVT. To alleviate this challenge Twins-PCPVT uses a conditional position encoding (CPE) first introduced in Conditional Position encoding Vision Transformer (CPVT) [70]. This is illustrated as the positional encoding generator (PEG) in Figure 10. It is capable of alleviating some of the issues encountered in fixed-position encoding.

Twins architectures have shown outstanding performance on computer vision tasks including image classification and semantic segmentation. The semantic segmentation results achieved by the two Twins architectures are highly competitive compared to the Swin Transformer [4] and PVT [71].

#### 4.7. Dense Prediction Transformer (DPT)

DPT [74] architecture is introduced with a transformer backbone inside the encoder-decoder design for fine-grained output segmentation predictions compared to the fully convolutional networks. The transformer encoder based on ViT [2] is capable of maintaining spatial resolution over all the stages of the Transformer architecture which is important for dense predictions.

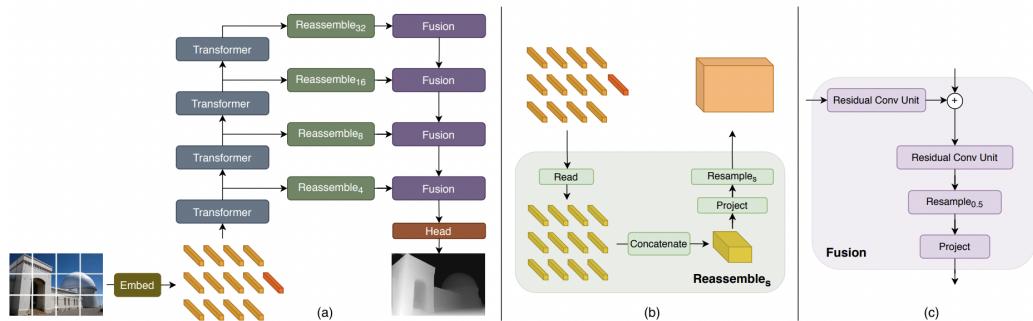


Figure 11: DPT architecture adapted from [74]. (a) Non-overlapping image patches are fed into the Transformer block. (b) Reassemble operation for assembling tokens into feature maps. (c) Fusion blocks for combining feature maps.

In the paper, the authors have introduced several models based on the used image embedding technique. The DPT-Base and DPT-Large models use patch-based embedding where the input image is separated into non-overlapping image patches. Then these are fed into the Transformer block with a learnable position embedding to locate the spatial position of each individual token (Figure 11a). DPT-Base has 12 transformer layers compared to the DPT-Large which has 24 layers with wide feature sizes. The other model is the DPT-Hybrid, which uses the convolutional backbone ResNet-50 as a feature extractor and uses the pixel-based feature maps as token inputs

to the 12-layer transformer block. The Transformer blocks reassemble the tokens with multi-head self-attention (MSA) [1] sequential blocks for global interaction between tokens. The tokens are reassembled into image-like feature representations in various resolutions (Figure 11b). Finally, these representations are combined using residual convolutional units in the decoder and fused together for the final dense prediction (Figure 11c).

The experimental results of the dense prediction transformer have provided improved accuracy results over several benchmark dataset comparisons. The results show that for a large training dataset, the model has the best performance. The comparisons were done for depth estimations and semantic segmentation. ADE20K dataset is used for segmentation and the DPT-Hybrid model has outperformed all the fully-convolutional models [74]. The DPT has the ability to identify precise boundaries of objects with less distortion. The DPT model was also compared with the PASCAL-Context dataset after fine-tuning.

#### 4.8. High-Resolution Transformer (HRFormer)

HRFormer [75] is an architecture model that is built using a depth-wise convolutional design with a Feed Forward Network (FFN) and a local window self-attention mechanism with a multi-resolution parallel transformer module. This model is developed for dense prediction tasks focusing on pose estimation and semantic segmentation. The model outperforms the conventional ViT model which produces low-resolution outputs. The HRFormer is designed to maintain the high-resolution using multi-resolution streams and is more efficient in computational complexity and memory usage.

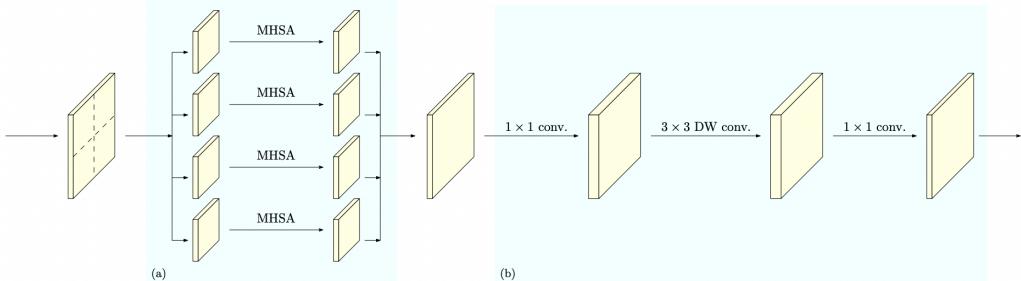


Figure 12: HRFormer architecture adapted from [75]. (a) Self-attention blocks. (b) FFN with depth-wise convolutions.

HRFormer has been incorporated by using the HRNet [76], which is a convolutional network consisting of a multi-scale parallel design. This architecture helps to capture feature maps in variant resolutions while maintaining high resolution. At each of these resolution blocks, partitioning is

done by creating non-overlapping windows, and self-attention is performed on each image window separately. This improved the efficiency significantly compared to overlapping local window mechanisms introduced earlier in different studies [77]. The self-attention blocks (Figure 12a) are followed by an FFN with depth-wise convolutions (Figure 12b) to increase the receptive field size by information exchange between local windows, which is vital in dense prediction. By incorporating a multi-resolution parallel transformer architecture with convolutional multi-scale fusions for the overall HRFormer architecture, the information between different resolutions is exchanged repeatedly. This process creates a high-resolution output with both local and global context information.

#### 4.9. Masked-attention Mask Transformer (*Mask2Former*)

*Mask2Former* [78] is a new transformer architecture that can be leveraged to do segmentation tasks including panoptic, instance, and semantic segmentation. It is a successful attempt to introduce a universal architecture for the segmentation tasks which outperforms the current specialized SOTA architectures for each of the segmentation tasks by the time of submission. Its key components consist of a transformer decoder with masked attention. Generally, a standard Transformer attends to the full feature map. In contrast, the masked attention operator in *Mask2Former* restricts the cross-attention to the foreground region of the predicted mask and then extracts the localized features. This makes the attention mechanism more efficient in this model.

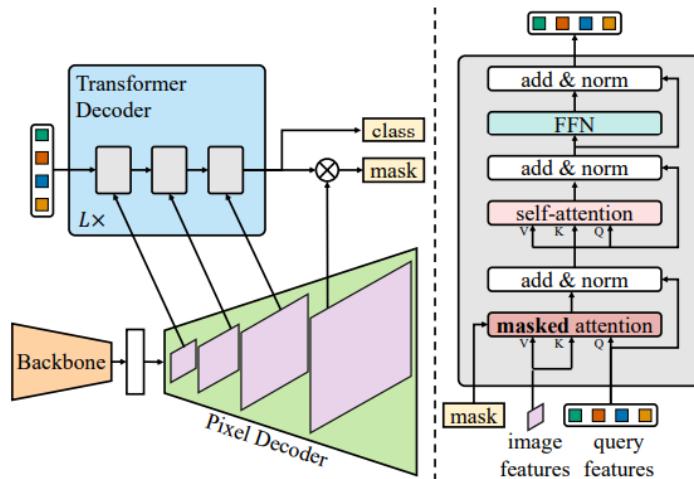


Figure 13: *Mask2Former* architecture adapted from [78]. The model consists of a backbone feature extractor, a pixel decoder, and a Transformer decoder.

The architecture of Mask2Former is similar in design to the previous MaskFormer [79] architecture. The main components are the backbone feature extractor, pixel decoder, and the Transformer decoder (Figure 13). The backbone could be either a CNN-based model or a Transformer based model. As the pixel decoder, they have used a more advanced multi-scale deformable attention Transformer (MSDeformAttn) [6] in contrast to the feature pyramid network [80] used in MaskFormer [79]. Masked attention has been used to enhance the effectiveness of the Transformer decoder.

Despite being a universal architecture for segmentation, Mask2Former still needs to be trained separately for each of the specific tasks. This is a common limitation of the universal architectures for segmentation tasks. Mask2Former has achieved new SOTA performance on all three segmentation tasks (panoptic, instance, semantic) in popular datasets such as COCO and ADE20K and Cityscapes. The semantic segmentation results are compared for ADE20K and Cityscapes datasets in Table 2.

Model	Variant	Backbone	#Params (M)	Datasets		
				ADE20K	Cityscapes	PASCAL-Context
SETR [5]	SETR- <i>Naive</i> (16,160k) <sup>ρ</sup>	ViT-L <sup>†</sup> [2]	305.67	48.06 / 48.80	-	-
	SETR- <i>PUP</i> (16,160k)	ViT-L <sup>‡</sup>	318.31	48.58 / 50.09	-	-
	SETR- <i>MLA</i> (16,160k)	ViT-L <sup>‡</sup>	310.57	<b>48.64 / 50.28</b>	-	-
	SETR- <i>PUP</i> (16,40k)	ViT-L <sup>‡</sup>	318.31	-	78.39 / 81.57	-
	SETR- <i>PUP</i> (16,80k)	ViT-L <sup>‡</sup>	318.31	-	<b>79.34 / 82.15</b>	-
	SETR- <i>Naive</i> (16,80k)	ViT-L <sup>‡</sup>	305.67	-	-	52.89 / 53.61
	SETR- <i>PUP</i> (16,80k)	ViT-L <sup>‡</sup>	318.31	-	-	54.40 / 55.27
	SETR- <i>MLA</i> (16,80k)	ViT-L <sup>‡</sup>	310.57	-	-	<b>54.87 / 55.83</b>
Swin <sup>N</sup> [4]	Swin-T		60	46.1	-	-
	Swin-S		81	49.3	-	-
	Swin-B <sup>‡</sup>		121	51.6	-	-
	Swin-L <sup>‡</sup>		234	<b>53.5</b>	-	-
Segmenter <sup>§</sup> [11]	Seg-B	DeiT-B <sup>†</sup> [81]	86	48.05	80.5	53.9
	Seg-B/Mask	DeiT-B <sup>†</sup>	86	50.08	80.6	55.0
	Seg-L	ViT-L <sup>‡</sup>	307	52.25	80.7	56.5
	Seg-L/Mask	ViT-L <sup>‡</sup>	307	<b>53.63</b>	<b>81.3</b>	<b>59.0</b>
SegFormer [69]	MiT-B0 <sup>†</sup>		3.4	37.4 / 38.0	76.2 / 78.1	-
	MiT-B1 <sup>†</sup>		13.1	42.2 / 43.1	78.5 / 80.0	-
	MiT-B2 <sup>†</sup>		24.2	46.5 / 47.5	81.0 / 82.2	-
	MiT-B3 <sup>†</sup>		44.0	49.4 / 50.0	81.7 / 83.3	-
	MiT-B4 <sup>†</sup>		60.8	50.3 / 51.1	82.3 / 83.9	-
	MiT-B5 <sup>†</sup>		81.4	<b>51.0 / 51.8</b>	<b>82.4 / 84.0</b>	-
PVT <sup>N</sup>	PVT-Tiny <sup>‡</sup>		17.0	35.7	-	-
	PVT-Small <sup>‡</sup>		28.2	39.8	-	-
	PVT-Medium <sup>‡</sup>		48.0	41.6	-	-
	PVT-Large <sup>‡</sup>		65.1	42.1	-	-
	PVT-Large <sup>‡*</sup>		65.1	<b>44.8</b>	-	-
PVT v1 [71]	PVT v2-B0 <sup>†</sup>		7.6	37.2	-	-
	PVT v2-B1 <sup>†</sup>		17.8	42.5	-	-
	PVT v2-B2 <sup>†</sup>		29.1	45.2	-	-
	PVT v2-B3 <sup>†</sup>		49.0	47.3	-	-
	PVT v2-B4 <sup>†</sup>		66.3	47.9	-	-
	PVT v2-B5 <sup>†</sup>		85.7	<b>48.7</b>	-	-
Twins-PCPVT	Twins-PCPVT-S <sup>†</sup>		54.6	46.2 / 47.5	-	-
	Twins-PCPVT-B <sup>†</sup>		74.3	47.1 / 48.4	-	-
	Twins-PCPVT-L <sup>†</sup>		91.5	<b>48.6 / 49.8</b>	-	-
Twins [73]	Twins-SVT-S <sup>†</sup>		54.4	46.2 / 47.1	-	-
	Twins-SVT-B <sup>†</sup>		88.5	47.7 / 48.9	-	-
	Twins-SVT-L <sup>†</sup>		133	<b>48.8 / 50.2</b>	-	-
DPT <sup>§</sup> [74]	DPT-Hybrid	ViT-Hybrid <sup>‡</sup>	123	<b>49.02</b>	-	<b>60.46</b>
	DPT-Large	ViT-L <sup>‡</sup>	343	47.63	-	-
HRFormer [75]	OCRNet(7,150k) <sup>ρ</sup>	HRFormer-S	13.5	44.0 / 45.1	-	-
	OCRNet(7,150k)	HRFormer-B	50.3	<b>46.3 / 47.6</b>	-	-
	OCRNet(7,80k)	HRFormer-S	13.5	-	80.0 / 81.0	-
	OCRNet(7,80k)	HRFormer-B	50.3	-	81.4 / 82.0	-
	OCRNet(15,80k)	HRFormer-B	50.3	-	<b>81.9 / 82.6</b>	<b>57.6 / 58.5</b>
	OCRNet(7,60k)	HRFormer-B	50.3	-	-	56.3 / 57.1
	OCRNet(7,60k)	HRFormer-S	13.5	-	-	53.8 / 54.6
Mask2Former [78]	Swin-T		-	47.7 / 49.6	-	-
	Swin-L <sup>‡</sup>		216	56.1 / 57.3	-	-
	Swin-L-FaPN <sup>‡</sup>		-	<b>56.4 / 57.7</b>	-	-
	Swin-L <sup>‡</sup>		216	-	83.3 / 84.3	-
	Swin-B <sup>‡</sup>		-	-	<b>83.3 / 84.5</b>	-

Table 2: Comparison of the ViT models specialized for the task of semantic segmentation according to mIoU (%) using different benchmark datasets. The best-performing variant of each model for a given dataset is highlighted. Overall top performing model variant for each dataset is shaded in gray. "SS / MS" contains both single-scale and multi-scale inferences. "<sup>N</sup>" - Single-scale inference only, "<sup>§</sup>" - Multi-scale inference only, "<sup>ρ</sup>" - (patch size, iterations), "<sup>†</sup>" - pre-trained on ImageNet-1K, "<sup>‡</sup>" - pre-trained on ImageNet-21K, "\*" - 320K training iterations and multi-scale flip testing

## 5. Discussion

In this survey, we discussed how ViTs became a powerful alternative to classical CNNs in various computer vision applications, their strengths as well as limitations, and how ViT contributed to the semantic segmentation of images with their usage across different domains such as remote sensing, medical and video processing. Even though we included some of the CNN architectures widely used in prior mentioned domains to provide a comparison between the ViT and CNNs, an in-depth discussion about CNN architectures is beyond the scope of this paper. We have summarized the different statistics regarding popular datasets used for semantic segmentation tasks and the results of different ViT architectures used for semantic segmentation to give a clear and high-level overview for the reader around the region of semantic segmentation.

## 6. Conclusions and Future Directions

Unlike mature convolutional neural networks, ViTs are still in the early stage of development. Nevertheless, we observed how powerful and competitive they are with their CNN counterparts. ViTs are progressing towards excellence and it is expected that they will replace traditional CNN-based methods widely used in the deep learning domain in the near future. Different variants of ViTs can be used for experiments with domains such as big data analytics that require a vast amount of data for processing. Exploring research areas with less adaptation to ViT usage can create more efficient, performance-increased outcomes for current implementation methods.

Even though ViTs have proven successful, they can be challenging to experiment with due to their high computational demand. Thus improvements to the ViT architecture are needed to make it lightweight and more efficient. This will inspire the community to open new pathways using ViTs.

We believe there is a plethora of new research areas that ViT, along with semantic segmentation can be applied to solve real-world problems.

## References

- [1] A. Vaswani, G. Brain, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Lukasz Kaiser, I. Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems 30 (2017).
- [2] 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).

- [3] C.-F. R. Chen, Q. Fan, R. Panda, Crossvit: Cross-attention multi-scale vision transformer for image classification, in: Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 357–366.
- [4] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10012–10022.
- [5] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr, et al., Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 6881–6890.
- [6] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable detr: Deformable transformers for end-to-end object detection, arXiv preprint arXiv:2010.04159 (2020).
- [7] X. Dai, Y. Chen, J. Yang, P. Zhang, L. Yuan, L. Zhang, Dynamic detr: End-to-end object detection with dynamic attention, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 2988–2997.
- [8] N. Park, S. Kim, How do vision transformers work?, arXiv preprint arXiv:2202.06709 (2022).
- [9] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9650–9660.
- [10] Z. Xu, W. Zhang, T. Zhang, Z. Yang, J. Li, Efficient transformer for remote sensing image segmentation, Remote Sensing 13 (18) (2021) 3585.
- [11] R. Strudel, R. Garcia, I. Laptev, C. Schmid, Segmenter: Transformer for semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 7262–7272.

- [12] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, M. Shah, Transformers in vision: A survey, ACM computing surveys (CSUR) 54 (10s) (2022) 1–41.
- [13] Y. Liu, Y. Zhang, Y. Wang, F. Hou, J. Yuan, J. Tian, Y. Zhang, Z. Shi, J. Fan, Z. He, A survey of visual transformers, IEEE Transactions on Neural Networks and Learning Systems (2023).
- [14] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, et al., A survey on vision transformer, IEEE transactions on pattern analysis and machine intelligence (2022).
- [15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
- [16] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, et al., Attention u-net: Learning where to look for the pancreas, arXiv preprint arXiv:1804.03999 (2018).
- [17] F. I. Diakogiannis, F. Waldner, P. Caccetta, C. Wu, Resunet-a: A deep learning framework for semantic segmentation of remotely sensed data, ISPRS Journal of Photogrammetry and Remote Sensing 162 (2020) 94–114.
- [18] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, J. Liang, Unet++: A nested u-net architecture for medical image segmentation, in: Deep learning in medical image analysis and multimodal learning for clinical decision support, Springer, 2018, pp. 3–11.
- [19] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 6 (02) (1998) 107–116.
- [20] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, J. Shlens, Stand-alone self-attention in vision models, Advances in Neural Information Processing Systems 32 (2019).
- [21] L. Zhu, J. Suomalainen, J. Liu, J. Hyypä, H. Kaartinen, H. Haggren, et al., A review: Remote sensing sensors, Multi-purposeful application of geospatial data (2018) 19–42.

- [22] M. Schmitt, J. Prexl, P. Ebel, L. Liebel, X. X. Zhu, Weakly supervised semantic segmentation of satellite images for land cover mapping—challenges and opportunities, arXiv preprint arXiv:2002.08254 (2020).
- [23] L. P. Olander, H. K. Gibbs, M. Steininger, J. J. Swenson, B. C. Murray, Reference scenarios for deforestation and forest degradation in support of redd: a review of data and methods, *Environmental Research Letters* 3 (2) (2008) 025011.
- [24] F. Pacifici, F. Del Frate, C. Solimini, W. J. Emery, An innovative neural-net method to detect temporal changes in high-resolution optical satellite imagery, *IEEE Transactions on Geoscience and Remote Sensing* 45 (9) (2007) 2940–2952.
- [25] A. Boguszewski, D. Batorski, N. Ziembka-Jankowska, T. Dziedzic, A. Zambrzycka, Landcover. ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 1102–1110.
- [26] L. P. Osco, J. M. Junior, A. P. M. Ramos, L. A. de Castro Jorge, S. N. Fatholahi, J. de Andrade Silva, E. T. Matsubara, H. Pistori, W. N. Gonçalves, J. Li, A review on deep learning in uav remote sensing, *International Journal of Applied Earth Observation and Geoinformation* 102 (2021) 102456.
- [27] J. Wang, Z. Zheng, A. Ma, X. Lu, Y. Zhong, Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation, arXiv preprint arXiv:2110.08733 (2021).
- [28] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, R. Raskar, Deepglobe 2018: A challenge to parse the earth through satellite images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 172–181.
- [29] L. Ding, D. Lin, S. Lin, J. Zhang, X. Cui, Y. Wang, H. Tang, L. Bruzzone, Looking outside the window: Wide-context transformer for the semantic segmentation of high-resolution remote sensing images, *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022) 1–13.
- [30] S. D. Olabarriaga, A. W. Smeulders, Interaction in the segmentation of medical images: A survey, *Medical image analysis* 5 (2) (2001) 127–142.

- [31] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.
- [32] Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, T. Zhang, S. Gao, J. Liu, Ce-net: Context encoder network for 2d medical image segmentation, *IEEE transactions on medical imaging* 38 (10) (2019) 2281–2292.
- [33] H. Huang, L. Lin, R. Tong, H. Hu, Q. Zhang, Y. Iwamoto, X. Han, Y.-W. Chen, J. Wu, Unet 3+: A full-scale connected unet for medical image segmentation, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, pp. 1055–1059.
- [34] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, Y. Zhou, Transunet: Transformers make strong encoders for medical image segmentation, *arXiv preprint arXiv:2102.04306* (2021).
- [35] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, M. Wang, Swin-unet: Unet-like pure transformer for medical image segmentation, in: Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III, Springer, 2023, pp. 205–218.
- [36] A. İşin, C. Direkoglu, M. Şah, Review of mri-based brain tumor image segmentation using deep learning methods, *Procedia Computer Science* 102 (2016) 317–324.
- [37] N. C. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, et al., Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic), in: 2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018), IEEE, 2018, pp. 168–172.
- [38] P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser, et al., The liver tumor segmentation benchmark (lits), *arXiv preprint arXiv:1901.04056* (2019).
- [39] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al., The multi-modal brain tumor image segmentation benchmark (brats), *IEEE transactions on medical imaging* 34 (10) (2014) 1993–2024.

- [40] D. Gorecky, M. Schmitt, M. Loskyll, D. Zühlke, Human-machine-interaction in the industry 4.0 era, in: 2014 12th IEEE international conference on industrial informatics (INDIN), Ieee, 2014, pp. 289–294.
- [41] R. T. Azuma, A survey of augmented reality, *Presence: teleoperators & virtual environments* 6 (4) (1997) 355–385.
- [42] J. Janai, F. Güney, A. Behl, A. Geiger, et al., Computer vision for autonomous vehicles: Problems, datasets and state of the art, *Foundations and Trends® in Computer Graphics and Vision* 12 (1–3) (2020) 1–308.
- [43] T. Gevers, A. Smeulders, Image search engines: An overview, *Emerging Topics in Computer Vision* (2004) 1–54.
- [44] S. Jain, X. Wang, J. E. Gonzalez, Accel: A corrective fusion network for efficient semantic segmentation on video, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8866–8875.
- [45] M. Ding, Z. Wang, B. Zhou, J. Shi, Z. Lu, P. Luo, Every frame counts: Joint learning of video segmentation and optical flow, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 10713–10720.
- [46] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3213–3223.
- [47] G. J. Brostow, J. Fauqueur, R. Cipolla, Semantic object classes in video: A high-definition ground truth database, *Pattern Recognition Letters* 30 (2) (2009) 88–97.
- [48] S. R. Richter, V. Vineet, S. Roth, V. Koltun, Playing for data: Ground truth from computer games, in: European conference on computer vision, Springer, 2016, pp. 102–118.
- [49] E. Shelhamer, K. Rakelly, J. Hoffman, T. Darrell, Clockwork convnets for video semantic segmentation, in: European Conference on Computer Vision, Springer, 2016, pp. 852–868.
- [50] X. Zhu, Y. Xiong, J. Dai, L. Yuan, Y. Wei, Deep feature flow for video recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2349–2358.

- [51] B. Mahasseni, S. Todorovic, A. Fern, Budget-aware deep semantic video segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1029–1038.
- [52] A. Garcia-Garcia, S. Orts-Escalano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, J. Garcia-Rodriguez, A survey on deep learning techniques for image and video semantic segmentation, *Applied Soft Computing* 70 (2018) 41–65.
- [53] Y. Wang, Z. Xu, X. Wang, C. Shen, B. Cheng, H. Shen, H. Xia, End-to-end video instance segmentation with transformers, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 8741–8750.
- [54] S. Yang, X. Wang, Y. Li, Y. Fang, J. Fang, W. Liu, X. Zhao, Y. Shan, Temporally efficient vision transformer for video instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2885–2895.
- [55] J. Wu, Y. Jiang, S. Bai, W. Zhang, X. Bai, Seqformer: Sequential transformer for video instance segmentation, in: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII, Springer, 2022, pp. 553–569.
- [56] S. Gustavsson, Object detection and semantic segmentation using self-supervised learning (2021).
- [57] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, J. Tang, Self-supervised learning: Generative or contrastive, *IEEE Transactions on Knowledge and Data Engineering* (2021).
- [58] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.
- [59] F. Milletari, N. Navab, S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 fourth international conference on 3D vision (3DV), IEEE, 2016, pp. 565–571.
- [60] S. Jadon, A survey of loss functions for semantic segmentation, in: 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), IEEE, 2020, pp. 1–7.

- [61] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Ur-tasun, A. Yuille, The role of context for object detection and semantic segmentation in the wild, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 891–898.
- [62] M. Everingham, J. Winn, The pascal visual object classes challenge 2012 (voc2012) development kit, Pattern Anal. Stat. Model. Comput. Learn., Tech. Rep 2007 (2012) 1–45.
- [63] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ade20k dataset, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 633–641.
- [64] S. Kuutti, R. Bowden, Y. Jin, P. Barber, S. Fallah, A survey of deep learning applications to autonomous vehicle control, IEEE Transactions on Intelligent Transportation Systems 22 (2) (2020) 712–733.
- [65] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, C. Jawahar, Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments, in: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2019, pp. 1743–1751.
- [66] A. Gaidon, Q. Wang, Y. Cabon, E. Vig, Virtual worlds as proxy for multi-object tracking analysis, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 4340–4349.
- [67] E. Alberti, A. Tavera, C. Masone, B. Caputo, Idda: a large-scale multi-domain dataset for autonomous driving, IEEE Robotics and Automation Letters 5 (4) (2020) 5526–5533.
- [68] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
- [69] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, P. Luo, Segformer: Simple and efficient design for semantic segmentation with transformers, Advances in Neural Information Processing Systems 34 (2021) 12077–12090.
- [70] X. Chu, Z. Tian, B. Zhang, X. Wang, X. Wei, H. Xia, C. Shen, Conditional positional encodings for vision transformers, arXiv preprint arXiv:2102.10882 (2021).

- [71] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pyramid vision transformer: A versatile backbone for dense prediction without convolutions, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 568–578.
- [72] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pvt v2: Improved baselines with pyramid vision transformer, Computational Visual Media 8 (3) (2022) 415–424.
- [73] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, C. Shen, Twins: Revisiting the design of spatial attention in vision transformers, Advances in Neural Information Processing Systems 34 (2021) 9355–9366.
- [74] R. Ranftl, A. Bochkovskiy, V. Koltun, Vision transformers for dense prediction, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 12179–12188.
- [75] Y. Yuan, R. Fu, L. Huang, W. Lin, C. Zhang, X. Chen, J. Wang, Hrformer: High-resolution vision transformer for dense predict, Advances in Neural Information Processing Systems 34 (2021) 7281–7293.
- [76] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al., Deep high-resolution representation learning for visual recognition, IEEE transactions on pattern analysis and machine intelligence 43 (10) (2020) 3349–3364.
- [77] H. Hu, Z. Zhang, Z. Xie, S. Lin, Local relation networks for image recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3464–3473.
- [78] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 1290–1299.
- [79] B. Cheng, A. Schwing, A. Kirillov, Per-pixel classification is not all you need for semantic segmentation, Advances in Neural Information Processing Systems 34 (2021) 17864–17875.
- [80] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.

- [81] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, Training data-efficient image transformers & distillation through attention, in: International Conference on Machine Learning, PMLR, 2021, pp. 10347–10357.