A Review of Semantic Segmentation, Vision Transformers and Domain Adaptation

Moritz Bergemann

19759948

Literature Review

School of Electrical Engineering, Computing and Mathematical Sciences

Curtin University

Australia

June 2022

Contents

List of Figures

List of Tables

Chapter 1

Literature Review

1.1 Semantic segmentation

Semantic segmentation is a core task in computer vision that involves performing classification on every pixel in an input image. Unlike other computer vision tasks such as image classification (identifying which of a set of classes an input image belongs to) or object detection (identifying, classifying, and locating objects in an image), semantic segmentation provides highly dense and semantically rich information, particularly about the shapes objects take up and the intersections between objects.

Figure 1.1: An example of an annotated semantic segmentation image from the Cityscapes dataset [1]. The left is the input to the model, and the right is the expected output.

Segmentation is a key pre-processing task for many applications. Use in self-driving cars and robotics is most often cited. To safely drive, a self-driving car must be able to identify the shape, course, and edges of the road in front of it rather than only

identifying it as a road, for instance. Medical analysis is another key segmentation task that can helps automate the essential task of medical identification and analysis. Many publications focus exclusively on the segmentation of CT or MRI images for medical applications [16]. Other applications include satellite mapping, video surveillance, augmented reality, and domain transformation [17].

Semantic segmentation is distinguished from other computer vision tasks through its density and computational complexity. Density is key in segmentation as the output is pixel-wise - therefore, a significant amount of information must be known about each pixel in the image to accurately segment it, especially for the boundaries between images. The increase in computational cost is also intuitive. No matter the scenario, generating an output that has the same dimensions as the model input will be more computationally intensive than a classification model that simply outputs a single vector.

As with most computer vision applications, deep convolutional neural networks (CNNs) have traditionally achieved state-of-the-art results on semantic segmentation tasks for accuracy and efficiency. The superior performance of CNNs is often cited to be due to their possession of vision-specific inductive biases. Translational equivariance means that patterns present anywhere in an image will be extracted in the same way due to the constant convolution kernel that "slides" across the input. Locality means that CNNs, due to the limited size of each kernel, will inherently put stronger weights towards input features that are closer together in 2D space (figure 1.3) [2]. All of this is achieved with fixed-size kernels, meaning the number of parameters that must be trained for any model does not increase with image size. Novel advances in general computer vision are typically first made on the simpler image classification task, then transferred to other tasks like segmentation.

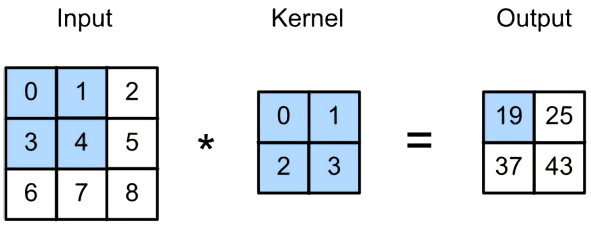

Figure 1.2: A single step in the 2D convolution of an input tensor with a kernel [2]. The next stage would be the computation of 25 via $1 \times 0 + 2 \times 1 + 4 \times 2 + 5 \times 3$.

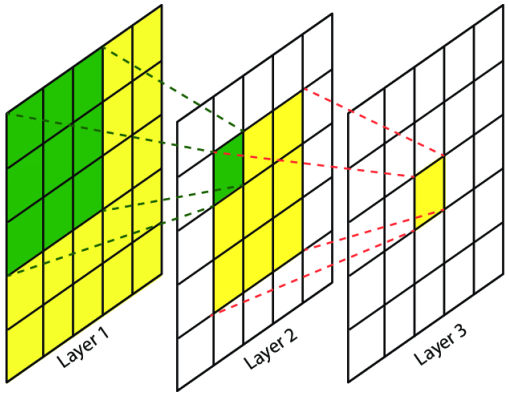

Figure 1.3: A visualisation of locality across CNN layers [3] - the neuron in each layer is only defined by the spatially close neurons in the previous layer. This is also demonstrates CNNs' limited receptive field.

Many initial segmentation approaches used CNNs in a multi-stage process, applying them only for some components. [18] use CNNs as the feature extractor, to which an SVM is applied for region proposal, followed by decision trees for segmentation. R-CNN [19] uses traditional machine learning approaches to first compute region proposals which are then passed through a CNN for classification or segmentation. However, the majority of recent publications find more success using an end-to-end approaches that can be trained via a single back-propagation pass.

FCN (Fully Convolutional Network) [20] was the first model to introduce the concept of using state-of-the-art deep classification networks as "backbones" for segmentation models. The classification models (commonly VGGNet [21], ResNet [22], or Xception [23] in convolutional models) are modified to act as feature extractors, retrieving high-level representations of the input image which can then be used for segmentation rather than classification. This approach, common in many state-of-the-art segmentation models, takes advantage of the backbone's proven architectures and their strong feature extraction abilities due to pre-training on much larger classification datasets such as ImageNet [24], or more recently JFT [25]. Though state-of-the-art, FCN struggled with the learning of global context and with feature localisation (important for dense prediction). These issues were addressed in early approaches through global pooling [26] and conditional random fields [27].

Most semantic segmentation approaches follow some kind of encoder-decoder

structure [2]. Common in many image-to-image and sequence-to-sequence deep learning tasks, this approach consists of two parts. The encoder first maps the input sequence to an abstract representation, typically known as a feature space. The backbones introduced in FCN may be considered encoders, for instance. The decoder is the second component, which reconstructs an output image from this representation (in this case a segmentation map). U-Net [28], ubiquitous in medical segmentation applications, is an early example of a backbone-less encoder-decoder for segmentation.

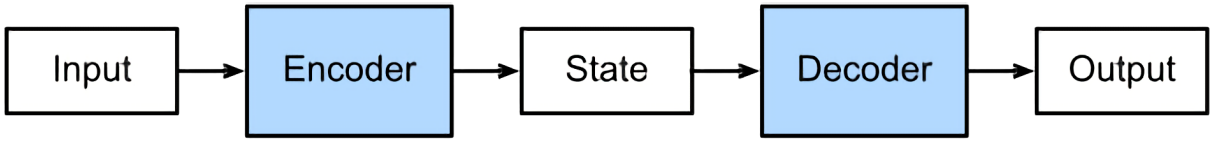

Figure 1.4: The encoder-decoder architecture [2].

Extraction of features at multiple scales is also important for dense prediction - a car in the distance will be differently sized to a car in the foreground, but both should be labelled accurately. This can be addressed by feeding in and parsing inputs at multiple scales, as in RefineNet [29] or using multi-scale pooling techniques. [30] introduced the Spatial Pyramid Pooling (SPP) module, which pools input features to multiple resolutions and then applies convolutions to the output to produce a multi-scale feature vector. PSPNet [4] applied this approach to semantic segmentation by producing, upsampling, and then appending the multi-scale feature maps while maintaining shape (figure 1.5) to produce segmentation results.

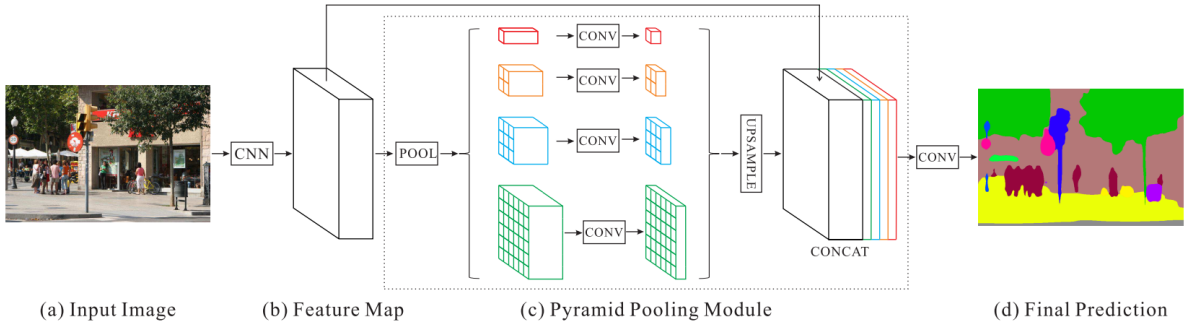

Figure 1.5: Architecture of PSPNet's Pyramid Pooling Module [4], which extracts features at multiple scales.

