Noname manuscript No.
(will be inserted by the editor)

Multi-Task Learning with Multi-Query Transformer for Dense Prediction

Yangyang Xu[1] · Xiangtai Li[2] · Haobo Yuan[1] · Yibo Yang[3] · Jing Zhang[4] · Yunhai Tong[2] · Lefei Zhang[1] · Dacheng Tao[3]

Received: date / Accepted: date

Abstract Previous multi-task dense prediction studies developed complex pipelines such as multi-modal distillations in multiple stages or searching for task relational contexts for each task. The core insight beyond these methods is to maximize the mutual effects between each task. Inspired by the recent query-based Transformers, we propose a simpler pipeline named Multi-Query Transformer (MQTransformer) that is equipped with multiple queries from different tasks to facilitate the reasoning among multiple tasks and simplify the cross task pipeline. Instead of modeling the dense per-pixel context among different tasks, we seek a task-specific proxy to perform cross-task reasoning via multiple queries where each query encodes the task-related context. The MQTransformer is composed of three key components: shared encoder, cross task attention and shared decoder. We first model each task with a task-relevant and scale-aware query, and then both the image feature output by the feature extractor and the task-relevant query feature are fed into the shared encoder, thus encoding the query feature from the image feature. Secondly, we design a cross task attention module to reason the dependencies among *multiple tasks and feature scales* from two perspectives including different tasks of the same scale and different scales of the same task. Then we use a shared decoder to gradually refine the image features with the reasoned query features from different tasks. Extensive experiment results on two dense prediction datasets (NYUD-v2 and PASCAL-Context) show that the proposed method is an effective approach and achieves the state-of-the-art result.

Keywords Scene Understanding · Multi-Task Learning · Dense Prediction · Vision Transformer

1 Introduction

Humans are excellent at accomplishing multiple tasks simultaneously in the same scene. In computer vision, Multi-Task Dense Prediction (MTDP) Prakash et al. (2021); Ghiasi et al. (2021) requires a model to directly output multiple task results such as semantic segmentation, depth prediction, normal prediction and edge detection. High-performance MTDP results are important for several applications including robot navigation and planning. Previous works use Convolution Neural Networks (CNNs) to capture different types of features for each task. Several approaches Kundu et al. (2019); Xu et al. (2018); Vandenhende et al. (2020); Phillips et al. (2021); Bruggemann et al. (2021) exploring task association then achieved impressive results in MTDP. Recently, transformer-based methods achieve promising results on various tasks Dosovitskiy et al. (2021); Wang et al. (2021b); Liu et al. (2021). However, how to adopt the vision transformer into MTDP is still an unexplored question.

Current MTDP methods are accomplished by learning a shared representation for multi-task features, and can be categorized into the *encoder-focused* (Fig. 1(a)) and *decoder-focused* (Fig. 1(b)) methods based on *where*

The first two authors contribute equally.

✉ Lefei Zhang (*Corresponding Author.*)
E-mail: zhanglefei@whu.edu.cn

[1] School of Computer Science, Wuhan University, Wuhan, China
[2] Key Laboratory of Machine Perception, MOE, School of Artificial Intelligence, Peking University, Beijing, China
[3] JD Explore Academy, JD.com, Beijing, China
[4] The University of Sydney, Sydney, Commonwealth of Australia

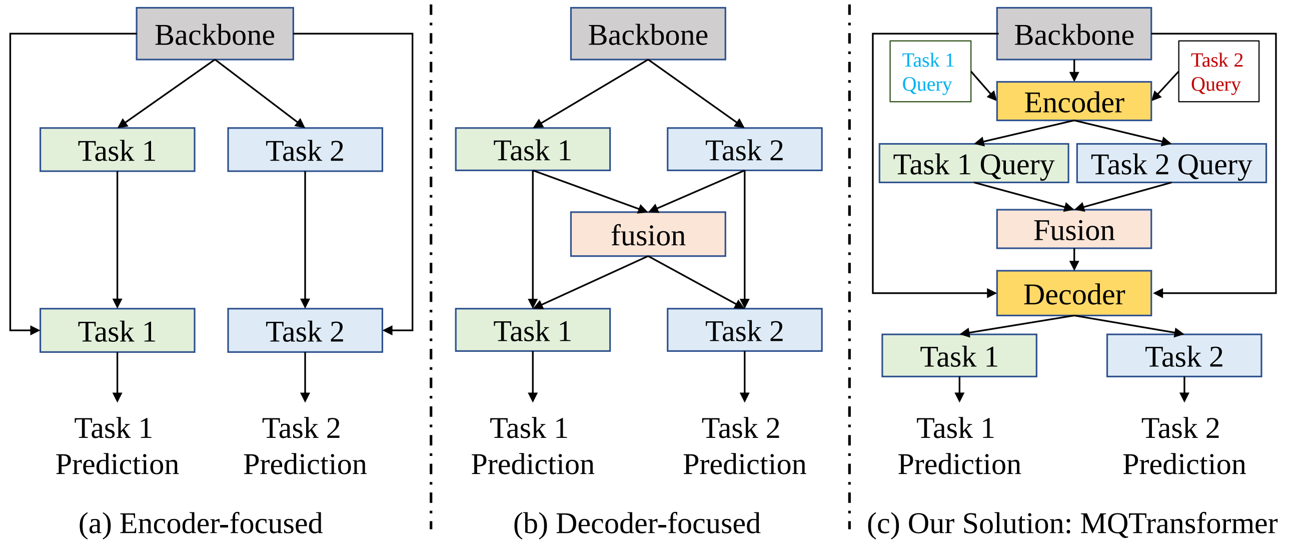
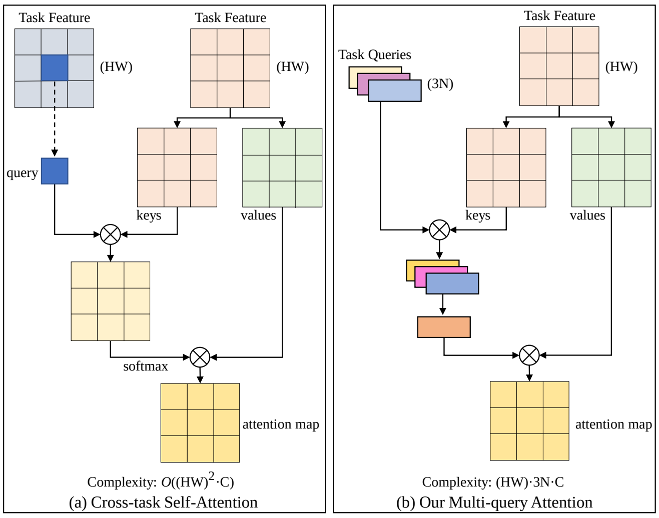


Fig. 1: Illustration of different approaches for solving the MTDP task. We separate the encoder-focused and decoder-focused models based on where the task fusion occurs. (a) The baseline method proposed in Liu et al. (2019) selects features from the shared encoder. (b) The Fusion in (b) is performed in different manners, such as spatial attention Zhang et al. (2019), distillation Xu et al. (2018); Bruggemann et al. (2021). (c) Our proposed model adopts multiple task-relevant queries with Transformer and performs joint learning among different queries with a shared encoder and decoder. Best view it on screen.

the fusion of task-specific features occurs. As shown in Fig. 1(a), the encoder-focused methods Liu et al. (2019); Kendall et al. (2018); Wang et al. (2021a) share a generic feature and each task has a specific head to make prediction. However, encoder-focused methods result in each task being individual and there is no task association. To this end, the decoder-focused models Bruggemann et al. (2021); Zhang et al. (2019); Xu et al. (2018); Vandenhende et al. (2020) focus on the relationships among tasks via a variety of approaches. Neural architecture search (NAS) Bruggemann et al. (2021); Sun et al. (2020); Yang et al. (2021) and knowledge distillation Shu et al. (2021) techniques are leveraged to find the complementary information via the associated tasks sharing. For example, the work Sun et al. (2020) is designed for determining the effective feature sharing across tasks. Decoder-focused models for MDPT are usually of high computational cost due to the multiple states and roads needed for the interaction, such as the multi-modal distillation module of ATRC Bruggemann et al. (2021). However, decoder-focused methods often contain complex pipelines and need specific human design.

Recently, several transformer models Carion et al. (2020) show simpler pipelines and promising results. In particular, the dense prediction Transformer (DPT) Ranftl et al. (2021) exploits vision Transformers as a backbone for dense prediction tasks. The encoder-decoder architecture with object query Vaswani et al. (2017); Carion et al. (2020) mechanism is proved to be effective in reasoning the relations among multiple objects and the global feature context. The object query design jointly tackles the interaction in two aspects: the relationship among queries and interaction between feature and query. These successes inspire us to explore the potential of the multi-query Transformer with multiple queries for multiple tasks learning where each query represents one specific task. Each task can be correlated via the query reasoning among different tasks.

In this work, we introduce MQTransformer for multi-task learning of five dense prediction tasks, including semantic segmentation, depth estimation, surface normals prediction, semantic edge detection, and saliency detection. As illustrated in Fig. 1 (c), we introduce multiple task-relevant queries (according to task number) as the input of the encoder. The encoder outputs the learnt task-relevant query feature of each task. We show two tasks for illustration purposes. Moreover, we add cross-scale learning in modeling these task queries. Depending on the number of tasks, these query features are concatenated into two sets, *i.e.*, query features of the same scale with different tasks are concatenated, and query features of the same task with different scales are concatenated. The concatenated query features collect representation across scale and task. In order to facilitate the interaction, we introduce a cross task attention module composed of cross-scale attention and cross task attention. The query features after cross task attention serve as the input of the following shared decoder. Finally, the decoder outputs the corresponding



(a) Cross-task Self-Attention

(b) Our Multi-query Attention

Fig. 2: The structure of different attention mechanisms for MTDP: (a) The cross-task self-attention mechanism with pixel-wised affinity learning. (b) Our model adopts multiple task-relevant queries with attention. Different color represents different task. By replacing pixel-level affinity calculation across task, our method introduce task-aware queries to encode task aware context and perform cross-task learning in an efficient manner. Best view it on screen.

feature maps according to the number of tasks and applies them separately to task prediction. Since the task association is performed on query level, we avoid heavy pixel-level context computation as used in previous work Bruggemann et al. (2021).

As demonstrated in Fig. 2, we show the difference between our multi-query attention and self-attention in more detail. Fig. 2 (a) shows the query, key and value in self-attention from the different task feature. Self-attention first calculates the dot product of the query with keys, and then uses a softmax function to obtain the attention map on the value. We initialize independent multiple task-relevant queries as the attention mechanism input of the query and the key and value come from the image feature. We compute the task queries and key to obtain task queries with a valuable feature. Finally, we compute the dot product of the task queries and value, resulting in the final values, as depicted in Fig. 2 (b). In addition, the computational complexity of our method is more inexpensive than cross-task self-attention. As shown in the experiment, we show that our proposed query based methods are

In our experiments, our method is compatible with a wide range of backbones, such as CNN Sun et al. (2019) and vision Transformer Liu et al. (2021); Zhang et al. (2022a). We show the effectiveness of our query interaction among different tasks in various settings on different task metrics. Moreover, our experimental results demonstrate that the MQTransformer achieves better results than the previous methods in Fig. 1 (a) and (b). Our finding demonstrates that multiple task-relevant queries play an important role in MTDP. Another key insight is that the interaction of query features captures dependencies for different tasks. The main contributions of this work can be summarized as follows:

1) We propose a new method named MQTransformer to incorporate the multi-query and different scales of the feature extraction for multi-task learning of dense prediction tasks, resulting in a simpler and stronger encoder-decoder network. *To the best of our knowledge, we are the first to explore multi-task dense prediction with task-relevant queries.*
2) We present the cross task attention module to enable sufficient interaction of the task-relevant query features across both *scale* and *task*. Thus, the dependencies carried by each task query feature can be maximally refined.
3) We conduct extensive experiments on two dense prediction datasets: NYUD-v2 and PASCAL-Context. Extensive experiment results show that MQTransformer consistently outperforms several competitive baseline methods. Notably, the proposed model exhibits *significant* improvements for different tasks, compared with the latest state-of-the-art results.

2 Related Work

Multi-Task learning for Dense Prediction (MTDP). As deep neural networks have gradually become the mainstream framework for computer vision, multi-task learning has also developed tremendously. Multi-task learning is typically used when related tasks can make predictions simultaneously. Many multi-task learning models Jalali et al. (2010); Kundu et al. (2019); Ling et al. (2020); Kanakis et al. (2020); Misra et al. (2016); Strezoski et al. (2019); Zhenyu et al. (2018); Vandenhende et al. (2020); Ghiasi et al. (2021) have been widely used in various computer vision tasks. Recent work like Ling et al. (2020) improves multi-task learning performance by co-propagating intra-task and inter-task pattern structures into task-level patterns, encapsulating them into end-to-end networks. Furthermore, Bruggemann et al. (2021) proposed an Adaptive Task-Relational Context (ATRC) module, which leverage a multi-modal distillation module to sample the features of all available contexts for each task pair, explicitly considering task relationships while using self-attention to enrich the features of the target task. Georgescu et al. (2021) designed an anomaly detection framework based on multi-task learning through self-supervised and model distillation tasks. CNN based architectures for dense prediction task Tateno et al. (2018); Takahashi and Mitsufuji (2021); Huang et al. (2021). In Shu et al. (2021); Bruggemann et al. (2021), the knowledge distillation is employed for dense prediction tasks. In MTI-Net Vandenhende et al. (2020), the multi-scale multi-modal distillation unit is designed to perform task interaction on very scale features and the aggregation unit aggregates the task features from all scales to produce the final predictions for each task. Recent work has attempted to use search learning to mold an optimal architecture. The vast majority of these methods are trained using multi-scale features, where a grid search is typically used to select appropriate weights Bruggemann et al. (2021). Some works focus on building encoder-decoder networks to back-propagate high-level semantic contextual information from small-scale features to large-scale features through layer-by-layer up-sampling. Several works Yang et al. (2020); Li et al. (2020) integrates the FPN Lin et al. (2017) backbone bottom-up to generate multi-scale feature pyramids for dense prediction. Dense Prediction Transformer Wang et al. (2021a); Ranftl et al. (2021) based encoder-decoder employs attention Vaswani et al. (2017) computational operations to obtain fine-grained and globally consistent features to perform dense prediction tasks. In this paper, we explore Multi-query Transformer for MTDP and propose a totally new method to adopt vision transformer into MTDP.

Vision Transformers. Currently, due to the unprecedented success of the Transformer Vaswani et al. (2017) in natural language processing (NLP), many computer vision efforts are enthusiastically applying the Transformer to vision tasks Li et al. (2021); Bumsoo et al. (2021); Ding et al. (2021); Xin et al. (2021); Hehe et al. (2021); Jack et al. (2021), starting with the Vision Transformer (ViT) Dosovitskiy et al. (2021). The image is segmented into a fixed number of patches and they are embedded into a "token" as input. And project them into the feature space, where the converter encoder computes the queries, keys and values, to generate the final result Guo et al. (2021). The encoder-decoder based design Transformer has been applied to object detection, image classification and instance segmentation tasks Carion et al. (2020); Liu et al. (2021); Wang et al. (2021b) and has demonstrated the great potential of attention-based models. Various Transformer variants such as Deformable DETR Zhu et al. (2021), T2T-ViT Yuan et al. (2021b), PVT Wang et al. (2021a), Swin-Transformer Liu et al. (2021) and ViTAE Xu et al. (2021). DeiT Hugo et al. (2021) further extended ViT employing a new distillation method in which the Transformer learns more from images than others with similar Transformers. Moreover, several vision transformers Wang et al. (2021b); Yuan et al. (2021a); Li et al. (2022b); Xu et al. (2022); Li et al. (2022a); Zhang et al. (2022b) adopt DETR-like architecture to simplify the complex pipeline. Unit Hu and Singh (2021) proposes to learn cross-modal and cross-task using DETR-like model. With respect to dense scene understanding tasks, attention mechanisms approach to efficiently maintain multi-scale features in the network. InvPT Ye and Xu (2022) models multi-task feature interaction in both spatial and all-task contexts and construct an InvPt transformer encoder and decoder based on multi-scale features.

Differs from previous models Ranftl et al. (2021), our work explores vision Transformers for multi-task representation and is complementary to these efforts where all the vision Transformers serve as the feature extractor or instance level learner. Moreover, different from the object query in DETR Carion et al. (2020) which represents the object in the scene, our task specific query explores the relationship context among different tasks where the task specific features are already encoded before reasoning.

3 Method

Overview. In this part, we will first introduce the problem formulation and motivation of our approach in Sec. 3.1. We present the details of our method and insights in Sec. 3.2. Then we give the description of the network architecture and loss function in Sec. 3.3.

3.1 Problem Formulation and Motivation

To facilitate the description of the components in the model, we first briefly introduce the basic notation of the Transformer Vaswani et al. (2017). For short, we term these operation including Multi-Head Self-Attention (MHSA) and Multi-Layer Perceptron (MLP), Layer Normalisation (LN). Moreover, the pyramid features that come form the feature extractor are $\{x_1, x_2, \ldots, x_s\}$, $x_1 \in \mathbb{R}^{H\times W\times C}$, $x_2 \in \mathbb{R}^{H/2\times W/2\times C}$. s denotes the feature scale. H, W and C are the height, width and channel. The task-relevant and scale-aware query is represented as $p_{(s,t)} \in \mathbb{R}^{N\times C}$. $\{p_{(1,1)}, \ldots, p_{(1,t)}\}$ is the set of task-relevant queries (scale 1 with t tasks). Note that we only consider two scales and more details can be found in Sec. 4. TN means Task Number.

The goal of MTDP is to directly output several independent dense prediction maps. Previous works including decoder focused and encoder focused pay more attention to the design space of a cross task interaction.

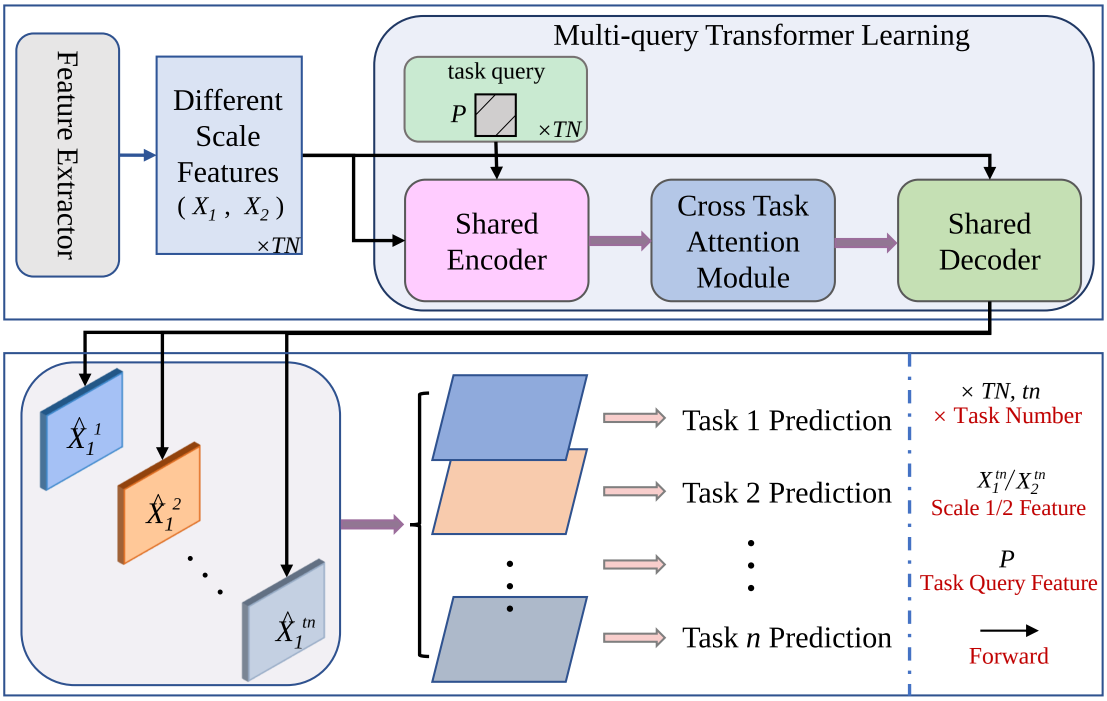


Fig. 3: An overview of MQTransformer. The MQTransformer represents multi-query via task queries and performs joint multi-task learning. Here, we show an example of task-specific policy learned using our method for the two-scale features (X_1 and X_2). A Multi-query Transformer is designed to extract image features from X_s and P and it outputs task-relevant features for MTDP.

There are two main problems: One is the huge computation cost and affinity memory in case of the cross task even only one scale considered. The other is the complex pipeline for modeling each decoder such as NAS. Moreover, *how to adapt vision transformer for MTDP is still an open question.* To tackle those two problems and explore a new vision transformer architecture for MTDP, we present a multi-query Transformer.

3.2 Multi-Query Transformer (MQTransformer)

Overview. As shown in Fig. 3, our MQTransformer contains feature extractor and Multi-query transformer learning modules. The former extract features while the latter is a multi-query transformer to perform cross-task association. The latter contains a shared encoder, a cross-task attention module and a shared decoder where the task-relevant queries are the inputs and perform that task association.

Feature Extractor. We first extract image features for each input image. It contains a backbone network (Convolution Network Sun et al. (2019) or Vision Transformer Liu et al. (2021)) with Feature Pyramid Network Lin et al. (2017) as neck. This results in a set of multi-scale features. We fuse these features into one single high-resolution feature map via addition and bilinear interpolation upsampling. Moreover, to generate scale aware feature, we perform a bilinear downsampling operation on the take-aware fused feature. As shown in Fig. 4, we take two tasks ($TN = 2$) and two-scale features as examples. Thus we obtain four features, two tasks with two scales (X_s^1 means scale s with task number 1).

Motivation of Multi-Query Transformer. Our key insight is to replace complex pixel-wised task association with task-aware queries. To achieve that, we use a shared encoder to encode each task-aware feature into their corresponding queries. Then the task association can be performed within these queries. Finally, the refined task-aware features can be decoded from the reasoned queries with another shared decoder.

Shared Encoder. We feed the extracted features and the task-relevant query into a shared encoder. Such encoder builds the one to one connection between features and queries on different tasks and scales which is shown in the pink box in Fig. 4.

A task-relevant query $p \in \mathbb{R}^{N\times C}$ is initialized first for each task and the learned positional encodings Vaswani et al. (2017) $e_q \in \mathbb{R}^{N\times C}$ (see Fig. 4 blue circle) is added before regularization using LN and then used as a query input for the MHSA. The feature $x \in \mathbb{R}^{H\times W\times C}$ from the feature extractor is reshaped to $x \in \mathbb{R}^{HW\times C}$, and

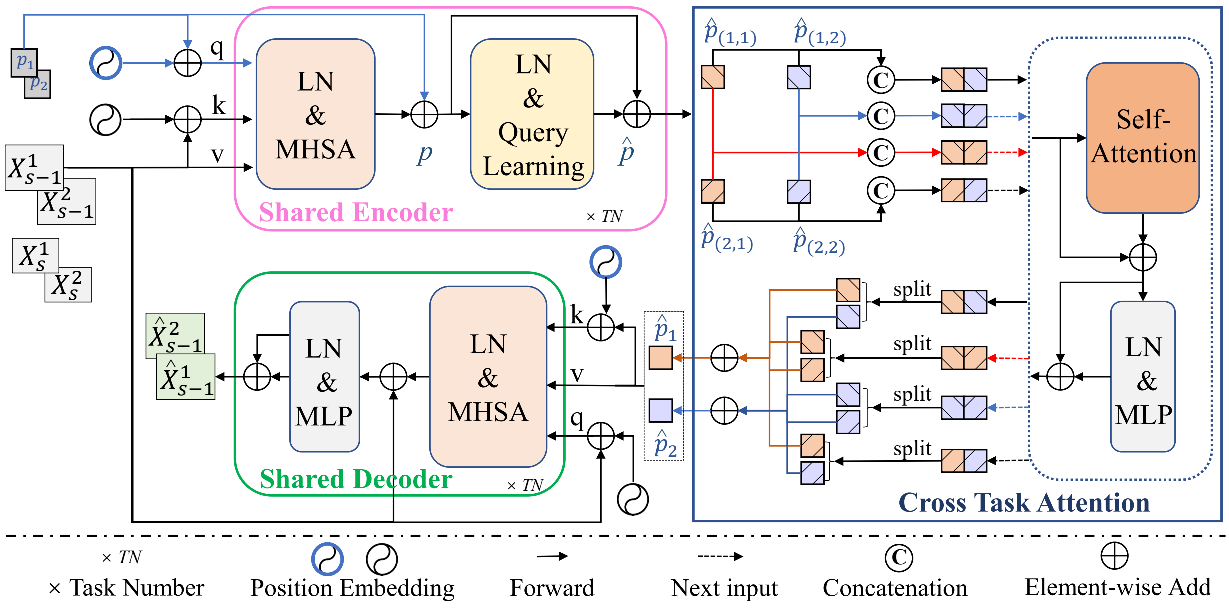


Fig. 4: Illustration of the Multi-Query Transformer. The TN is two. We use a shared encoder to obtain the valuable task-relevant query through the input of image features X and task-relevant query $p_{(s,t)}$ (s and t denotes scale and the task number, respectively). The task-relevant queries $\hat{p}$ after the operation of upstream are concatenated by cross-scale and cross-task, respectively. The cross task attention module leads to the concatenated task-relevant query interaction via self-attention. Then, the task-relevant query is split into a shared decoder where it outputs the task-relevant feature maps for final predictions.

added to positional encoding $e_k \in \mathbb{R}^{HW\times C}$ (see Fig. 4 black circle), and then input to the MHSA as key and value. The formulation of MHSA is

$$\mathrm{MHSA}(Q,K,V) = \mathrm{softmax}(QK^T/\sqrt{d})V, \tag{1}$$

where $Q \in \mathbb{R}^{N\times C}$, $K \in \mathbb{R}^{HW\times C}$ and $V \in \mathbb{R}^{HW\times C}$ are the query, key and value tensors; d denotes dimension C; $\mathrm{MHSA}(Q,K,V) \in \mathbb{R}^{N\times C}$. And then the encoder is calculated as follows where task-relevant features are encoded into query format:

$$\hat{p} = p + \mathrm{MHSA}(Q = \mathrm{LN}(p), K = \mathrm{LN}(x), V = \mathrm{LN}(x)). \tag{2}$$

Query Learning in Encoder. Then the task-relevant query $\hat{p}$ captures the task-relevant context. We use a linear layer (linear mapping) along with an identity connection as:

$$\hat{p}^l = \hat{p} + \mathrm{MLP}(\mathrm{LN}(\hat{p})), \tag{3}$$

where the $\hat{p} \in \mathbb{R}^{N\times C}$ is the result after the query learning step. To enhance query learning, we adopt an extra MLP and non-linear GELU with an identity connection to update the task-relevant query. This operation can further reduce the inductive biases and enhance the communications in query. Each task-relevant query is processed independently and only gradually interacts with the image feature to learn dependencies.

As shown in Fig 4, we take queries with two task with two scale for illustration. Each query follow the same pipeline above. In the end, we obtain four different queries. ($\hat{p}_{(1,1)}$ $\hat{p}_{(1,2)}$ and $\hat{p}_{(2,1)}$ $\hat{p}_{(2,2)}$).

Cross Task Attention. Cross Task Attention module aims to perform cross-task and cross-scale learning for task-aware and scale-aware query($\hat{p}_{(1,1)}$ $\hat{p}_{(1,2)}$ and $\hat{p}_{(2,1)}$ $\hat{p}_{(2,2)}$). We concatenate the task-relevant query of the same scale from all tasks: $Q_1 = \mathrm{concat}(\hat{p}_{(1,1)}, \hat{p}_{(1,2)})$ and $Q_2 = \mathrm{concat}(\hat{p}_{(2,1)}, \hat{p}_{(2,2)})$. $Q_1 \in \mathbb{R}^{(N+N)\times C}$ and $Q_2 \in \mathbb{R}^{(N+N)\times C}$ are cross-task query. Moreover, we also concatenate the task-relevant query of same task from all scales. $Q_3 = \mathrm{concat}(\hat{p}_{(1,1)}, \hat{p}_{(2,1)})$ and $Q_4 = \mathrm{concat}(\hat{p}_{(1,2)}, \hat{p}_{(2,2)})$. $Q_3 \in \mathbb{R}^{(N+N)\times C}$ and $Q_4 \in \mathbb{R}^{(N+N)\times C}$ are cross-scale query. Then we perform MHSA operation as follows:

$$\hat{Q} = Q + \mathrm{MHSA}(Q), \tag{4}$$

$$\hat{Q} = \hat{Q} + \mathrm{MLP}(\mathrm{LN}(\hat{Q})). \tag{5}$$

Then we split the result $\hat{Q}_1 \in \mathbb{R}^{(N+N)\times C}$ into ($\hat{p}_{(1,1)}$, $\hat{p}_{(1,2)}$). Q_2, Q_3 and Q_4 do the same operation separately and we get ($\hat{p}_{(2,1)}$, $\hat{p}_{(2,2)}$), ($\hat{p}_{(*,1)}$, $\hat{p}'_{(*,1)}$) and ($\hat{p}_{(*,2)}$, $\hat{p}'_{(*,2)}$) ($*$ denotes cross scale task-relevant query). The cross task attention module was designed to enhance the communication of the multiple task-relevant queries from different scales. Finally, we employ residuals to

Table 1: Complexity comparison. We compare the computational complexity of the different schemes for cross-task communication. H and W are the image height and width. C, N, and K are hyper-parameters. We adopt $C = 256$, $N = 100$, and $K = 9$ when calculating GFLOPs.

Method	Complexity	GFLOPs	
		64×64	128×128
No Communication	0	0	0
Global Context Bruggemann et al. (2021); Dosovitskiy et al. (2021)	$\mathcal{O}((HW)^2C)$	9.74	142.83
Local Context Bruggemann et al. (2021); Liu et al. (2021)	$\mathcal{O}(HWC^2 \cdot K^2)$	21.74	86.98
Cross Task Attention (Ours)	$\mathcal{O}(C \cdot N^2)$	0.03	0.03

add the task-relevant query of the same task in sequence. Finally, we obtain $\hat{p}_1$ for task 1 and $\hat{p}_2$ for task 2, respectively.

$$\hat{p}_1 = \hat{p}_{(1,1)} + \hat{p}_{(2,1)} + \hat{p}_{(*,1)} + \hat{p}'_{(*,1)}, \tag{6}$$

$$\hat{p}_2 = \hat{p}_{(1,2)} + \hat{p}_{(2,2)} + \hat{p}_{(*,2)} + \hat{p}'_{(*,2)}, \tag{7}$$

where $\hat{p}_1 \in \mathbb{R}^{N\times C}$ and $\hat{p}_2 \in \mathbb{R}^{N\times C}$ are the output of the cross task attention module.

Our cross task attention builds the relation from both scale and task via query level association. Compared with previous pixel-level or patch-level cross-task learning, it avoids heavy affinity costs. Tab. 1 shows the complexity and GFLOPs of query-based attention. Our task-relevant query interaction reduces the required computation (ours: 0.03 GFLOPs v.s. ATRC Bruggemann et al. (2021): 9.74 GFLOPs) with the 64×64 image feature as inputs.

Shared Decoder. The shared decoder takes the output of the previous fused queries and features as input and outputs the refined feature for final prediction. As shown in Fig. 4, the shared decoder contains one MHSA, one LN and one MLP. We use the learned query $\hat{p}$ as key and value, and image feature X_{s-1}. We use $\hat{p}$ and x_{s-1} to perform cross-attention Vaswani et al. (2017). In particular, the q, k and v are input into the MHSA and get the image feature $\hat{x} \in \mathbb{R}^{HW\times C}$. The the task-relevant query and image features interaction in the shared decoder can be written as follows:

$$x = x + \mathrm{MHSA}(Q = \mathrm{LN}(x), K = \mathrm{LN}(\hat{p}), V = \mathrm{LN}(\hat{p})), \tag{8}$$

$$\hat{x} = x + \mathrm{MLP}(\mathrm{LN}(x)). \tag{9}$$

Note that one can also use a non-shared encoder and decoder. However, we argue that the task-aware queries can absorb relevant features in a compact manner without the extra parameters brought by encoding or decoding for this purpose. We verify this in Sec. 4.

3.3 Loss Function

After adopting prediction heads for each task via 1×1 convolution, we employ task-specific loss functions for each task. For semantic segmentation, human part segmentation and saliency detection, the cross-entropy loss is adopted. We use L1-Loss and balance binary cross-entropy losses for the depth, surface normals and edge supervision, respectively. Then final training objective $\mathcal{L}$ for each task can be formulated as follows:

$$\begin{aligned}\mathcal{L} = &\lambda_{seg}\mathcal{L}_{seg} + \lambda_{depth}\mathcal{L}_{depth} + \lambda_{normals}\mathcal{L}_{normals} \\ &+ \lambda_{edge}\mathcal{L}_{edge} + \lambda_{partseg}\mathcal{L}_{partseg} + \lambda_{sal}\mathcal{L}_{sal},\end{aligned} \tag{10}$$

where $\mathcal{L}$ denotes a total loss function. We set λ_{seg}=1.0, λ_{depth}=1.0, $\lambda_{normals}$= 10.0, λ_{edge}=50.0, $\lambda_{partseg}$=2.0, and λ_{sal}=5.0. For fair comparison, we adopt the same setting as ATRC Bruggemann et al. (2021).

4 Experiment

4.1 Experimental setup

Datasets. Following previous works Bruggemann et al. (2021) in MTDP, we use NYUD-v2 Silberman et al. (2012) and PASCAL-Context Chen et al. (2014) datasets. The NYUD-v2 dataset is comprised of video sequences of 795 training and 654 testing images of indoor scenes and contains four tasks including semantic segmentation (SemSeg), Monocular depth estimation (Depth), surface normals prediction (Normals) and semantic edge detection (Bound). PASCAL-Context contains 10,103 training images and 9,637 testing images. It contains five tasks including semantic segmentation (SemSeg), human part segmentation (PartSeg), saliency detection (Sal), surface normals prediction (Normals), and semantic edge detection (Edge).

Implementation details. Swin Transformer (Swin-T, Swin-S and Swin-B indicate Swin Transformer with tiny, small and base, respectively.) Liu et al. (2021), ViTAEv2-S Zhang et al. (2022a) and HRNet (HRNet18, HRNet48) Sun et al. (2019) models are adopted as the feature extractor. We mainly perform ablation studies using Swin Transformer. Our model follows the initialization scheme proposed in Bruggemann et al. (2021). We use Pytorch Framework to implement all the experiments in one codebase. During training, we augment input images during training by random scaling

Table 2: Comparison results on NYUD-v2 dataset. The notation '↓': lower is better. The notation '↑': higher is better. "Params" denotes parameters. We report comparisons on various baseline models (in the first and second sub-figures) and several recent works. We report comparisons on various baseline models (in the first and second sub-figures) and several recent works.

Model	Backbone	Params (M)	GFLOPs (G)	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
multi-task baseline	HRNet18	4.52	17.59	36.35	0.6284	21.02	76.36
multi-task baseline	HRNet48	66.99	103.81	41.96	0.5543	20.36	77.62
multi-task baseline	ViTAEv2-S	22.17	76.44	43.65	0.5971	21.02	76.20
multi-task baseline	Swin-T	32.5	96.29	38.78	0.6312	21.05	75.60
multi-task baseline	Swin-S	53.82	116.63	47.90	0.6053	21.17	76.90
Cross-StitchMisra et al. (2016)	HRNet18	4.52	17.59	36.34	0.6290	20.88	76.38
Pad-NetXu et al. (2018)	HRNet18	5.02	25.18	36.70	0.6264	20.85	76.50
PAPZhang et al. (2019)	HRNet18	4.54	53.04	36.72	0.6178	20.82	76.42
PSDLing et al. (2020)	HRNet18	4.71	21.10	36.69	0.6246	20.87	76.42
NDDR-CNNGao et al. (2019)	HRNet18	4.59	18.68	36.72	0.6288	20.89	76.32
MTI-NetVandenhende et al. (2020)	HRNet18	12.56	19.14	36.61	0.6270	20.85	76.38
ATRCBruggemann et al. (2021)	HRNet18	5.06	25.76	38.90	0.6010	20.48	76.34
ATRCBruggemann et al. (2021)	HRNet48	73.58	196.47	46.27	0.5495	20.20	77.60
MQTransformer	HRNet18	5.23	22.97	40.47	0.5965	20.34	76.60
MQTransformer	ViTAEv2-S	26.18	94.77	48.37	0.5769	20.73	76.90
MQTransformer	Swin-T	35.35	106.02	43.61	0.5979	20.05	76.20
MQTransformer	Swin-S	56.67	126.37	49.18	0.5785	20.81	77.00

with values between 0.5 and 2.0, and random cropping to the input size. We deploy the SGD optimizer with default hyper parameters. The learning rate is set to 0.001 and weight decay is set to 0.0005. All the models are trained for 40k iteration steps with an 8 batch size on the NYUD-v2 dataset. For the PASCAL-Context dataset, we follow the same setting in NYUD-v2. All the models adopt the single inference for both ablation and comparison.

Strong Multi-task baseline. The baseline model has the same architecture MQTransformer. However, it does not contain Multi-Query Transformer prediction heads. It contains TN different heads for different tasks. For Swin Transformer, we adopt the ADE-20k pre-trained model as the strong baseline. For HRNet, we follow the same setting as ATRC Bruggemann et al. (2021). We argue that our baseline models are different from previous method by using Swin Transformer. However, even on such strong baseline, our method can still achieve significant improvements. Moreover, we also prove the effectiveness and generation of our method on CNN backbone in ablation study (Sec. 4.3).

Evaluation Metric. Our evaluation follows the metrics scheme proposed in Bruggemann et al. (2021). Semseg and PartSeg tasks are evaluated with the mean intersection over union (IoU), and Depth task using the root mean square error (rmse). Normals task using mean angular error (mErr), Sal task using maximum F-measure (maxF), and Bound task using the optimal-dataset-scale F-measure (odsF).

4.2 Comparison with the state-of-the-art methods

Results on NYUD-v2. As shown in Tab. 2, we report our MQTransformer results compared with both previous work Bruggemann et al. (2021); Vandenhende et al. (2020) and strong multi-task baseline. Note that the transformer-based multi-task baselines in Tab. 2 achieve strong results. Our MQTransformers achieve consistent gains for different backbone for each tasks. In particular, we observe 2%-3% gains on depth prediction and semantic segmentation, 1%-2% gains on normal prediction and boundary prediction within an extra 3% GFlops increase. Moreover, adopting the same backbone and same pre-training, our method consistently outperforms recent work ATRC Bruggemann et al. (2021) on HRNet18 with fewer parameters and GFlops. ATRC distills on the largest scale and task interactions do not appear to be sufficient. This implies that our cross-task and cross-scale queries design can adequately capture more dependencies between multiple tasks. Finally, adopting the Swin-S backbone, our method achieves better results for semantic segmentation and comparable results on normals and boundary prediction. However, the depth prediction on NYUD-v2 is not perfect using Swin, we argue that this is because of the limited dataset size. We believe adding more depth data may lead to better results where similar findings are in previous works Dosovitskiy et al. (2021); Ranftl et al. (2021, 2020).

Results on PASCAL-Context. As shown in Tab. 3, we also carry out experiments on PASCAL-Context dataset. Again, compared with the strong baseline, our

Table 3: Results on the PASCAL-Context dataset. We also report comparisons on various baseline models (in the first and second sub-figure) and several recent works. The notation '↓': lower is better. The notation '↑': higher is better.

Model	Backbone	SemSeg (IoU)↑	PartSeg (IoU)↑	Sal (maxF)↑	Normals (mErr)↓	Bound (odsF)↑
multi-task baseline	HRNet18	51.48	57.23	83.43	14.10	69.76
multi-task baseline	ViTAEv2-S	64.66	56.65	83.89	13.73	70.70
multi-task baseline	Swin-T	64.74	53.25	76.88	15.86	69.00
multi-task baseline	Swin-S	68.10	56.20	80.64	16.09	70.20
MTI-Net Vandenhende et al. (2020)	HRNet18	61.70	60.18	84.78	14.23	70.80
PAD-Net Xu et al. (2018)	HRNet18	53.6	59.6	65.8	15.3	72.50
ATRC Bruggemann et al. (2021)	HRNet18	57.89	57.33	83.77	13.99	69.74
ASPP Chen et al. (2018)	ResNet50	62.70	59.98	83.81	14.34	71.28
BMTAS Bruggemann et al. (2020)	ResNet50	56.37	62.54	79.91	14.60	72.83
DYMU Raychaudhuri et al. (2022)	MobileNetV2	63.60	59.41	64.94	-	-
ATRC Bruggemann et al. (2021)	ResNet50	62.99	59.79	82.25	14.67	71.20
MQTransformer	HRNet18	58.91	57.43	83.78	14.17	69.80
MQTransformer	ViTAEv2-S	69.10	56.23	83.51	14.93	71.30
MQTransformer	Swin-T	68.24	57.05	83.40	14.56	71.10
MQTransformer	Swin-S	71.25	60.11	84.05	14.74	71.80

Table 4: Ablation studies and analysis on NYUD-v2 dataset using a Swin-S backbone. Query learning (QL) and Cross Task Attention Module (CTAM) are part of our model. Q&C indicates QL and CTAM. HR48 denotes HRNet48. The notation '↓': lower is better. The notation '↑': higher is better. The **w/o** indicates **"without"**.

(a) Ablation on cross task attention module

Model	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
w/o QL	48.97	0.5807	20.82	76.1
w/o CTAM	48.93	0.5824	20.80	75.6
w/o Q&C	48.64	0.5854	20.74	75.7
our	49.18	0.5785	20.8	77.0

(b) Ablation on the depths (D) of our MQTransformer

D	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
1	49.18	0.5785	20.80	77.0
2	47.80	0.6006	21.08	76.5
4	47.88	0.5983	21.21	76.5

(c) Ablation on N. Query: $P \in \mathbb{R}^{N \times C}$

N	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
8	48.32	0.5974	20.90	76.9
32	49.11	0.5803	20.58	77.0
64	49.18	0.5785	20.80	77.0
128	49.63	0.5820	20.84	77.1
156	48.81	0.5941	20.87	76.8
256	48.41	0.6014	21.01	76.7

(d) Ablation on backbones

Backbone	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
HR48	41.96	0.5543	20.36	77.62
HR48,ours	43.33	0.5511	20.09	77.60
Swin-S	47.90	0.6053	21.17	76.90
Swin-S,our	49.18	0.5785	20.80	77.00
Swin-B	51.44	0.5813	20.44	77.00
Swin-B,our	52.06	0.5439	19.85	77.80

Table 5: Ablation on shared encoder and decoder using different backbones on NYUD-v2.

Model	Backbone	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
w/o shared	HRNet18	39.05	0.5985	20.41	76.50
w/o shared	Swin-S	48.63	0.5873	20.89	77.00
Our	HRNet18	40.47	0.5965	20.34	76.60
Our	Swin-S	49.18	0.5785	20.80	77.00

method achieve consistent gains over five different tasks. Moreover, our MQTransformer achieves state-of-the-art results and outperforms previous works by a significant margin. It can be shown by Tab. 3 that the proposed method achieves an improved prediction accuracy for different tasks on different backbones.

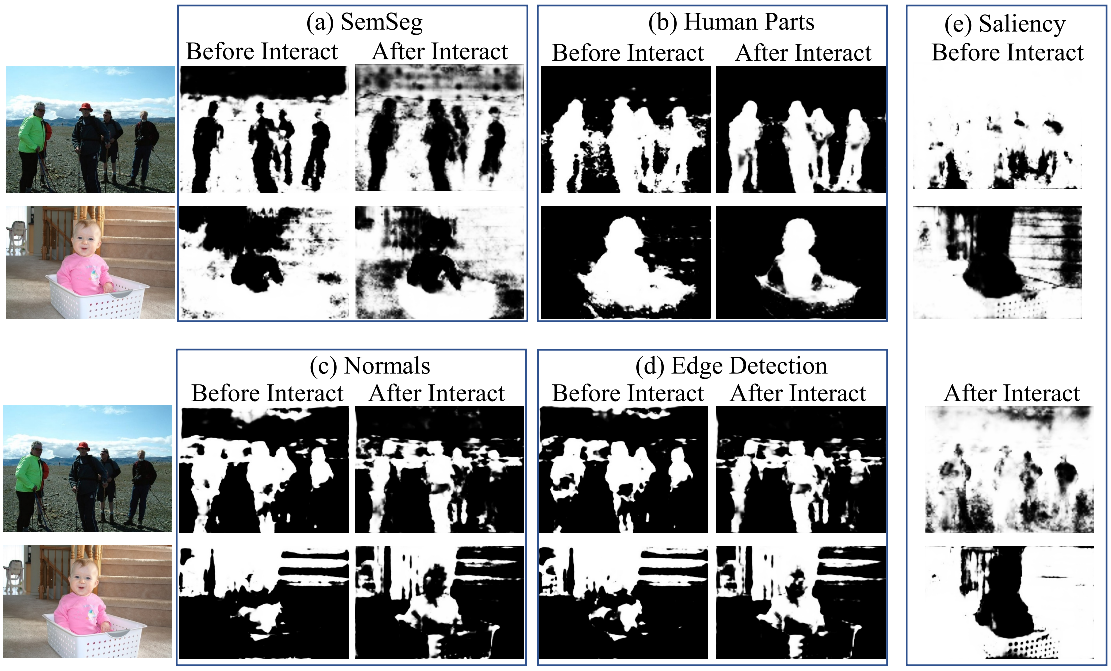


Fig. 5: Visualization activation maps on PASCAL-Context dataset. Our model can result in the correct prediction on semantic segmentation, human parts segmentation, saliency estimation, surface normal prediction, and bound (edge) detection tasks.

Table 6: Ablation on different number of scales using Swin-S backbone on NYUD-v2.

Model	SemSeg (IoU)↑	Depth (rmse)↓	Normals (mErr)↓	Bound (odsF)↑
Single-Scale	48.92	0.5839	20.88	75.7
Two-Scale	49.18	0.5785	20.8	77.0
Four-Scale	49.38	0.6006	20.99	76.9

4.3 Ablation studies and Analysis

Settings. For ablations, all models are trained on Swin-S with batch size 8 and iteration 40k unless a specific statement. For visual analysis, we adopt a trained model with Swin-S.

Network components. In Tab. 4a, simply removing the Query Learning (QL) and Cross Task Attention Module (CTAM) only on the shared encoder-decoder produces a significant degradation in performance. Despite only using Single Scale Features (SSF), we can still achieve competitive results for each task. This indicates that query learning and cross task attention benefit both segmentation and depth estimation. However, removing both QL and CTAM leads to all sub-tasks. This indicates the effectiveness of our framework.

Depth of MQTransformer. In Tab. 4b, as the number of the encoder and decoder depth increases, the accuracy does not seem to improve and even tends to decay. This means only adding one encoder and decoder is enough. However, we believe using more data can achieve better results with the increase of depth which is observed in Dosovitskiy et al. (2021); Ranftl et al. (2021). We set the depth to 1 and thus our model is lightweight and efficient.

Varying number of query N. In Tab. 4c, we change the N within $N \in \{8, 32, 64, 128, 156, 256\}$. Compared to N=8, N=64 results in a significant improvement of 2.71% in segmentation accuracy. However this improvement saturates when more sections are added to the network at N=128. Notably, larger N is not always better, when N=256, it makes an explicit decrease in accuracy for each task. Considering all the metrics, we chose N=64 by default.

Influences of different backbones. Tab. 4d presents the effect of the SemSeg, Depth, Normals, and Bound values when adopting the individual backbones. Shown in that table, swin backbones have a high SemSeg value. Notably, our model applies to both CNN Sun et al. (2019) and vision Transformer Liu et al. (2021) backbones and improves accuracy on each task, indicating the generation ability of our approach. For these different backbones, our method improves the results of different task consistently.

Effect of Shared Encoder and Decoder. In Tab. 5, we explore the shared encoder and decoder design where we find that using *a shared encoder and decoder* in MQ-Transformer leads to better results in different settings

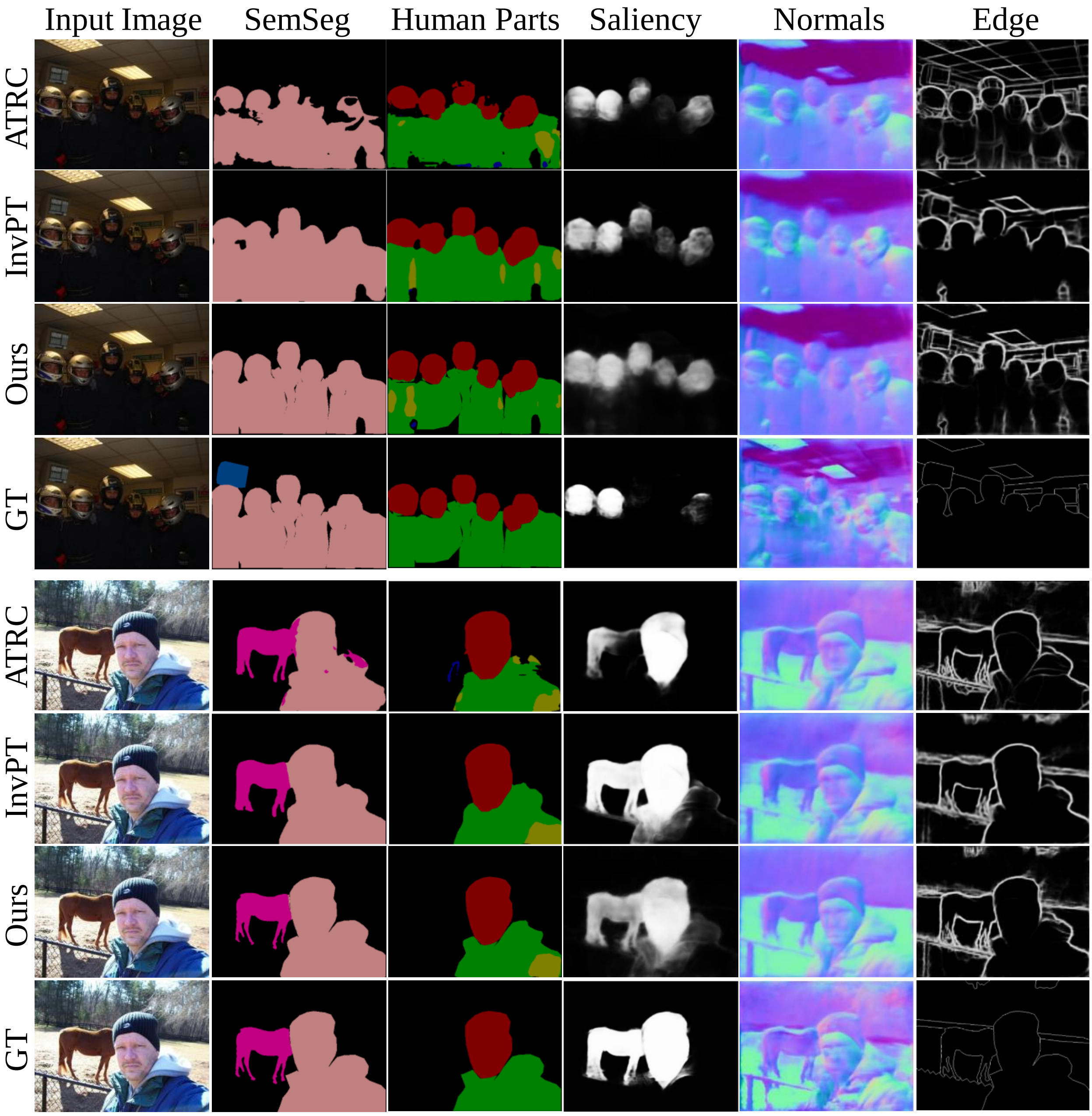


Fig. 6: Visual comparison results with ATRC Bruggemann et al. (2021) and InvPT Ye and Xu (2022) model predictions on PASCAL-Context. Our model results in the **better** prediction on semantic segmentation, human parts segmentation, saliency estimation, surface normal prediction, and edge detection.

with less parameter increase. This verifies our motivation and key design in Sec. 3.

Effect of Scale number in Query Design. In Tab. 6, we find that two scales are good enough. Compared with a single scale, our design obtains significant gains over four tasks. Adding more scales does not bring extra gains. Thus we set the scale number to 2 by default.

Visualization on Learned Query Activation Maps. In Fig. 5, we visualize the attention map for each query in each task. We randomly choose one query for visualization. As shown in that figure, after the cross task interaction, we found more structures and fine-grained results for each task which prove the benefits of our query-based cross task attention design.

Visualization Comparison. We also present several visual comparisons with recent works on PASCAL-Context dataset in Fig. 6. Compared with recent work Bruggemann et al. (2021); Ye and Xu (2022), our method has better visual results for all five tasks. The advantage of our method is that the semantic segmentation and

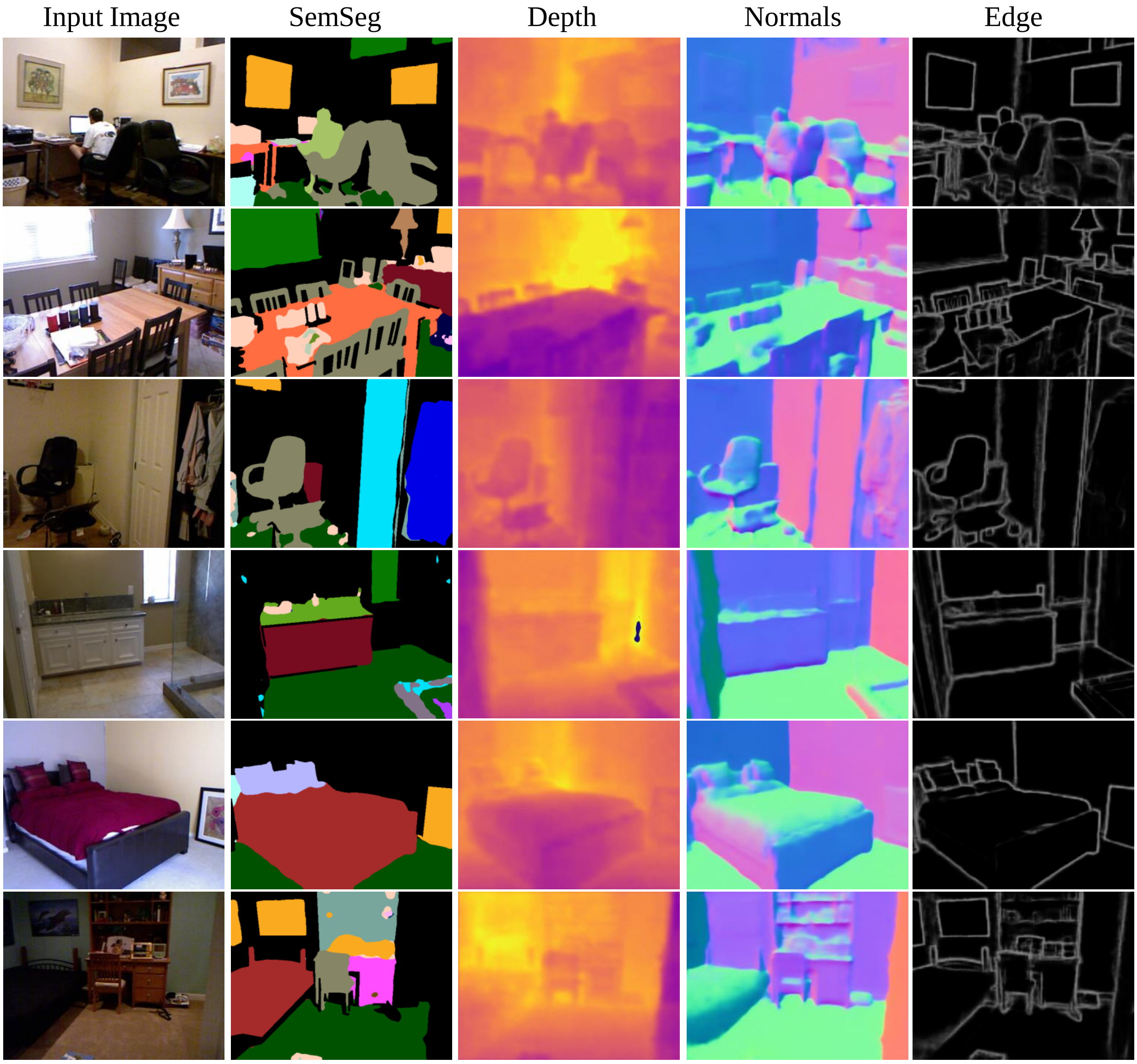


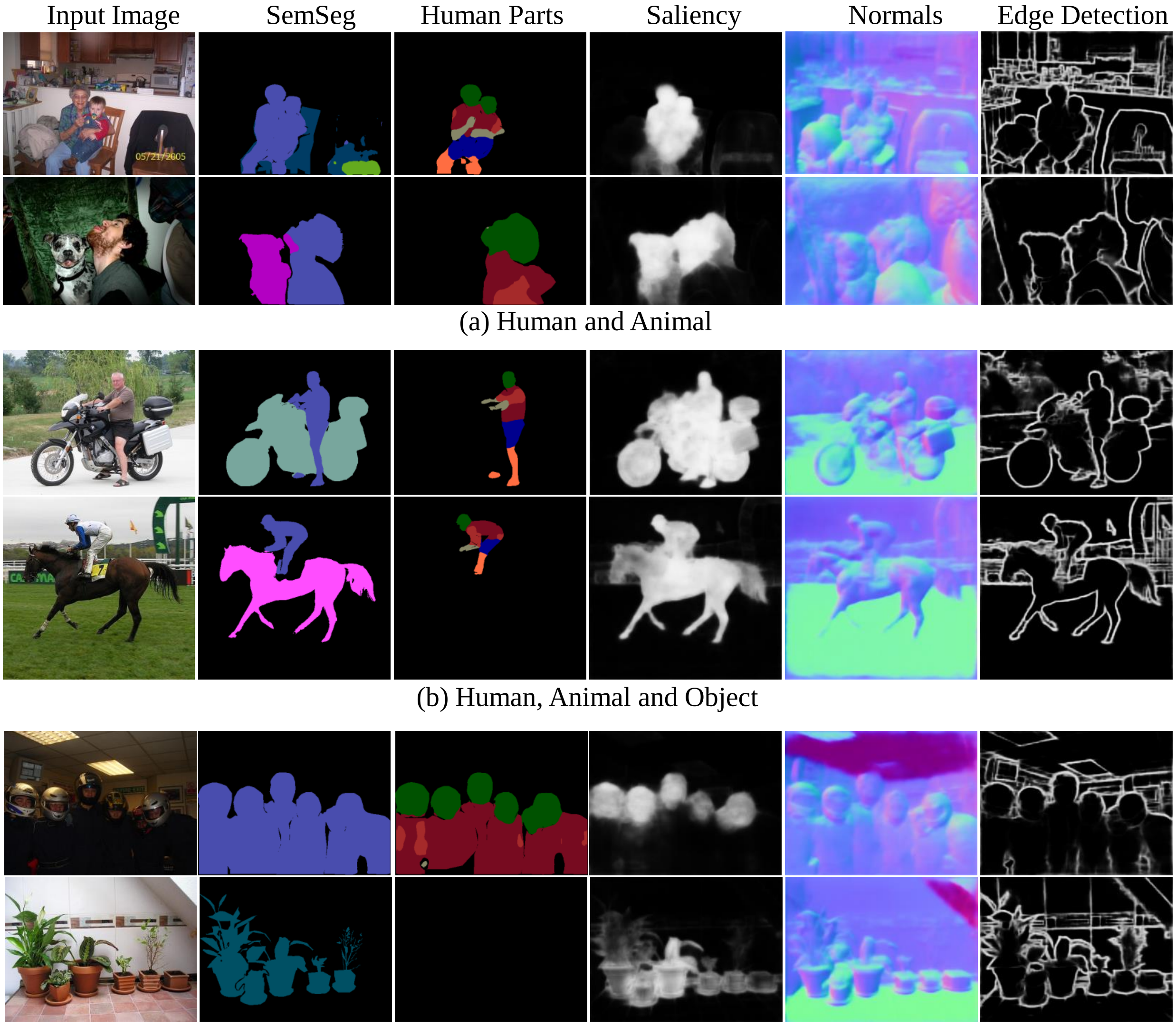
Fig. 7: Visualization results on NYUD-v2 dataset. Our model can result in the correct prediction on semantic segmentation, depth estimation, surface normal prediction, and edge detection tasks.

human parts segmentation tasks show more reliable segmentation. Fig. 6 shows some exemplars of prediction results and it suggests that our model makes effective use of a query-based transformer. Fig. 6 and Tab. 3 indicate that our MQTransformer could work finely.

Visualization experiments on NYUD-v2. For a more vivid understanding of our model, Fig. 7 shows images of the qualitative results of our MQTransformer with a Swin-S backbone on the NYUD-v2 dataset. Our method also achieves strong visualization results.

Visualization experiments on PASCAL-Context. As shown in Fig. 8, we group the images into three groups for a stronger visual contrast. Fig. 8 (a) shows two sets of images of a woman holding a child and a man holding a dog. On the human parts segmentation task, our model segments the two people in the first set of images, while the second set of images only segments the human part, ignoring the dog. Fig. 8 (b) also verifies the accuracy of the human parts segmentation task. In the last row of images (see Fig. 8 (c)), it is worth noting that the third image is all black since this one is all plants with no human parts. Moreover, in other tasks, qualitative results also demonstrate good visual performance.

Limitation and Future Work. We demonstrate the advantages of our proposed MQTransformer, however, one limitation of our method is that our task-relevant query features are randomly initialized. As shown in



(a) Human and Animal

(b) Human, Animal and Object

(c) Human and Plant

Fig. 8: Qualitative results on PASCAL-Context. We group the visualization results into three groups for comparison. (a) Human and animal; (b) Human, animal and object; (c) Human and plant Our model is able to find the correct image feature corresponding to different tasks and eventually get the correct prediction on semantic segmentation, human parts segmentation, saliency estimation, surface normals prediction, and edge (bound) detection tasks.

Tab. 4c, we show the effect of the N of the task-relevant query feature ($P \in \mathbb{R}^{N \times C}$) on the model. We first generate task-relevant query features using image information of different tasks. Thus, more relevant and fine details can be obtained. In the future, we will be exploring the task-relevant query features and the potential of the Transformer architecture in MTDP.

5 Conclusion

In this paper, we propose a new vision transformer architecture for MTDP. We design a novel task-relevant and scale-aware query to extract features from different tasks and scales. Then we perform task association via query learning which avoids huge pixel-level computation and cost that are used in previous works. Then a shared encoder and decoder network is adopted to exchange information between queries and corresponding task-aware features. Extensive experiments show that our model can achieve significant improvements on different metrics with various strong baselines. Moreover, our method achieves state-of-the-art results on NYUD-v2 and PASCAL-Context datasets. We hope our method can be a new simple yet effective transformer baseline for MTDP.

Boarder Impact. Our method explores the multi-task dense prediction with a novel multi-query transformer architecture. It is a new encoder-decoder baseline for this task and may inspire the new design of the multi-task learning framework.

6 Acknowledgement

This work was supported by the National Key Research and Development Program of China (No.2020YFB2103402).

Data availability

The datasets generated during and/or analysed during the current study are available in the NYUD-v2 and PASCAL-Context repositories, https://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html and https://www.cs.stanford.edu/~roozbeh/pascal-context/

References

Bruggemann D, Kanakis M, Georgoulis S, Van Gool L (2020) Automated search for resource-efficient branched multi-task networks. arXiv preprint arXiv:200810292

Bruggemann D, Kanakis M, Obukhov A, Georgoulis S, Gool LV (2021) Exploring relational context for multi-task dense prediction. ICCV

Bumsoo K, Junhyun L, Jaewoo K, Eun-Sol K, J KH (2021) Hotr: End-to-end human-object interaction detection with transformers. CVPR

Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: ECCV

Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H (2018) Encoder-decoder with atrous separable convolution for semantic image segmentation. In: ECCV

Chen X, Mottaghi R, Liu X, Fidler S, Urtasun R, Yuille A (2014) Detect what you can: Detecting and representing objects using holistic models and body parts. In: CVPR

Ding L, Lin D, Lin S, Zhang J, Cui X, Wang Y, Tang H, Bruzzone L (2021) Looking outside the window: Wide-context transformer for the semantic segmentation of high-resolution remote sensing images. arXiv

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR

Gao Y, Ma J, Zhao M, Liu W, Yuille AL (2019) Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In: CVPR

Georgescu MI, