MTL-TransMODS: Cascaded Multi-Task Learning for Moving Object Detection and Segmentation with Unified Transformers

Eslam Mohamed BAKR
Deep Learning Researcher
Valeo R&D Cairo, EGYPT
eslam.mohamed-abdelrahman@valeo.com

Ahmad El Sallab
AI Senior Expert
Valeo R&D Cairo, EGYPT
ahmad.el-sallab@valeo.com

Abstract

Recently, transformer-based networks have achieved state-of-the-art performance in computer vision tasks. In this paper, we propose a new cascaded MTL transformer-based framework, termed MTL-TransMODS, that tackles the moving object detection and segmentation tasks due to its importance for Autonomous Driving tasks. A critical problem in this task is how to model the spatial correlation in each frame and the temporal relationship across multiple frames to capture the motion cues. MTL-TransMODS, introducing a vision transformer to employ the temporal and spatial associations, and tackle both tasks using only one fully shared transformer architecture with unified queries. Extensive experiments demonstrate the superiority of our MTL-TransMODS over state-of-the-art methods on the KittiMoSeg dataset (56). Results show 0.3% mAP improvement for Moving Object Detection, and 5.7% IoU improvement for Moving Object Segmentation, over the state-of-the-art techniques. Qualitative results can be found on the following link.

1 Introduction

Moving object detection (MOD) or segmentation (MOS), as a fundamental task in the computer vision community, attracts more and more attention in recent years due to its potential application in video surveillance, activity recognition, autonomous driving, etc. In this work, we learn both tasks jointly, moving object detection and segmentation (MODS), to exploit the constructive relation among them. MODS is a challenging task due to illumination, dynamic background changes, and the relative motion (4) (21) (24) (50) (51) (54) of surrounding moving objects w.r.t the visual sensors that are planted on a moving robot or vehicle. Besides, MODS could be formulated as two dimensional problem; spatial dimension and temporal dimension, where modeling the spatial and temporal relationship jointly is challenging.

For many years, ConvNets have been the architecture of choice in computer vision in general, and for performing object detection and segmentation tasks in particular. Recently, transformers have shown a great success compared to ConvNets in capturing the spatial relation in various applications, in classification (15) (76) (10) (19) (6) (31), in detection, (5) (72), in segmentation (79) (42) (74) (75) and in tracking (65) (43). Witnessing the great success of transformers in handling the sequential data (68), which is firstly introduced in the natural language processing (NLP) domain where the sequential input is the sequence of words, therefore, we proposed a new transformer-based architecture for MODS, that jointly exploits the temporal relationships and the spatial relationships.

Multi-task learning (MTL) (12) (78) (13) approaches offer advantages like improved data efficiency, reduced overfitting through shared representations, and fast learning by leveraging auxiliary information. Recent related fields, i.e., meta-learning (16) (70), transfer learning (81) (66) (37) (26), and

Machine Learning for Autonomous Driving Workshop at the 35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

continuous learning (40) (52) (53) are mimicking the learning process for babies by integrating prior knowledge across different tasks. For example, babies in their entry-level are struggling to learn basic tasks from scratch like walking and talking (41) (3) (38), but once they learn the simple tasks they build upon them while learning more complex ones, such as playing football. By analogy, neural networks require such numerous training datasets (48) (60) and computation time for each task to be learned from scratch, neglecting the useful learned representation from other related tasks. However, learning multi-task is suffering from the negative transfer phenomenon (34) (77) (17) (7) (73), where one or more tasks dominate other tasks by destroying the learned representations.

Therefore, in this paper, we tackle two challenges. i.e., 1) How to effectively learn joint representations for object detection and segmentation using Unified Transformer (UT) architecture? 2) How to adequately utilize the temporal and spatial relationships since it is crucial for moving object recognition?

A novel entirely shared transformer encoder-decoder architecture with Unified Queries (UQ) is proposed to tackle the first challenge. To tackle the second challenge, the encoder module is adopted from the one-time step MODETR (44), and extended to be more generic by exploiting previous time-stamps while encoding motion cues, moreover, spatio-temporal transformer architecture is proposed equipped by a novel Temporal Positional Encoding (TPE) module to adequately utilize the temporal and spatial correlations. Accordingly, our main contributions can be summarized as follows:

- We proposed a new end-to-end MTL transformer-based framework for moving object detection and segmentation, termed MTL-TransMODS, that effectively learn a joint representations for both tasks while exploiting the spatial and temporal correlations.
- We propose a novel way for sharing the learnable representations based on transformer architecture across different tasks via learning robust Unified Queries (UQ).
- Through detailed analysis along with ablation studies, we examine the internal behavior and validity of our method.
- We verify the effectiveness of MTL-TransMODS on KittiMoSeg dataset (56), and achieves new state-of-the-art accuracy.

The rest of the paper is outlined as follows, first, we discuss the related work, followed by the details of the proposed model. Then we present detailed ablation studies to settle on the best architectural design, and finally, illustrate the experimental setup for the various experiments we conducted for every contribution.

2 Related work

Our novel architecture draws success from several areas, including Vision transformers, Spatio-Temporal correlation, and Multi-Task-Learning.

Transformers for classification. ViT (15) exploits the self-attention operation which is the core of the transformers that produce global awareness of the given input data that overcome the local interactions of convolutions. Therefore, ViT (15) replaces the CNN backbone and propose a fully transformer-based architecture. T2T-ViT (76) using an adaptive tokenization technique, a layer-wise Tokens-to-Token (T2T) transformation block, instead of the simple tokenization used in (15). CrossViT (6) exploits the multi-scale features by combining image patches of different sizes. Transformer-iN-Transformer(TNT) (19) utilizes both patch-level and pixel-level representation, to fully utilize the intrinsic structure information inside each patch. CPE (10) dynamically generates learnable positional encodings conditioned on the local neighborhood of the input tokens. LocalViT (31) adds locality to vision transformers by introducing depth-wise convolution into the feed-forward network. Swin (35) proposes a hierarchical transformer and shifted windows to simplify the transformer complexity by applying the self-attention on the non-overlapped windows. Twins (9) propose a spatially separable self-attention (SSSA) module, which is composed of two types of attention operations, i.e., local-grouped self-attention (LSA), and global sub-sampled attention (GSA).

Transformers for detection. DETR (5) and Deformable-DETR (80) treat the input image as a sequence of spatial features that enables the extension of the traditional transformer, previously used in NLP (68), in computer vision problems, by outputting the final set of predictions directly from the global image context. PVT (72) uses a progressive shrinking pyramid to reduce computations

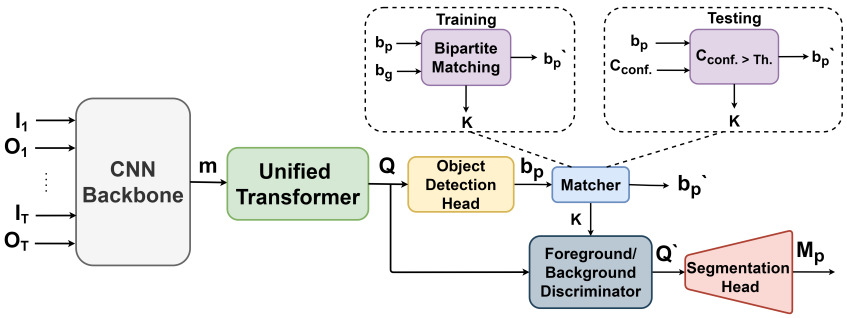

Figure 1: Overview of our proposed Spatio-Temporal Cascaded-MTL Transformer-based Model Architecture for Joint MOD and MOS. Unified transformer produces unified queries that represent both tasks simultaneously. Foreground/Background Discriminator generates class-queries from the produced unified queries.

of large feature maps to achieve a high output resolution on different tasks, e.g., object detection, semantic, and instance segmentation.

Transformers for Segmentation. TransVOS (42) extends DETR from a 2d attention transformer to a 3d attention to exploit both spatial and temporal relationships, which suffers from extensively computations. SETR (79) adopts ViT as a backbone while the decoder is based on the progressive up-sampling segmentation head like FCN (36). SegFormer (75) adapt DETR encoder to output multi-scale features and propose lightweight decoder using only stacked MLP blocks. Trans2Seg (74) adopts DETR's architecture (5) by using the same transformer encoder-decoder that relies on self-attention mechanism, and exploits the produced attention map from self-attention operation to produce segmentation masks. Segmenter (64) build on ViT (15) by obtaining a label for each patch.

Spatio-Temporal methods. ConvNets and LSTM mixed architectures, like ConvLSTM (59) have been used to handle both the spatial and temporal nature of videos, like in Moving Object Detection (MOD) (61; 62), and Instance Moving Object Segmentation (46) tasks. Recently, the Space-Time Memory (STM) based approaches (27) (30) (32) (39) (49) (58) capture the temporal information and neglecting the spatial relationships of pixels inside each frame. In contrast, siamese-based approaches (28) (18) (8) (71) (29) (2) compute spatial attention across the template patch against search region. The temporal relationships are not fully utilized since they ignore the dependency among all the previous frames. STEm-Seg (1) models the video data directly as 3D spatio-temporal volume which increases the model complexity. Recently, fully attention transformers (68) are replacing RNN, LSTM, and GRU in NLP. This motivates us to jointly capture the spatial and temporal relationships as they are crucial for MODS using an efficient, and powerful transformer architecture.

MTL-Transformer-based. UniT (22) propose Unified-multi-Modal Transformer model to learn seven tasks from different domains. UniT (22) uses two decoders, one for vision tasks and another one for the NLP tasks where each task learns its queries; task-specific query embedding. In contrast, our proposed architecture learns two vision tasks, i.e., detection and segmentation, using a joint encoder-decoder transformer and unified queries. Thus, we have the edge by proposing a fully unified transformer architecture across different tasks and showing its effectiveness.

3 Spatio-Temporal Cascaded-MTL Transformer

In this section we will quickly summarize the vanilla detection transformer (5) (Sec. 3.1), then illustrate our novel cascaded-MTL approach that learns object detection and segmentation jointly. Our approach is divided into three main parts; 1) Novel formulation for the segmentation task w.r.t transformer architectures (Sec. 3.2). 2) Cascaded-MTL-Transformer model (Sec. 3.3). 3) Spatio-Temporal transformer architecture (Sec. 3.4).

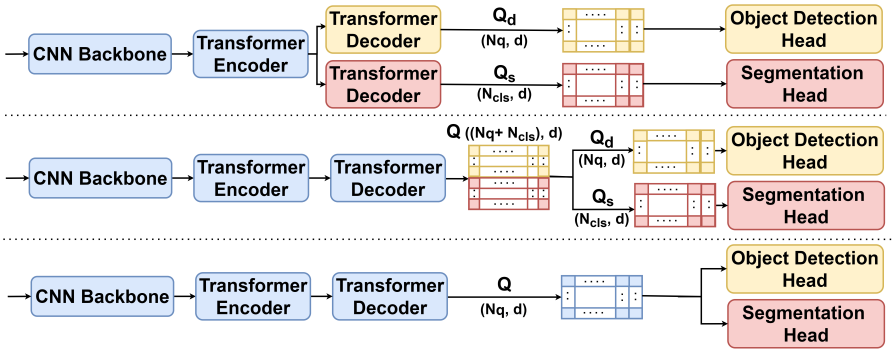

Figure 2: Comparison between different MTL approaches based on transformer architecture, where the blue modules indicates the shared representations between both tasks.

3.1 Vanilla detection transformer

The spatial one-step encoder-decoder architecture (5), termed as DETR but we will refer to it as vanilla detection transformer, mainly consists of three main steps, i.e., feature extraction, spatial transformer, and query transformer.

Features extractor consists of an arbitrary CNN backbone, followed by a 1×1 convolution layer to transform the channels dimension from C into the hidden dimension d, where d<C, that transforms the input image $I \in \mathbb{R}^{H_1 \times W_1 \times C}$ into feature map $m' \in \mathbb{R}^{H \times W \times d}$. Then the feature map m' is flattened across the spatial dimensions to be $m \in \mathbb{R}^{HW \times d}$. Transformer Encoder (TE) treats the spatial features m as a sequence of HW spatial features, each of dimension d and producing $E \in \mathbb{R}^{HW \times d}$. Transformer Decoder (TD) maps the spatial features E into object features based on learn-able object queries $Q \in \mathbb{R}^{N_q \times d}$, where N_q are the number of object queries. Finally, the object queries Q is feed into object detection head to produce the final boxes.

3.2 Segmentation transformer

Unlike the existing segmentation transformer-based architectures (42) (79) (75) (64), which only exploit the transformer encoder to provide a global receptive field in contrast to CNN's local receptive field, then use CNN decoder to produce segmentation masks, we adopt DETR's architecture (5) by formulating the semantic segmentation task as a problem of dictionary look-up table, and adapt DETR decoder by representing each category using a set of learnable queries while DETR's queries are representing each object. Therefore, we term DETR's queries detection-queries $Q_d \in \mathbb{R}^{N_q \times d}$, where N_q are the number of object queries that represent the maximum number of objects in the scene, which is varying from one dataset to another. And term the segmentation-queries $Q_s \in \mathbb{R}^{N_{cls} \times d}$, where N_{cls} are the number of class queries that represent the number of different categories in the dataset. By following this formulation for the segmentation task, we are able to follow the vanilla detection transformer pipeline (Sec. 3.1) to produce class-aware queries driven from the produced object queries. Several ways of generating the class-aware queries Q_s are discussed in details in the following section (Sec. 3.3) and demonstrated in Fig. 2.

3.3 Cascaded-MTL-Transformer model

Learning segmentation queries that represent object categories can be achieved by using a dedicated TD, as shown in the upper part at Fig. 2, where only the CNN encoder and TE are shared across the two tasks. The first TD could be termed as detection decoder as it produces object queries Q_d, while the second TD could be termed as segmentation decoder as it generates class-aware queries Q_s, that encodes the class category.

Another approach is to use one shared TD to produce the two tasks queries $Q \in \mathbb{R}^{(N_q+N_{cls}) \times d}$ in a one-shot, which later are been split into two sets; detection queries Q_d and segmentation queries Q_s, as shown in the middle part at Fig. 2. Inspired by these two approaches, an important question arises, can we learn both tasks more efficiently by sharing further modules?

To answer this question, we propose our novel MTL approach based on the Unified Queries (UQ) by decomposing the segmentation task into two related sub-tasks,i.e. 1) Discriminating objects. 2) Mask generation. After discriminating the objects in the scene, we generate the segmentation masks by predicting the category-wise label for each pixel guided by the discriminated objects. Formulating the segmentation task in this way allows us to fully utilize the learned representations in the detection task while learning the semantic segmentation task by cascading the segmentation task over the detection task.

Interpreting the cascaded analogy to the transformer based neural network, discriminating objects can be regarded as detection queries Q_d, or generally object queries $Q \in \mathbb{R}^{N_q \times d}$ as shown in the lower part in Fig.2 and in Fig.1. By aggregating the learnable object queries Q to the detection output head, the bounding boxes b_p are generated, where $b_p \in \mathbb{R}^{N_q \times 5}$, where each box is represented by the object center c_x, c_y, width W, height H and class type cls.

A matcher is used to refine the generated boxes b_p and produce refined boxes ${b_p}'$, as shown in the upper section of Fig. 1. At the training phase, the bipartite matching (5) (63) is applied on b_p producing refined bounding boxes ${b_p}' \in \mathbb{R}^{N_q^{'} \times 5}$, where $N_q^{'}$ is the number of the object queries that best match the ground truth boxes, and their indices K. At the inference phase, the refined bounding boxes ${b_p}'$, and their indices K are produced based on a thresholding process using the produced class confidence $C_{conf.}$.

To exploit the learned representations from the detection task, a foreground/background discriminator is proposed to discriminate the learned object queries Q to N_{Cls} class-based queries Q^C based on Eq. 1, and background object queries Q^b based on Eq. 2, where N_{Cls} is the number of classes in dataset. As shown in Eq. 1, the class-based queries are produced by conditional summation over the learned object queries Q, based on the desired class C and the selected indices K.

$$Q^C = \sum_{i=1}^{N_q} 1_{\{C=Cls[C]\}} 1_{\{i=K[i]\}} Q[i], \quad \text{for} \quad C = 1, 2, ..., N_{Cls}, \quad : Q^C \in \mathbb{R}^{1 \times d} \tag{1}$$

$$Q^b = \sum_{i=1}^{N_q} 1_{\{i \neq K[i]\}} Q[i], \quad : Q^b \in \mathbb{R}^{1 \times d} \tag{2}$$

The foreground object queries $Q^f \in \mathbb{R}^{N_{Cls} \times d}$ is produced by concatenating the class-based queries $Q^f = [Q^0, Q^1, .., Q^{N_{Cls}}]$. The foreground/background discriminator maps the learned queries from objects space to classes space. To generate the final segmentation masks, we concatenate the foreground and background object queries producing $Q^{\text{`}} \in \mathbb{R}^{(N_{Cls}+1) \times d}$. Finally, as shown in Fig. 1, a simple segmentation head is used to transform the transformer-decoder output, Q_s, into final mask $M_p \in \mathbb{R}^{N_{cls} \times H_1 \times W_1}$, that consists of stacked up sampling and reshaping layers. This requires to set $d = H \times W$, so that we can perform the reverse mapping that we did in the CNN backbone.

3.4 Spatio-Temporal transformer architecture

Our novel MTL approach could be used for the standard object detection and segmentation, however, we have explored a more challenging task, i.e., moving object detection and segmentation. Therefore, we adapt the vanilla 1-step MODETR (44) to exploit the spatial and temporal correlations across T timestamps instead of using only 1-step, as shown in Fig. 3. Accordingly, We encode the motion information through two ways, i.e., 1) Through the Optical Flow (OF) using FlowNet 2.0 (23), where the fusion between appearance (RGB) and motion (OF) is performed on the feature level. 2) By using multiple streams of T time stamps. As shown in Fig. 3, in contrast to MODETR (44), which uses two TEs, one for the image branch and another one for OF branch, we only use one unified transformer instead of using T transformer encoders (T-TE) for each timestamp.

As only one unified transformer is used, two variants of temporal features aggregation are studied, as shown in Fig. 4, where we can early aggregate the spatial features over the temporal dimension in the TE, or defer the temporal aggregation to the TD to be done late over the object queries.

Early temporal aggregation. In this approach, the list of T spatial features L are aggregated and flattened into $m \in \mathbb{R}^{HW \times Td}$, and fed to the TE that performs multi-head self-attention over this

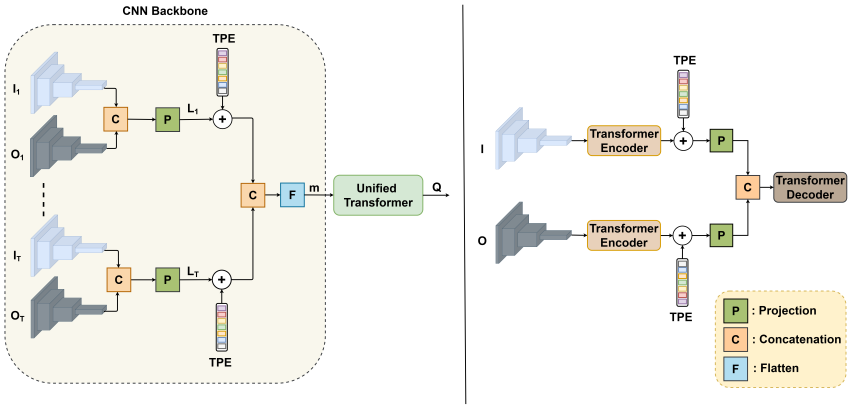

Figure 3: Comparison between our Spatio-Temporal model architecture, on the left, and MODETR (44), on the right.

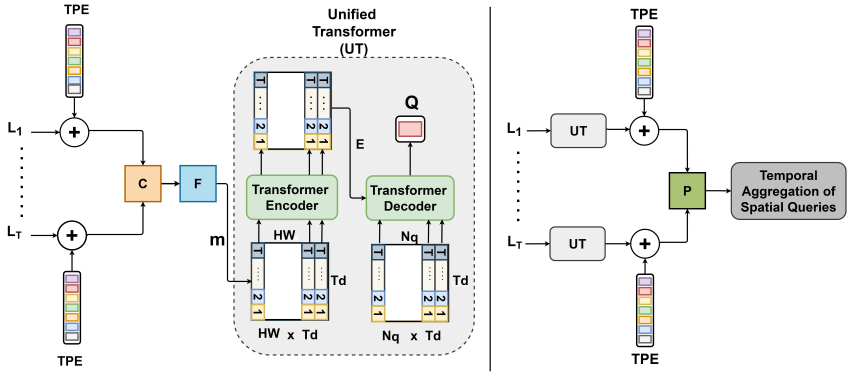

Figure 4: Comparing the two variants of the Spatio-Temporal Transformer model architecture. The ealry variant on the left, and the late variant on the right.

spatio-temporal map of object features traces. The TD will perform multi-head query-to-spatio-temporal features traces attention producing $Q \in \mathbb{R}^{N_q \times Td}$.

Late temporal aggregation. In contrast to the early aggregation, The TE is formed on T Spatial Transformer Encoders, as shown in the right part of Fig. 4, each resulting also in a list of T spatial features $E \in \mathbb{R}^{HW \times d}$. Finally, the TD is formed of two levels of decoders: 1) Spatial Query Decoders which are a list of T decoders, each resulting in a list of T query features $Q \in \mathbb{R}^{N_q \times d}$, which are then reshaped into an aggregated tensor over the temporal dimension to be $Q \in \mathbb{R}^{T \times N_q d}$. 2) Temporal Query Decoder, which transforms the Spatio-temporal queries traces Q into the final query features, by first flatten it such that $Q \in \mathbb{R}^{T N_q \times d}$, then using multi-head self-attention.

Temporal positional encoding. Transformers are a replacement to recurrent models, due to their fast parallel encoding nature (68). However, this comes at the cost of losing the sequential information of the input. To overcome that, positional encoding embedding was proposed in (68). DETR (5) incorporates the spatial positional encoding module to encode the spatial positional information despite both spatial and temporal positional encoding is crucial to encode the pixels of the moving object across time. Accordingly, we have introduced a temporal positional encoding (TPE) module, which is added just before the temporal aggregation takes place, being it early across the spatial features traces, as shown in the left part of Fig. 4 or late across the object queries traces, as shown in the right part of Fig. 4.

Motion cues	mAP_{Total}	AP_{50}	AP_{75}
RGB-only	23.1	42.2	23.7
RGB + RGB	25.3	47.2	24.5
RGB + OF	**33.9**	**59.3**	**37.2**

Table 1: Detailed comparisons on the effect of the motion features.

Temporal Aggregation	mAP_{Total}	AP_{50}	AP_{75}
N/A	33.9	59.3	37.2
Early	**36.1**	**62.3**	**43.4**
Late	34.0	61.1	36.1

Table 2: Studying the effect of the temporal aggregation.

4 Experiments

In this section, we first describe the used datasets. After that, we specify the experimental setup, including all hyper-parameters, and hardware specifications. Then, we perform controlled ablation experiments to settle on the best design for our proposed module and assess its sub-modules. Finally, we evaluate the performance of the proposed Cascaded-MTL-TransMODS, on the KittiMoSeg dataset (56).

4.1 Dataset

There is a huge limitation in publicly available datasets regarding moving object detection. (62) provides 1300 images only with weak annotation for MOD task. (69) provides 255 annotated frames only on KITTI dataset, and 3475 annotated frames on Cityscapes dataset (11). Thus, We use the extended version (56) of the publicly available KittiMoSeg dataset (62). (56) dataset consists of 12919 frames. The image resolution is 1242×375, and the labels determine whether the object is moving or static, includes the object bounding box and the motion mask.

4.2 Implementation details

We initialize our backbone networks with the weights pre-trained on ImageNet (14), then train the whole network for 30 epochs on COCO dataset (33) while freezing the backbone during the first ten epochs. In all our experiments, ResNet-50 (20) was used as a backbone. Our network is trained with Adam optimizer (25) with a scheduled learning rate that is decreased from $1e^{-3}$ to $1e^{-5}$, the whole network is end-to-end trained with learning rate exponentially decayed. We train a total of 200 epochs, using a warm-up learning rate of $1e^{-3}$ to $5e^{-3}$ in the first five epochs, and a learning rate exponentially decayed from $1e^{-3}$ to $1e^{-5}$ in the rest of epochs. 512×512 resolution images have been used across all the experiments, and T, which represents the number of temporal frames that we are using, is set to two. Our approach is implemented in Python using the PyTorch framework on two PCs with Intel Xeon(R) 4108 1.8GHz CPU, 64G RAM, Nvidia Titan-XP.

4.3 Ablation studies and analysis

4.3.1 Motion cues

Previous works on MOD (61; 62) indicate that input features can have a strong impact on the results. In particular, features holding motion cues can be of high impact. Thus, we evaluate the best input features at each time step, where we compare RGB, RGB+RGB, and RGB+OF options. In this setup, we use the vanilla 1-step DETR architecture. The results are in favor of the RGB+OF setup as shown in Table 1.

4.3.2 Early Vs. Late temporal aggregation:

In this setup, we evaluate the two architectural alternatives in Figure 4. For the sake of comparison, we fix the time window $T = 2$, the number of queries $N_q = 100$ and the transformer hidden dimension $d = 256$. Results are shown in Table 2. Both results of early and late architectures improve over the vanilla one-step DETR that doesn't contain temporal aggregation. However, the early architecture provides a significant improvement of 5% mAP.

T	mAP_{Total}	AP_{50}	AP_{75}
1-Step	33.9	59.3	37.2
2-Steps	**36.1**	62.3	**43.4**
4-Steps	**36.1**	**62.5**	43.1

Table 3: Quantitative results showing the effect of the temporal window size T.

TPE	mAP_{Total}	AP_{50}	AP_{75}
W/O	36.1	62.3	43.4
With TPE	**38.7**	**63.1**	**44.6**

Table 4: Quantitative results showing the effect of TPE.

Shared parts	mAP_{Total}	AP_{50}	AP_{75}	IoU
B-E	36.8	60.7	40.6	79.6
B-E-D	**42.8**	64.9	**50.1**	80.9
B-E-D-Q	42.2	**64.9**	49.3	**85.2**

Table 5: Showing the effect of increasing the joint representations between tasks.

Method	mAP_{Total}	AP_{50}	AP_{75}	IoU
MOD	41.3	**65.5**	47.1	N/A
MOS	N/A	N/A	N/A	78.9
MTL	**42.2**	64.9	**49.3**	**85.2**

Table 6: Comparing the Detection, Segmentation, and MTL architecture.

4.3.3 Effect of TPE:

Building on the results of early temporal aggregation in Table 2, we perform this comparison on the early temporal setup as shown in the left part of Fig. 4. As expected, results in Table 4, show 2% mAP improvement over the variant without TPE.

4.3.4 Effect of the temporal window size T:

We evaluate the effect of the increased window size, for $T = 1, 2, 4$. Results in Table 3 show increased performance with the increase of T. However, a saturation barrier is hit at $T = 4$.

4.3.5 MTL Shared representations

We evaluate the effect of increasing the shared modules and shrinking the task-specific modules across different tasks. Starting with sharing only the encoders,i.e., CNN backbone and the transformer encoder(shared B+E), going through sharing the transformer decoder besides the encoders(shared B+E+D), and ending with sharing the CNN backbone and the full transformer architecture which means both tasks have the same learnable object queries(shared B+E+D+Q), which is termed as Unified Transformer (UT). The results mentioned in Table 5 show an improvement in the results as the joint part between the two tasks increases. The three approaches are demonstrated in Fig. 2.

4.3.6 MTL vs. individual models evaluation

To show the value of our MTL architecture, we compare it against the individual tasks models. We refer to the segmentation model as MOS, the detection as MOD and the joint as MTL in Table 6. In the three architectures we have an early spatio-temporal aggregation, Sec. 3.4.

4.4 Bench-marking against state-of-the-art

We evaluate our proposed cascaded MTL-Transformer network on the KittiMoSeg dataset (56). We have re-run the whole mentioned architectures in Table 7 on our input image resolution,i.e., 512×512, except for the ones which have the † symbol that indicates the reported accuracy are adopted from the original paper and their input image size is 1224×256. To be able to compare against Trans2Seg (74), we replace its simple encoder with our spatio-temporal encoder described at Sec. 3.4.

Our proposed approach outperforms the SOTA with 0.3% for detection task and with 5.7% for the semantic segmentation task. The reasons behind our improvement is two fold. 1) Exploiting the shared representations between both tasks while utilizing the spatial and temporal correlations. 2) The cascaded MTL approach provides a better representation for motion as the learnable object queries are shared between both tasks, which enables the segmentation head to leverage from the detection features.

Method		mAP_{Total}	AP_{50}	AP_{75}	IoU
MODNet(62)	DarkNet53 (RGB + OF)	35.24	59.87	38.20	72.0
	ResNet50 (RGB + OF)	32.04	61.60	29.47	71.8
FuseMODNet (56)	(RGB + OF) †	N/A	N/A	N/A	74.2
	(RGB + LIDAR) †	N/A	N/A	N/A	75.3
RST-MODNet(55)	LSTM-Late	N/A	N/A	N/A	69.5
	LSTM-Multistage	N/A	N/A	N/A	71.4
	LSTM-Multistage †	N/A	N/A	N/A	76.3
MPNet (67) †		31.52	59.92	32.21	69.3
VM-MODNet (57) †		N/A	N/A	N/A	77.6
Monocular Instance Motion Segmentation (47)		40.60	59.21	49.28	79.5
MODETR (44)		39.29	63.70	49.15	N/A
ST-DETR (45)		41.90	64.80	**50.03**	N/A
Trans2Seg (74)		N/A	N/A	N/A	78.9
MTL-TransMODS (ours)		**42.23**	**64.90**	49.30	**85.2**

Table 7: Quantitative evaluation on KittiMoSeg dataset (56) for our proposed cascaded joint detection and motion segmentation network.

Figure 5: MTL output and attention maps visualization across different timestamps. a) shows the MTL output for both detection and segmentation tasks. b) shows the attention maps.

4.5 Qualitative results

Figure 5 demonstrates the MTL output, and attention maps visualization across different timestamps, where row (a) shows the MTL output for both detection and segmentation tasks, row (b) shows the attention maps for the learnable object queries. More qualitative results can be found on the following link.

5 Conclusion

In this paper, we presented a Spatio-Temporal cascaded-MTL Transformer-based architecture for joint MOD and MOS that showing the state-of-the-art performance on the KittiMoSeg dataset (56). We compared the MTL setup against the individual tasks, which shows 1% mAP improvement on MOD and 6.3.% IoU on MOS. The cascaded MTL approach suggests a clear advantage on the dimension of fast inference in the shared MTL architecture over the individual models, which will almost double the inference time and the memory footprint over the shared model, besides the advantage on the dimension of the accuracy where it outperforms the SOTA with 0.3% for detection task and with 5.7% for the semantic segmentation task.

References

[1] Ali Athar, Sabarinath Mahadevan, Aljosa Osep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *European Conference on Computer Vision*, pages 158–177. Springer, 2020.

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016.

[3] Richard F Betzel and Danielle S Bassett. Multi-scale brain networks. *Neuroimage*, 160:73–83, 2017.

[4] PJ Burt, JR Bergen, Rajesh Hingorani, R Kolczynski, WA Lee, A Leung, J Lubin, and H Shvayster. Object tracking with a moving camera. In *Proceedings Workshop on Visual Motion*, pages 2–3. IEEE Computer Society, 1989.

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[6] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. *arXiv preprint arXiv:2103.14899*, 2021.

[7] Xinyang Chen, Sinan Wang, Bo Fu, Mingsheng Long, and Jianmin Wang. Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning. 2019.

[8] Zedu Chen, Bineng Zhong, Guorong Li, Shengping Zhang, and Rongrong Ji. Siamese box adaptive network for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6668–6677, 2020.

[9] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *arXiv preprint arXiv:2104.13840*, 2021.

[10] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. *arXiv preprint arXiv:2102.10882*, 2021.

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Scharwächter, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset. In *CVPR Workshop on the Future of Datasets in Vision*, volume 2, 2015.

[12] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

[13] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3150–3158, 2016.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. Ieee, 2009.

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[16] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[17] Lin Gui, Ruifeng Xu, Qin Lu, Jiachen Du, and Yu Zhou. Negative transfer detection in transductive transfer learning. *International Journal of Machine Learning and Cybernetics*, 9(2):185–197, 2018.

[18] Dongyan Guo, Jun Wang, Ying Cui, Zhenhua Wang, and Shengyong Chen. Siamcar: Siamese fully convolutional classification and regression for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6269–6277, 2020.

[19] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *arXiv preprint arXiv:2103.00112*, 2021.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[21] Gim Hee Lee, Friedrich Faundorfer, and Marc Pollefeys. Motion estimation for self-driving cars with a generalized camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2746–2753, 2013.

[22] Ronghang Hu and Amanpreet Singh. Unit: Multimodal multitask learning with a unified transformer. *arXiv preprint arXiv:2102.10772*, 2021.

[23] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017.

[24] Moritz Kampelmühler, Michael G Müller, and Christoph Feichtenhofer. Camera-based vehicle velocity estimation from monocular video. *arXiv preprint arXiv:1802.07094*, 2018.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2661–2671, 2019.

[27] Zihang Lai, Erika Lu, and Weidi Xie. Mast: A memory-augmented self-supervised tracker. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6479–6488, 2020.

[28] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019.

[29] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8971–8980, 2018.

[30] Yu Li, Zhuoran Shen, and Ying Shan. Fast video object segmentation using the global context module. In *European Conference on Computer Vision*, pages 735–750. Springer, 2020.

[31] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

[32] Yongqing Liang, Xin Li, Navid Jafari, and Qin Chen. Video object segmentation with adaptive feature bank and uncertain-region refinement. *arXiv preprint arXiv:2010.07958*, 2020.

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[34] Shengchao Liu, Yingyu Liang, and Anthony Gitter. Loss-balanced task weighting to reduce negative transfer in multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9977–9978, 2019.

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[36] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[37] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *International conference on machine learning*, pages 2208–2217. PMLR, 2017.

[38] Hongjing Lu, Ying Nian Wu, and Keith J Holyoak. Emergence of analogy from relation learning. *Proceedings of the National Academy of Sciences*, 116(10):4176–4181, 2019.

[39] Xinkai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. *arXiv preprint arXiv:2007.07020*, 2020.

[40] Davide Maltoni and Vincenzo Lomonaco. Continuous learning in single-incremental-task scenarios. *Neural Networks*, 116:56–73, 2019.

[41] Adam H Marblestone, Greg Wayne, and Konrad P Kording. Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10:94, 2016.

[42] Jianbiao Mei, Mengmeng Wang, Yeneng Lin, and Yong Liu. Transvos: Video object segmentation with transformers. *arXiv preprint arXiv:2106.00588*, 2021.

[43] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. *arXiv preprint arXiv:2101.02702*, 2021.

[44] Eslam Mohamed and Ahmad El-Sallab. Modetr: Moving object detection with transformers, 2021.

[45] Eslam Mohamed and Ahmad El-Sallab. St-detr: Spatio-temporal object traces attention detection transformer, 2021.

[46] Eslam Mohamed, Mahmoud Ewaisha, Mennatullah Siam, Hazem Rashed, Senthil Yogamani, and Ahmad El-Sallab. Instancemotseg: Real-time instance motion segmentation for autonomous driving. *arXiv preprint arXiv:2008.07008*, 2020.

[47] Eslam Mohamed, Mahmoud Ewaisha, Mennatullah Siam, Hazem Rashed, Senthil Yogamani, Waleed Hamdy, Muhammad Helmi, and Ahmad El-Sallab. Monocular instance motion segmentation for autonomous driving: Kitti instancemotseg dataset and multi-task baseline. *arXiv preprint arXiv:2008.07008*, 2020.

[48] Maryam M Najafabadi, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of big data*, 2(1):1–21, 2015.

[49] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.

[50] Mohamed Okasha and Brett Newman. Relative motion guidance, navigation and control for autonomous orbital rendezvous. In *AIAA Guidance, Navigation, and Control Conference*, pages 08–11. SciELO Brasil, 2011.

[51] Brian Paden, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, 1(1):33–55, 2016.

[52] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

[53] Robi Polikar, Lalita Upda, Satish S Upda, and Vasant Honavar. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE transactions on systems, man, and cybernetics, part C (applications and reviews)*, 31(4):497–508, 2001.

[54] Lalintha G Polpitiya and Kamal Premaratne. Real-time detection and prediction of relative motion of moving objects in autonomous driving. In *The Thirty-Third International Flairs Conference*, 2020.

[55] Mohamed Ramzy, Hazem Rashed, Ahmad El Sallab, and Senthil Yogamani. Rst-modnet: Real-time spatio-temporal moving object detection for autonomous driving. *arXiv preprint arXiv:1912.00438*, 2019.

[56] Hazem Rashed, Mohamed Ramzy, Victor Vaquero, Ahmad El Sallab, Ganesh Sistu, and Senthil Yogamani. Fusemodnet: Real-time camera and lidar based moving object detection for robust low-light autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[57] Hazem Rashed, Ahmad El Sallab, and Senthil Yogamani. Vm-modnet: Vehicle motion aware moving object detection for autonomous driving. *arXiv preprint arXiv:2104.10985*, 2021.

[58] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *European Conference on Computer Vision*, pages 629–645. Springer, 2020.

[59] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *arXiv preprint arXiv:1506.04214*, 2015.

[60] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[61] Mennatullah Siam, Heba Mahgoub, Mohamed Zahran, Senthil Yogamani, Martin Jagersand, and Ahmad El-Sallab. Modnet: Moving object detection network with motion and appearance for autonomous driving. *arXiv preprint arXiv:1709.04821*, 2017.

[62] Mennatullah Siam, Heba Mahgoub, Mohamed Zahran, Senthil Yogamani, Martin Jagersand, and Ahmad El-Sallab. MODNet: Motion and appearance based moving object detection network for autonomous driving. In *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2859–2864, 2018.

[63] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2325–2333, 2016.

[64] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. *arXiv preprint arXiv:2105.05633*, 2021.

[65] Peize Sun, Yi Jiang, Rufeng Zhang, Enze Xie, Jinkun Cao, Xinting Hu, Tao Kong, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple-object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020.

[66] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.

[67] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4481–4490, 2017.

[68] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[69] Johan Vertens, Abhinav Valada, and Wolfram Burgard. Smsnet: Semantic motion segmentation using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 582–589. IEEE, 2017.

[70] Ricardo Vilalta and Youssef Drissi. A perspective view and survey of meta-learning. *Artificial intelligence review*, 18(2):77–95, 2002.

[71] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1328–1338, 2019.

[72] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *arXiv preprint arXiv:2102.12122*, 2021.

[73] Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11293–11302, 2019.

[74] Enze Xie, Wenjia Wang, Wenhai Wang, Peize Sun, Hang Xu, Ding Liang, and Ping Luo. Segmenting transparent object in the wild with transformer. *arXiv preprint arXiv:2101.08461*, 2021.

[75] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021.

[76] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021.

[77] Wen Zhang, Lingfei Deng, Lei Zhang, and Dongrui Wu. Overcoming negative transfer: A survey. *arXiv preprint arXiv:2009.00909*, 2020.

[78] Yu Zhang and Qiang Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

[79] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *arXiv preprint arXiv:2012.15840*, 2020.

[80] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

[81] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.