In recent years, the DeepLab series of models have set the baseline for efficient and performant semantic segmentation. DeepLab [27] introduced the atrous convolu-

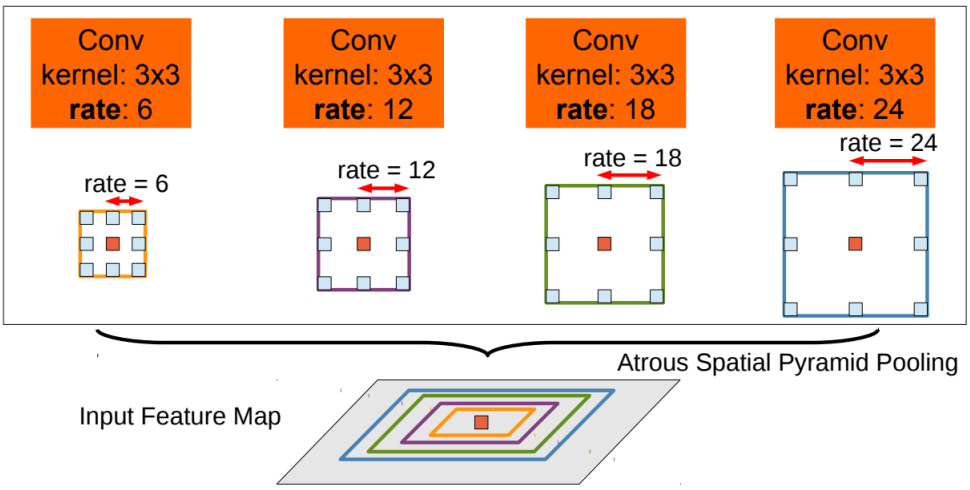

Figure 1.6: Visualisation of atrous spatial pyramid pooling [5].

tion for dense prediction tasks. Atrous convolutions introduce gaps between the weights used to inform the next layer's pixels, allowing for a larger receptive field without significantly increasing model parameters. DeepLabV2 [5] builds on this via Atrous Spatial Pyramid Pooling (ASPP), which combines PSPNet's [4] multi-scale features with a wider receptive field (figure 1.6). DeeplabV3 [31] introduces a deeper architecture and adds a resizing component to ASPP that better suits atrous convolution's large receptive field. DeepLabV3+ [32] introduced an encoder-decoder design with DeepLabV3 as the encoder, and applies a simple decoder to better refine class boundaries. DeeplabV3+ achieves 82.1% mIoU on the Cityscapes dataset [1], a common benchmark for segmentation models.

1.1.1 Fast semantic segmentation

The density and number of predictions (typically one per input pixel) required for semantic segmentation makes it one of the most computationally expensive computer vision tasks. The large number of predictions in the output layer itself is computationally expensive, but dense predictions also typically require a higher-resolution feature map, further increasing memory and compute requirements. Many state-of-the-art segmentation models are therefore only capable of running at a speed of 2-3 predictions a second on modern hardware [6]. Since many uses of segmentation, such as self-driving cars and robotics, require segmentation to be performed in real-time, extensive research has been done to

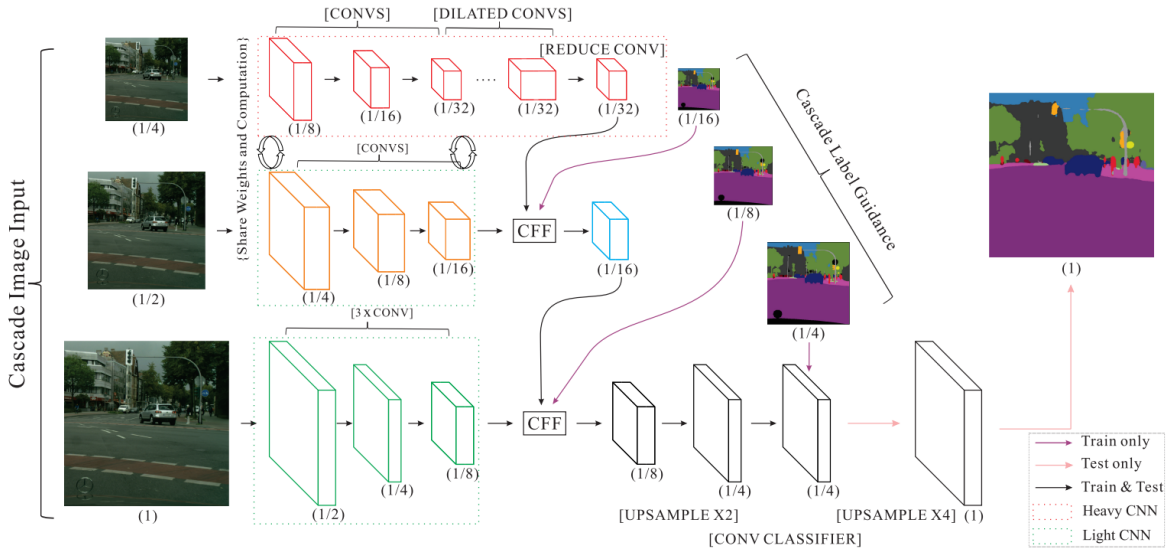

Figure 1.7: ICNet architecture [6].

improve the efficiency of semantic segmentation.

ENet (Efficient Net) [33] was one of the first models to explore low-resource segmentation, focusing on reasonable accuracy capable of running in real-time on mobile devices. It achieves this by making select sacrifices to minimise model parameters and computations while mitigating drops in accuracy. Input features are quickly downsampled to a low resolution ($\frac{1}{8}$th input) with a small number of features, as minimising information density early in the model (forcing the model to "compress" image information) was found to only minimally reduce performance. The model also minimises the size of the decoder, and employs atrous and factorised convolutions to increase receptive field without increasing computation. E-Net achieves 58.3% mIoU on the Cityscapes test set with 0.37M parameters, compared to the then state-of-the art DeepLab's 63.1% mIoU with 134.3M parameters.

ICNet (Image Cascade Network) [6] in 2018 was the next major model to tackle the fast segmentation problem. Initial approaches had an inherent trade-off: reduced input size increases speed, as less convolutions must be performed, but reduces segmentation accuracy, as there are less fine details. ICNet addresses this by taking in the same input image at different scales. High-resolution inputs are parsed through low-filter-count layers that are lightweight but still capable of extracting low-level details like edge and texture. Low-resolution inputs are parsed through more expensive high-filter-count layers

capable of extracting high-level object information (e.g. "these pixels represent a car"), where the low resolution keeps computation time low. The computed features are then fused together through multiple "cascades" to produce the final features. This approach - delegating high-resolution branches to low-level details and vice versa - has become a staple of fast segmentation. BiSeNet [34] extends ICNet's multi-resolution approach, finding that two 'branches', one high-resolution (the "spatial path") and one low resolution (the "context path") is most effective.

Many efficient segmentation models rely on innovations from other computer vision domains. The MobileNet [35] classification models introduced the depthwise separable convolution block, which splits a standard convolution into a (far less expensive) combination of pointwise and depthwise convolution with minimal accuracy loss. MobileNetV2 [36] introduces the bottleneck residual block, which performs efficient convolutions by transforming inputs into a high-dimensional manifold. ContextNet [37] uses a two-branch architecture like BiSeNet [34], and introduces depthwise separable convolutions and bottleneck residual blocks for the high (shallow) and low (deep) resolution branches respectively. ContextNet's successor, Fast-SCNN [7], has become a benchmark for fast segmentation performance. Fast-SCNN replaces the high-resolution branch with a "learning-to-downsample" module that downsamples the input, learns simple high-resolution features, and is used as input to the deep branch. The shallow high-resolution information is then simply fed into the final output features via a skip-connection (figure 1.8)

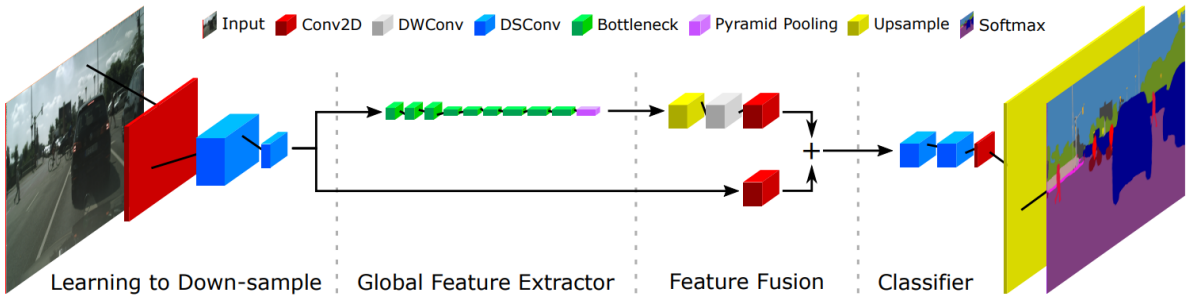

Figure 1.8: Fast-SCNN architecture [7].

1.1.2 Transformers for semantic segmentation

The transformer architecture revolutionised the deep learning field in 2017 [38]. Initially applied for sequence-based natural language processing (NLP) tasks, transformers propose a fundamentally different approach to CNNs. Transformers are based on the concept of multi-head attention. Given two sequences A and B (sequences of words in the initial paper), attention computes the importance of each element in one sequence to each element in the other. This is done by retrieving a query Q from B, and a key K and value V from B - all computed through trainable linear layers. Self-attention then effectively applies a mask to V based on the importance between Q and K. Self-attention is an instance of this where Q, K, and V are computed from the same input, meaning the computed mask defines the importance between each item in the input sequence. "Multi-Head" implies this process is performed multiple times with different linear layers to compute Q, K, and V, similarly to having multiple filters for different features in CNNs.

In the same way that sentences can be considered an ordered sequence of words, an image could be considered an ordered sequence of pixels for use with a transformer (figure 1.9). While various works have explored the integration of self-attention into CNN-based architectures, ViT [8] was the first attempt to produce a fully transformer-based vision architecture (as in [38]) modified for image inputs. Rather than encoding each pixel into the input sequence (which would be prohibitively expensive), ViT groups the input image into 16×16 patches for encoding into the input sequence. Transformers became ubiquitous in NLP due to their ability to map long-distance relationships and their extreme scalability with increasing amounts of data [39] [40]. However, transformers lack the translational equivariance and locality that are believed to make CNNs so effective for vision, and experiments indeed showed that ViT is outperformed by state-of-the-art CNNs when trained on smaller datasets. However, when trained on extremely large datasets, the scalability of transformers appears to win out against these biases.

Other models have since built on ViT. Most significantly for segmentation, Pyramid Vision Transformer (PVT) [41] modifiers ViT to be more suitable for dense prediction, implementing 4×4 instead of 16×16 image patches for increased feature detail. To maintain performance (and develop multi-scale features), it then progressively merges patches

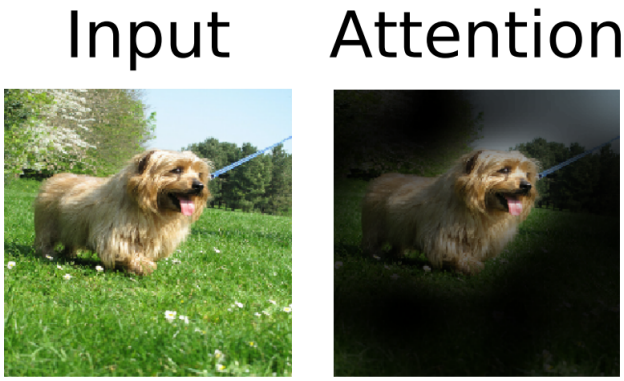

Figure 1.9: Visualisation of self-attention for images in ViT [8]. Brighter pixels had greater importances computed between them by the transformer.

between transformer blocks. DeIT [42] proposes a student-teacher and distilled training approach that allows ViT's architecture to perform strongly even when pre-trained on far smaller datasets. Swin Transformer [43] builds hierarchical feature maps like PVT by merging image patches, but only computes self-attention within each window for efficiency. Twins [44] combines locally-grouped and global self-attention to achieve state-of-the-art performance.

While CNNs need to gradually reduce image resolution to effectively increase receptive field, transformers have possess global attention. Significant relationships within the input sequence can be learned at all parts of the model regardless of their distance from one another. [45] argues that this maintaining of resolution makes transformers especially suited for dense prediction. Their architecture, SETR, uses an encoder-decoder architecture with a pretrained traditional vision transformer (such as ViT or DeIT) as a backbone. Multiple decoders were assessed, with a progressive upsampling and convolution-based approach proving most effective. Most notably (as with ViT), it was found that backbone pre-training was essential. Without pre-training, SETR achieved only 42% mIoU on Cityscapes, worse than models with 1% the parameters [33]. However, with pre-training, SETR achieved a state-of-the-art 82.15% mIoU on Cityscapes. This is further evidence that, while ineffective with small data, vision transformers are capable of learning a far larger feature distribution and are therefore more transferable.

SETR demonstrated the capacity of transformers in segmentation, but was limited by its use of a backbone designed for classification. SegFormer [46] addressed a

number of these issues using it's Mix Transformer Encoder (MiT). First, dense 4×4 pixel patches and progressive patch merging are used as in PVT [41]. Rather than pure-MLP layers after each attention block, SegFormer employs MLPs mixed with 3×3 convolutions without zero-padding. This leaks spatial spatial information to the model that bypasses the need for positional encoding. Combined with a simple all-MLP encoder, possible due to the transformer's receptive field, SegFormer achieves a state-of-the art 51.0% mIoU on the new, challenging ADE20K dataset [47].

1.1.3 Fast segmentation with transformers

Due to the success of transformer architectures in segmentation, recent research has explored the use of transformers for efficient, low-memory segmentation. Some approaches look to modify existing segmentation transformer architectures. [48] dynamically prunes the Q, K, and V neurons of SegFormer's [46] parameters by removing all but the $r\%$ most activated. A knowledge distillation [49] approach is then used to bridge the gap between the original and pruned SegFormer. The un-pruned model becomes the 'teacher' and the pruned the 'student' - at each SegFormer block, the mean-squared error between the outputs of the student and teacher are added to the loss function.

MobileViT [9] sought to modify the ViT architecture to reintroduce some of CNN's inductive biases and therefore reduce the parameters required for strong results. Rather than treating the input as a sequence of patch embeddings, MobileViT blocks take input and produce output as a $H \times W \times C$ tensor as CNNs do (figure 1.10). This is then transformed to a ViT-like patch-embedding, where attention is applied along the size of the patches to avoid the loss of positional information that removes inductive bias. MobileViT was found to outperform MobileNetV2 across computer vision tasks. TopFormer [50] uses a U-Net-like [28] CNN-based encoder-decoder architecture where self-attention is applied only to the deepest features to minimise the required computation.

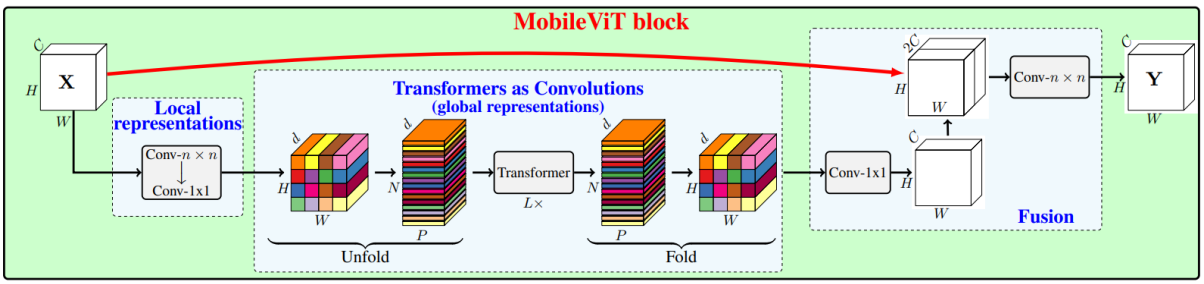

Figure 1.10: Architecture of a MobileViT block [9]. Features are input in CNN-like $H \times W \times C$ format, unfolded into a transformer input sequence, then re-folded into before being output.

1.2 Domain adaptation

Domain adaptation is a diverse area of deep learning, as it seeks to address one of the greatest ongoing issues in the field - the lack of large amounts of high-quality training data for most tasks. Definitions of domain adaptation vary across literature, but in general, domain adaptation can be considered a subset of transfer learning: the transferring of learned knowledge about one domain to a new domain. To explain transfer learning, consider we are building a model to classify images with only a small amount of training data, which we denote the target dataset. We find our model cannot sufficiently learn high-level representations of the target dataset from the available training data, and will likely overfit to a local minima in the training set. However, if we first train the model on a much larger dataset such as ImageNet (denoted the source set), we will produce a model that has learned many features of the classes in this dataset, but also robust interpretations of common image features in general. If we initialise the model for training on the target set with the weights learned from the source set, we can intuit that the model will begin learning to classify the target set much closer to a global optimum for the task. This kind of transfer learning, using appropriate datasets, has been found to improve neural network performance almost universally [25].

Put more formally, consider the source dataset D_S consisting of inputs X_S and labels Y_S as samples from an overall distribution P_S. Similarly, the target set D_T consists of X_T and Y_T from a distribution P_T. Due to the domain shift between the datasets, we can intuit $P_T \neq P_S$. Therefore, transfer learning involves strategies for moving from

P_S to P_T given our intuition P_S will be a strong starting point for learning P_T. Domain adaptation, then, is an extension of this task where the target dataset's labels are not available [51] - Y_T is not known, making this an unsupervised learning problem.

Domain Adaptation is an extremely common and practical problem in deep learning due to the high cost of labelling images for supervised training, especially in semantic segmentation. Each finely labelled image in the Cityscapes dataset, for instance, took human volunteers over 1.5 hours to produce on average [1]. Therefore, extensive research has been performed on the topic across all deep learning domains, including computer vision and semantic segmentation.

Domain adaptation can massively improve the accessibility of domain adaptation. The majority of datasets for machine learning tasks, especially datasets with small and detailed labels like semantic segmentation [1] [27] [52], are sourced from first-world countries where most machine learning research takes place. This means these demographics will disproportionately see the benefits of machine learning technologies. A self-driving car trained primarily on datasets from Europe and North America, for instance, would be prone to far more issues if operated without adjustment in other areas of the world. Domain adaptation makes it easier for research groups and demographics with less access to large datasets to see similar benefits, which can be safety-critical depending on the application.

Another important distinction is between feature-level (i.e. domain-level) and label-level (i.e. task-level) adaptation. Feature-level is more common in domain adaptation, which is simply where the source and target distributions differ ($P_T \neq P_S$) [51]. Label-level adaptation has the set of labels to apply to the source and target dataset differ - a problem that requires vastly different approaches. This review discusses research into feature-level domain adaptation.

1.2.1 Domain adaptation approaches in deep learning

Many approaches have been proposed for transferring strong predictive power from a source domain to an unlabeled target domain. A large subset of strategies focus on

aligning the feature space between the source and target dataset - if a model trained on the source set can produce similar features on the target set, we intuit it will likely be better at classifying/decoding these features for the target domain.

One approach to this is "domain invariant feature learning" [51] - explicitly encouraging the model encoder to produce similar feature representations for source and target inputs by adding a new component to the loss. Some approaches compute difference as the distance between the distributions of the features. This can be evaluated through metrics like maximum mean discrepancy (MMD) [53] or correlation alignment (CORAL) [54]. [55], for instance, produce a model with two parallel streams (for source/target data respectively), and employ MMD between extracted features in the loss to adapt from synthetic to real drone detection data.

Recently, many approaches have applied adversarial approaches to domain invariant feature learning. Adversarial networks [56] are a subclass of deep neural networks that make two models compete in a zero-sum game. In a typical scenario, a generator model will produce outputs in an attempt to 'trick' a discriminator, which takes in the generator's output, into making the wrong prediction. The performance of each model is then used to negatively affect the other's loss - both attempt to minimise the performance of the other. A common example is the creation of artificial faces [57] - the generator produces artificial faces, while the discriminator takes in images and attempts to classify whether they are real or produced by the generator. Both models then compete against each other in alternate training until convincing face generation is achieved. [58] employ an adversarial discriminator model applied to a model's encoder output, which tries to distinguish whether the features were produced from the source or target distribution. This model's performance is then added to the main model loss (via gradient reversal), encouraging the model to produce source and target domain features that are indistinguishable from one another.

Other approaches move domain invariance a step backwards by adapting the labelled source input images themselves to cross the domain gap and match the target distribution [51] (figure 1.11). This is most commonly achieved using GANs [59] [11]. Alternatively, training can be done on labelled source images transferred to match the target distribution for inference on the unmodified target distribution images. A key

Figure 1.11: An example of GTA5 [10] to Cityscapes [1] domain adaptation using CyCADA [11].

assumption in this approach is that the differences in the source/target domain are low-level [51].

Pseudo-labelling [51], sometimes called transductive learning [60], is another core approach. It is based on the idea that the highest quality/most "confident" predictions on the unlabelled target data will provide more valuable than non-valuable knowledge to the model. [60] analyse this in the context of information gain - given an appropriate confidence, choosing the $r\%$ most confident predictions on the target set and adding these to the training set will provide positive information gain. Whether a prediction is considered 'confident' can be determined by using an ensemble of diverse models. If a prediction is consistent in a high proportion of the ensemble, we can intuit it is more likely to be correct. Many approaches, especially in semantic segmentation however, instead assume that the magnitude of a model's softmax prediction can be used as a measure of confidence and therefore pseudo-labelling [12]. Recent classification approaches such as SHOT [61] instead produce pseudo-labels via k-means clustering followed by k-nearest neighbours classification on the zero-shot [62] target dataset predictions, and achieve state-of-the-art results.

Many authors combine a number of the above and other approaches. Target discriminative approaches, for instance, take advantage of the cluster assumption [63], which states that features with common labels should be clustered together in the feature space. Therefore, class boundaries should lie in the low density regions between clusters. Numerous adversarial approaches have been explored to move the decision boundary into

these low-density regions [51].

1.2.2 Domain adaptation for semantic segmentation

As a task with especially challenging label production [1], domain adaptation for semantic segmentation has been explored extensively in recent years. Many approaches focus on adaptation from far more easily attainable synthetic segmentation datasets (such as GTA5 [10] and SYNTHIA [64]) to real-world datasets (such as Cityscapes [1], where labels are considered missing and only used for benchmarking).

As with general semantic segmentation, many developments for segmentation DA are adapted and inspired from developments in simpler computer vision tasks. Approaches described in the previous section have been applied to the semantic segmentation task in various forms. However, early approaches found domain adaptation for segmentation is far more challenging due to the increased complexity and scale of the task [65]. Using FCN, [66] addresses these issues by learning from both a pixel-wise domain discriminator loss and a class-wise discriminator loss as multiple classes will likely be present in each prediction. This work is often considered the earliest domain adaptation approach segmentation. Another U-Net [28] based approach [67] applies MMD at multiple upsample layers across the model to adapt the denser representations. DANNet [68] applies adversarial discriminator loss directly to the output predictions of target domain day-night images. [69] apply GAN-based image translation to aerial images to achieve significant improvements in segmentation performance. However, even with these more advanced approaches, the success of pure domain alignment for semantic segmentation has been limited.

Much greater success has been found with pseudo-labelling strategies, which are a common feature of many recent state-of-the art segmentation domain adaptation approaches. In their highly influential work, [12] make a number of contributions. First, the concept of pseudo-labeling is extended to semantic segmentation - the magnitude of softmax score is considered as the prediction confidence for each pixel. After each epoch, all target set images are added to the training set as pseudo-labels, though all but the $r\%$ most confident of all pixel predictions are "masked out" (made to not influence

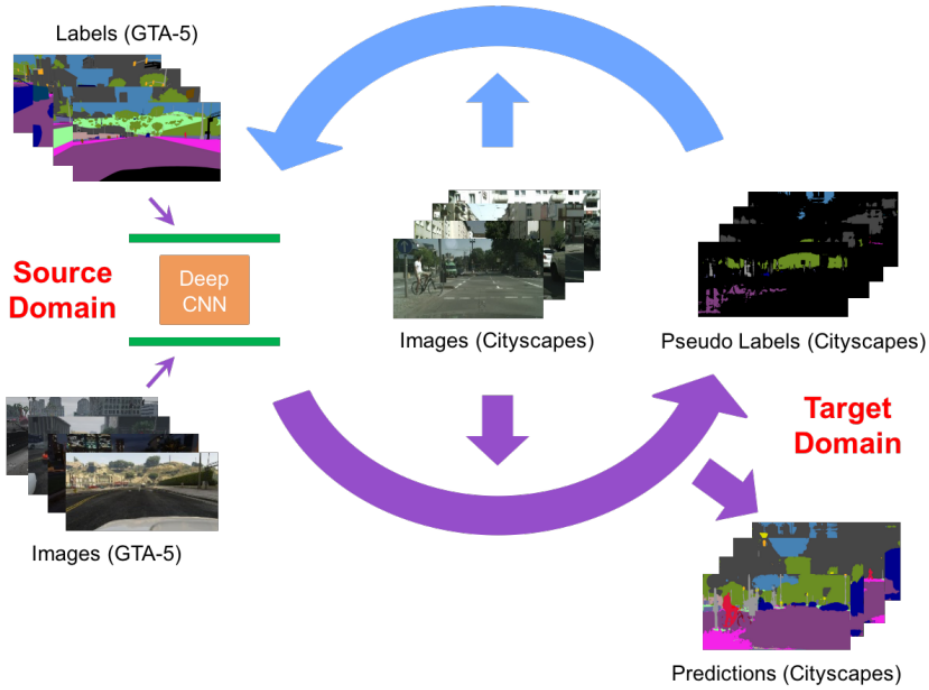

Figure 1.12: An overview of the class-balanced self-training (CBST) pipeline [12].

training) (figure 1.12). Intuiting that the model's target set performance will initially be poor, a scheduling system is introduced where r starts low and is gradually increased as the model improves. [60] use ensembles to better estimate model confidence, finding underconfident models provide much more information to the task. [12] also introduces class-balancing: given predictions in segmentation need to be made for multiple classes on each image, we can presume some classes will have a smaller domain gap between D_S and D_T. For instance, for a road-scene labelling task, roads between the two datasets may be highly similar while cars are highly different. Therefore, simply choosing the $r\%$ highest-confidence predictions for pseudo-labels would likely make the model choose only easy-to-adapt classes as pseudo-labels and ignore difficult classes, which would only get worse as training continues. Therefore, predictions are normalised per-class, with the top $r_c\%$ (for class c) predictions then chosen per-class. Finally, the concept of spatial priors is introduced - the average frequency of each class at a given location in the source dataset is used to modify the confidence of target dataset predictions for pseudo-labelling, encouraging pseudo-labels with a similar distribution (e.g. sky is more likely to appear at the top of the image, and should be encouraged).

CBST's have been widely adopted across fields of domain adaptation, though not

all agreement has been universal. ADVENT [70] actually found training on low-confidence regions to boost performance. Based on the observation that source set predictions are often overconfident (low-entropy) and target underconfident (high-entropy), it's approach focuses on minimising model entropy using an adversarial loss. [71] combines GAN image translation with pseudo-labelling through bidirectional learning. After initial training of a segmentor and a X_S to X_T GAN, pseudo label predictions are made, and the most confident labels are both (a) fed into the training set and (b) used to re-train the GAN, improving its ability to support crossing the domain gap. [72] also explores challenge of aligning different image subsets in segmentation, denoting the alignment difference between stuff (background image content like roads) and things (instance based content like cars). While stuff classes can often be aligned globally between datasets, instances of things may differ in distribution even within a single source image. Therefore, separate adversarial losses are applied for each - one for global stuff alignment and one for instance-based thing alignment. Ultimately, adversarial loss and pseudo-label loss have become the primary approaches for domain alignment for semantic segmentation and general computer vision.

1.2.3 Domain adaptation for vision transformers

The application of transformer networks to domain adaptation is a new and highly promising field. In many domains, transformers have also been widely adopted due to their strong transfer learning capabilities [40] [73]. The same applies for vision - as explored by ViT [8] and SETR [45], transformers are very good at learning overall distributions through pre-training and then adapting to new labelled data.

Many initial domain adaptation approaches applied transformers tangentially or to non-computer vision fields. [74] apply transformers for text generation in NLP domain adaptation. [75] find training models on multiple NLP tasks results in improved domain adaptation performance over typical pre-training. [76] apply a transformer after traditional CNN-based computer vision feature extractors to improve source-free domain adaptation by applying focus on relevant input object features.

TVT [15] explores vision transformer's general ability to perform domain adap-

Algorithm		A→ W	D→ W	W→ D	A→ D	D→ A	W→ A	Avg
Source Only	AlexNet	61.6	95.4	99.0	63.8	51.1	49.8	70.1
DDC	AlexNet	61.8	95.0	98.5	64.4	52.1	52.2	70.6
DAN	AlexNet	68.5	96.0	99.0	67.0	54.0	53.1	72.9
RevGrad	AlexNet	73.0	96.4	99.2	72.3	53.4	51.2	74.3
JAN	AlexNet	75.2	96.6	99.6	72.8	57.5	56.3	76.3
CDAN	AlexNet	78.3	97.2	100.0	76.3	57.3	57.3	77.7
PFAN	AlexNet	83.0	99.0	99.9	76.3	63.3	60.8	80.4
Source Only	ResNet	68.4	96.7	99.3	68.9	62.5	60.7	76.1
DDC	ResNet	75.6	96.0	98.2	76.5	62.2	61.5	78.3
DAN	ResNet	80.5	97.1	99.6	78.6	63.6	62.8	80.4
RevGrad	ResNet	82.0	96.9	99.1	79.7	68.2	67.4	82.2
JAN	ResNet	86.0	96.7	99.7	85.1	69.2	70.7	84.6
CDAN	ResNet	94.1	98.6	100.0	92.9	71.0	69.3	87.7
TADA	ResNet	94.3	98.7	99.8	91.6	72.9	73.0	88.4
TAT	ResNet	92.5	99.3	100.0	93.2	73.1	72.1	88.4
SHOT	ResNet	90.1	98.4	99.9	94.0	74.7	74.3	88.6
ALDA	ResNet	95.6	97.7	100.0	94.0	72.2	72.5	88.7
Source Only	ViT	89.18	98.87	100.0	88.76	80.09	79.77	89.45
Baseline	ViT	91.57	98.99	100.0	90.56	80.16	80.12	90.23
TVT	ViT	96.35	99.37	100.0	96.39	84.91	86.05	93.85

Table 1.1: Performance comparison of TVT against other domain adaptation approaches [15]. Note the "Source Only" TVT model (bottom row) consistently outperforms CNN-based domain adaptation approaches with access to the source data.

tation. Surprisingly, ViT's zero-shot transferability was found to outperform many state-of-the art techniques for classification domain adaptation. On the Office-31 [77] domain adaptation dataset, an ImageNet-pre-trained ViT achieves 81.45%, a 7% accuracy increase over SHOT [61]. Multiple domain adaptation modifications were evaluated for TVT. Applying an adversarial domain discriminator on class token output (equivalent to CNN features) provided some improvements. Best results were seen by applying a patch-wise discriminator that de-emphasises patch weights that are easily discriminated, combined with a loss that maximises feature clustering [63]. [78] similarly explore an original transformer architecture for object detection domain adaptation, applying adversarial feature alignment on each input patch.

Transformers, specifically cross-attention, have been found to be effective in tasks that require crossing even extremely large task or domain gaps. Cross-attention has successfully been used to learn inter-image feature extraction [79] and even image-to-text speech alignment [80]. CDTrans [13] applies a pseudo-labelling approach as in SHOT [61] to take advantage of this for domain adaptation. It uses 3 ViT-like backbone branches - source, target, and source-target - all with shared weights (figure 1.13). The source and

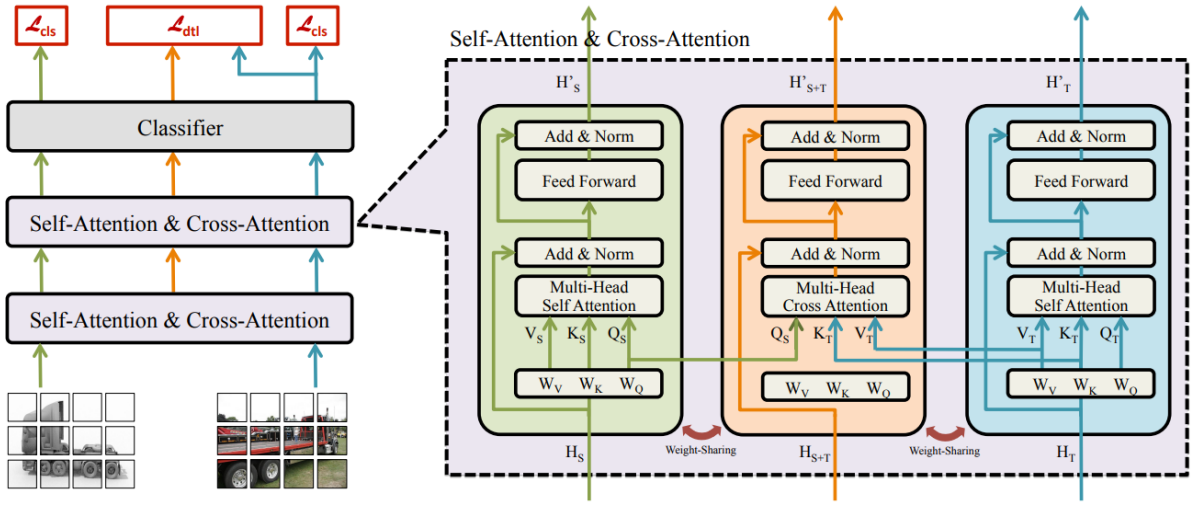

Figure 1.13: Architecture of CDTrans [13]. The source-target branch (centre) receives output from the source branch's Q and the target branch's K and V values.

target branches take in similar images from the source and target sets respectively, and compute self-attention on them. The source-target branch then computes cross-attention between the outputs at each block. The intuition behind this is that if false pseudo-labels are given (the main drawback of pseudo-labelling), cross attention will assign low weights to the dissimilar features produced, which will mitigate the backpropagation effects of these blocks. The output of cross-attention (which should possess the most domain-invariant features) is then used to make a prediction on the target, and the result's difference to the target branch output is added to the loss to encourage learning-cross domain features (and minimise the effect of false positives). Once domain-invariant feature extraction is learned throughout the network, only the target branch is used for inference.

The idea of using cross-attention is extended in BCAT (Bi-Directional Cross-Attention Transformer) [14] which introduces a target-source branch alongside the source-target branch that computes cross-attention on the source rather than the target data (figure 1.14). Instead of CDTrans' student-teacher approach, an MMD transfer loss is then between the merged feature outputs of the target/source-target branch and the source/target-source branch (figure 1.15). This approach maintains the benefit of cross-attention in terms of false-positive pseudo-labels while more directly encouraging the learning of domain-invariant features.

Despite the recent promising results achieved using segmentation transform-

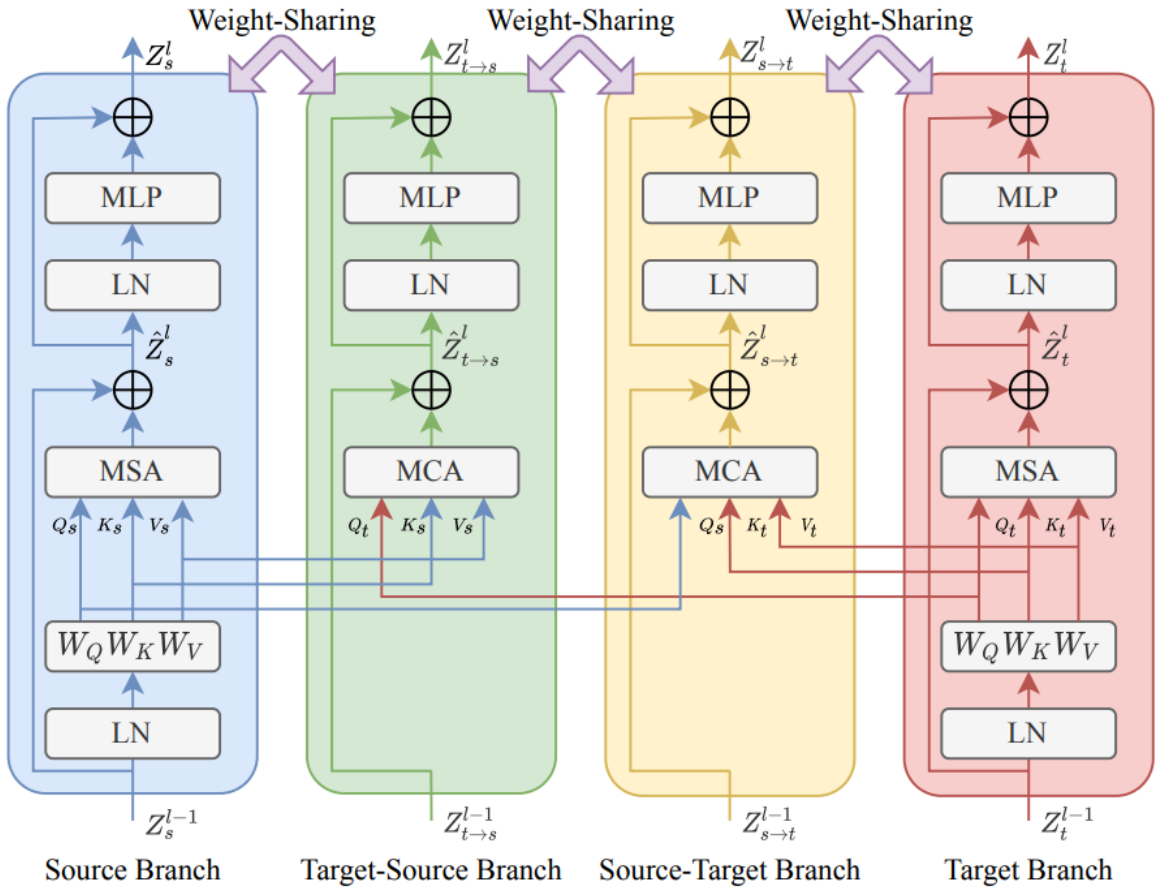

Figure 1.14: Structure of BCAT block [14].

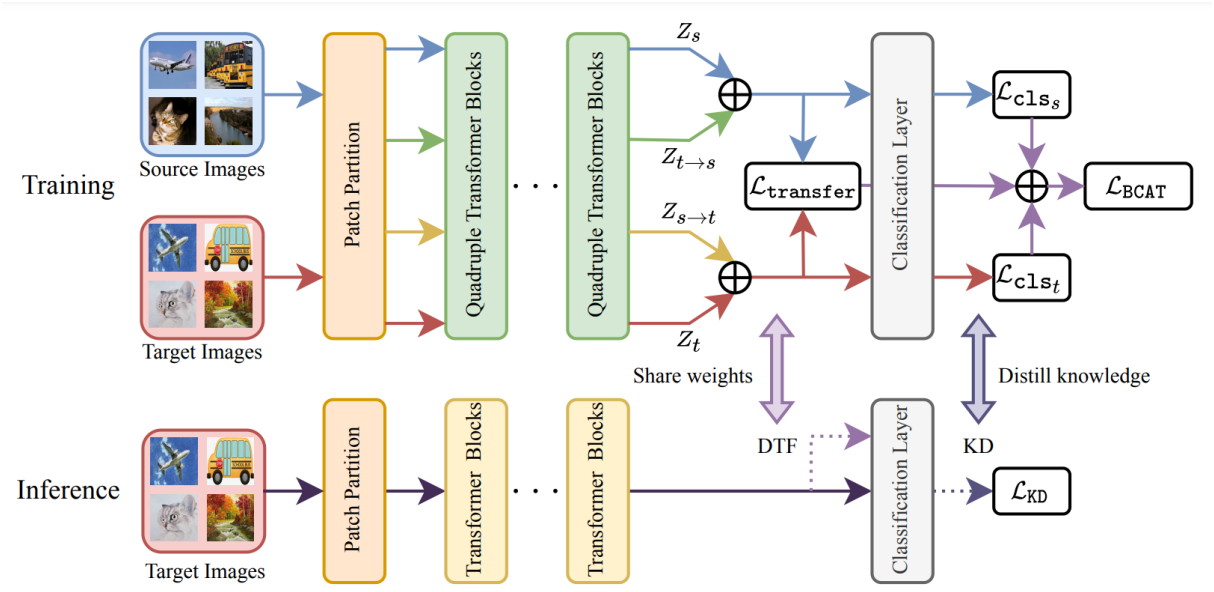

Figure 1.15: Overall BCAT architecture [14]. Transfer loss is computed between the fused source/target source and target/source-target features, then added to overall loss.

ers and the established strong domain adaptation abilities of transformers in computer vision, only one publication (at the time of writing) has explored transformer domain adaptation for semantic segmentation. DTNet [81] focuses on the medical segmentation. The architecture improves the efficiency of multi-head-self-attention by applying attention separately on a channel-wise, neighbour-wise, and dilated basis. This is combined with a patch-wise domain discriminator similar to [15] to prioritise transferable features. Therefore, there exists an opportunity to apply other transformer-specific domain adaptation approaches [15] [13] [14] to semantic segmentation with appropriate architectural modifications.

Bibliography

[1] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," arXiv, Tech. Rep. arXiv:1604.01685, Apr. 2016, arXiv:1604.01685 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1604.01685

[2] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*, 2019.

[3] H. Lin, Z. Shi, and Z. Zou, "Maritime Semantic Labeling of Optical Remote Sensing Images with Multi-Scale Fully Convolutional Network," *Remote Sensing*, vol. 9, p. 480, May 2017.

[4] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," arXiv, Tech. Rep. arXiv:1612.01105, Apr. 2017, arXiv:1612.01105 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1612.01105

[5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," arXiv, Tech. Rep. arXiv:1606.00915, May 2017, arXiv:1606.00915 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1606.00915

[6] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for Real-Time Semantic Segmentation on High-Resolution Images," arXiv, Tech. Rep. arXiv:1704.08545, Aug. 2018, arXiv:1704.08545 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1704.08545

[7] R. P. K. Poudel, S. Liwicki, and R. Cipolla, "Fast-SCNN: Fast Semantic Segmentation Network," arXiv, Tech. Rep. arXiv:1902.04502, Feb. 2019,

arXiv:1902.04502 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1902.04502

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," arXiv, Tech. Rep. arXiv:2010.11929, Jun. 2021, arXiv:2010.11929 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2010.11929

[9] S. Mehta and M. Rastegari, "MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer," arXiv, Tech. Rep. arXiv:2110.02178, Mar. 2022, arXiv:2110.02178 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2110.02178

[10] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for Data: Ground Truth from Computer Games," arXiv, Tech. Rep. arXiv:1608.02192, Aug. 2016, arXiv:1608.02192 [cs] version: 1 type: article. [Online]. Available: http://arxiv.org/abs/1608.02192

[11] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell, "CyCADA: Cycle-Consistent Adversarial Domain Adaptation," arXiv, Tech. Rep. arXiv:1711.03213, Dec. 2017, arXiv:1711.03213 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1711.03213

[12] Y. Zou, Z. Yu, B. V. K. V. Kumar, and J. Wang, "Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training," arXiv, Tech. Rep. arXiv:1810.07911, Oct. 2018, arXiv:1810.07911 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1810.07911

[13] T. Xu, W. Chen, P. Wang, F. Wang, H. Li, and R. Jin, "CDTrans: Cross-domain Transformer for Unsupervised Domain Adaptation," arXiv, Tech. Rep. arXiv:2109.06165, Sep. 2021, arXiv:2109.06165 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2109.06165

[14] X. Wang, P. Guo, and Y. Zhang, "Domain Adaptation via Bidirectional Cross-Attention Transformer," arXiv, Tech. Rep. arXiv:2201.05887, Jan. 2022,

arXiv:2201.05887 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2201.05887

[15] J. Yang, J. Liu, N. Xu, and J. Huang, "TVT: Transferable Vision Transformer for Unsupervised Domain Adaptation," arXiv, Tech. Rep. arXiv:2108.05988, Nov. 2021, arXiv:2108.05988 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2108.05988

[16] M. H. Hesamian, W. Jia, X. He, and P. Kennedy, "Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges," *Journal of Digital Imaging*, vol. 32, no. 4, pp. 582–596, Aug. 2019. [Online]. Available: https://doi.org/10.1007/s10278-019-00227-x

[17] S. R. Richter, H. A. AlHaija, and V. Koltun, "Enhancing Photorealism Enhancement," arXiv, Tech. Rep. arXiv:2105.04619, May 2021, arXiv:2105.04619 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2105.04619

[18] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," arXiv, Tech. Rep. arXiv:1407.5736, Jul. 2014, arXiv:1407.5736 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1407.5736

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," arXiv, Tech. Rep. arXiv:1311.2524, Oct. 2014, arXiv:1311.2524 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1311.2524

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," arXiv, Tech. Rep. arXiv:1411.4038, Mar. 2015, arXiv:1411.4038 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1411.4038

[21] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv, Tech. Rep. arXiv:1409.1556, Apr. 2015, arXiv:1409.1556 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1409.1556

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," arXiv, Tech. Rep. arXiv:1512.03385, Dec. 2015, arXiv:1512.03385 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1512.03385

[23] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," arXiv, Tech. Rep. arXiv:1610.02357, Apr. 2017, arXiv:1610.02357 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1610.02357

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," arXiv, Tech. Rep. arXiv:1409.0575, Jan. 2015, arXiv:1409.0575 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1409.0575

[25] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era," arXiv, Tech. Rep. arXiv:1707.02968, Aug. 2017, arXiv:1707.02968 [cs] version: 2 type: article. [Online]. Available: http://arxiv.org/abs/1707.02968

[26] W. Liu, A. Rabinovich, and A. C. Berg, "ParseNet: Looking Wider to See Better," arXiv, Tech. Rep. arXiv:1506.04579, Nov. 2015, arXiv:1506.04579 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1506.04579

[27] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," arXiv, Tech. Rep. arXiv:1412.7062, Jun. 2016, arXiv:1412.7062 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1412.7062

[28] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," arXiv, Tech. Rep. arXiv:1505.04597, May 2015, arXiv:1505.04597 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1505.04597

[29] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation," arXiv, Tech. Rep. arXiv:1611.06612, Nov. 2016, arXiv:1611.06612 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1611.06612

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," 2014, vol. 8691, pp. 346–361, arXiv:1406.4729 [cs]. [Online]. Available: http://arxiv.org/abs/1406.4729

[31] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," arXiv, Tech. Rep. arXiv:1706.05587, Dec. 2017, arXiv:1706.05587 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1706.05587

[32] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," arXiv, Tech. Rep. arXiv:1802.02611, Aug. 2018, arXiv:1802.02611 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1802.02611

[33] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation," arXiv, Tech. Rep. arXiv:1606.02147, Jun. 2016, arXiv:1606.02147 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1606.02147

[34] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation," arXiv, Tech. Rep. arXiv:1808.00897, Aug. 2018, arXiv:1808.00897 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1808.00897

[35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv, Tech. Rep. arXiv:1704.04861, Apr. 2017, arXiv:1704.04861 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1704.04861

[36] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," arXiv, Tech. Rep. arXiv:1801.04381, Mar. 2019, arXiv:1801.04381 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1801.04381

[37] R. P. K. Poudel, U. Bonde, S. Liwicki, and C. Zach, "ContextNet: Exploring Context and Detail for Semantic Segmentation in Real-time," arXiv, Tech. Rep.

arXiv:1805.04554, Nov. 2018, arXiv:1805.04554 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1805.04554

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *arXiv:1706.03762 [cs]*, Dec. 2017, arXiv: 1706.03762. [Online]. Available: http://arxiv.org/abs/1706.03762

[39] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv, Tech. Rep. arXiv:1810.04805, May 2019, arXiv:1810.04805 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1810.04805

[40] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," p. 24, 2019.

[41] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions," arXiv, Tech. Rep. arXiv:2102.12122, Aug. 2021, arXiv:2102.12122 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2102.12122

[42] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," arXiv, Tech. Rep. arXiv:2012.12877, Jan. 2021, arXiv:2012.12877 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2012.12877

[43] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," arXiv, Tech. Rep. arXiv:2103.14030, Aug. 2021, arXiv:2103.14030 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2103.14030

[44] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, and C. Shen, "Twins: Revisiting the Design of Spatial Attention in Vision Transformers," arXiv, Tech. Rep. arXiv:2104.13840, Sep. 2021, arXiv:2104.13840 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2104.13840

[45] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. S. Torr, and L. Zhang, "Rethinking Semantic Segmentation from a

Sequence-to-Sequence Perspective with Transformers," *arXiv:2012.15840 [cs]*, Jul. 2021, arXiv: 2012.15840. [Online]. Available: http://arxiv.org/abs/2012.15840

[46] 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:2105.15203 [cs]*, Oct. 2021, arXiv: 2105.15203. [Online]. Available: http://arxiv.org/abs/2105.15203

[47] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba, "Semantic Understanding of Scenes through the ADE20K Dataset," arXiv, Tech. Rep. arXiv:1608.05442, Oct. 2018, arXiv:1608.05442 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1608.05442

[48] H. Bai, H. Mao, and D. Nair, "Dynamically pruning segformer for efficient semantic segmentation," arXiv, Tech. Rep. arXiv:2111.09499, Nov. 2021, arXiv:2111.09499 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2111.09499

[49] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning Efficient Object Detection Models with Knowledge Distillation," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/e1e32e235eee1f970470a3a6658dfdd5-Abstract.html

[50] W. Zhang, Z. Huang, G. Luo, T. Chen, X. Wang, W. Liu, G. Yu, and C. Shen, "TopFormer: Token Pyramid Transformer for Mobile Semantic Segmentation," arXiv, Tech. Rep. arXiv:2204.05525, Apr. 2022, arXiv:2204.05525 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2204.05525

[51] G. Wilson and D. J. Cook, "A Survey of Unsupervised Deep Domain Adaptation," arXiv, Tech. Rep. arXiv:1812.02849, Feb. 2020, arXiv:1812.02849 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/1812.02849

[52] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364913491297

[53] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola, "A Kernel Method for the Two-Sample-Problem," in *Advances in Neural Information Processing Systems*, vol. 19. MIT Press, 2006. [Online]. Available: https://proceedings.neurips.cc/paper/2006/hash/e9fb2eda3d9c55a0d89c98d6c54b5b3e-Abstract.html

[54] B. Sun, J. Feng, and K. Saenko, "Return of Frustratingly Easy Domain Adaptation," arXiv, Tech. Rep. arXiv:1511.05547, Dec. 2015, arXiv:1511.05547 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1511.05547

[55] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond Sharing Weights for Deep Domain Adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 4, pp. 801–814, Apr. 2019, arXiv:1603.06432 [cs]. [Online]. Available: http://arxiv.org/abs/1603.06432

[56] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," arXiv, Tech. Rep. arXiv:1406.2661, Jun. 2014, arXiv:1406.2661 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/1406.2661

[57] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive Growing of GANs for Improved Quality, Stability, and Variation," arXiv, Tech. Rep. arXiv:1710.10196, Feb. 2018, arXiv:1710.10196 [cs, stat] version: 3 type: article. [Online]. Available: http://arxiv.org/abs/1710.10196

[58] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-Adversarial Training of Neural Networks," arXiv, Tech. Rep. arXiv:1505.07818, May 2016, arXiv:1505.07818 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/1505.07818

[59] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," arXiv, Tech. Rep. arXiv:1411.1784, Nov. 2014, arXiv:1411.1784 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/1411.1784

[60] K. Kamnitsas, S. Winzeck, E. N. Kornaropoulos, D. Whitehouse, C. Englman, P. Phyu, N. Pao, D. K. Menon, D. Rueckert, T. Das, V. F. J. Newcombe, and B. Glocker, "Transductive image segmentation: Self-training and effect of uncertainty

estimation," arXiv, Tech. Rep. arXiv:2107.08964, Aug. 2021, arXiv:2107.08964 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2107.08964

[61] J. Liang, D. Hu, and J. Feng, "Do We Really Need to Access the Source Data? Source Hypothesis Transfer for Unsupervised Domain Adaptation," arXiv, Tech. Rep. arXiv:2002.08546, Jun. 2021, arXiv:2002.08546 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2002.08546

[62] H. Larochelle, D. Erhan, and Y. Bengio, "Zero-data Learning of New Tasks," p. 6, 2008.

[63] O. Chapelle and A. Zien, "Semi-Supervised Classification by Low Density Separation," p. 8, 2005.

[64] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 3234–3243. [Online]. Available: http://ieeexplore.ieee.org/document/7780721/

[65] G. Csurka, R. Volpi, and B. Chidlovskii, "Unsupervised Domain Adaptation for Semantic Image Segmentation: a Comprehensive Survey," arXiv, Tech. Rep. arXiv:2112.03241, Dec. 2021, arXiv:2112.03241 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2112.03241

[66] J. Hoffman, D. Wang, F. Yu, and T. Darrell, "FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation," arXiv, Tech. Rep. arXiv:1612.02649, Dec. 2016, arXiv:1612.02649 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1612.02649

[67] R. Bermúdez-Chacón, P. Márquez-Neila, M. Salzmann, and P. Fua, "A domain-adaptive two-stream U-Net for electron microscopy image segmentation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Apr. 2018, pp. 400–404, iSSN: 1945-8452.

[68] X. Wu, Z. Wu, H. Guo, L. Ju, and S. Wang, "DANNet: A One-Stage Domain Adaptation Network for Unsupervised Nighttime Semantic Segmentation," arXiv,

Tech. Rep. arXiv:2104.10834, Apr. 2021, arXiv:2104.10834 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2104.10834

[69] B. Benjdira, Y. Bazi, A. Koubaa, and K. Ouni, "Unsupervised Domain Adaptation using Generative Adversarial Networks for Semantic Segmentation of Aerial Images," *Remote Sensing*, vol. 11, no. 11, p. 1369, Jun. 2019, arXiv:1905.03198 [cs]. [Online]. Available: http://arxiv.org/abs/1905.03198

[70] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez, "ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation," arXiv, Tech. Rep. arXiv:1811.12833, Apr. 2019, arXiv:1811.12833 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1811.12833

[71] Y. Li, L. Yuan, and N. Vasconcelos, "Bidirectional Learning for Domain Adaptation of Semantic Segmentation," arXiv, Tech. Rep. arXiv:1904.10620, Apr. 2019, arXiv:1904.10620 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1904.10620

[72] Z. Wang, M. Yu, Y. Wei, R. Feris, J. Xiong, W.-m. Hwu, T. S. Huang, and H. Shi, "Differential Treatment for Stuff and Things: A Simple Unsupervised Domain Adaptation Method for Semantic Segmentation," arXiv, Tech. Rep. arXiv:2003.08040, Jun. 2020, arXiv:2003.08040 [cs, eess] type: article. [Online]. Available: http://arxiv.org/abs/2003.08040

[73] D. Wright and I. Augenstein, "Transformer Based Multi-Source Domain Adaptation," arXiv, Tech. Rep. arXiv:2009.07806, Sep. 2020, arXiv:2009.07806 [cs, stat] type: article. [Online]. Available: http://arxiv.org/abs/2009.07806

[74] S. Shakeri, C. N. d. Santos, H. Zhu, P. Ng, F. Nan, Z. Wang, R. Nallapati, and B. Xiang, "End-to-End Synthetic Data Generation for Domain Adaptation of Question Answering Systems," arXiv, Tech. Rep. arXiv:2010.06028, Oct. 2020, arXiv:2010.06028 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2010.06028

[75] X. Liu, P. He, W. Chen, and J. Gao, "Multi-Task Deep Neural Networks for Natural Language Understanding," arXiv, Tech. Rep. arXiv:1901.11504,

May 2019, arXiv:1901.11504 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1901.11504

[76] G. Yang, H. Tang, Z. Zhong, M. Ding, L. Shao, N. Sebe, and E. Ricci, "Transformer-Based Source-Free Domain Adaptation," arXiv, Tech. Rep. arXiv:2105.14138, May 2021, arXiv:2105.14138 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2105.14138

[77] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting Visual Category Models to New Domains," in *Computer Vision – ECCV 2010*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer, 2010, pp. 213–226.

[78] W. Wang, Y. Cao, J. Zhang, F. He, Z.-J. Zha, Y. Wen, and D. Tao, "Exploring Sequence Feature Alignment for Domain Adaptive Detection Transformers," in *Proceedings of the 29th ACM International Conference on Multimedia*, Oct. 2021, pp. 1730–1738, arXiv:2107.12636 [cs]. [Online]. Available: http://arxiv.org/abs/2107.12636

[79] X. Li, Y. Hou, P. Wang, Z. Gao, M. Xu, and W. Li, "Trear: Transformer-based RGB-D Egocentric Action Recognition," arXiv, Tech. Rep. arXiv:2101.03904, Jan. 2021, arXiv:2101.03904 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2101.03904

[80] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal Transformer for Unaligned Multimodal Language Sequences," arXiv, Tech. Rep. arXiv:1906.00295, Jun. 2019, arXiv:1906.00295 [cs] type: article. [Online]. Available: http://arxiv.org/abs/1906.00295

[81] Y. Li, J. Li, R. Dan, S. Wang, K. Jin, G. Zeng, J. Wang, X. Pan, Q. Zhang, H. Zhou, Q. Jin, L. Wang, and Y. Wang, "Dispensed Transformer Network for Unsupervised Domain Adaptation," arXiv, Tech. Rep. arXiv:2110.14944, Oct. 2021, arXiv:2110.14944 [cs] type: article. [Online]. Available: http://arxiv.org/abs/2110.14944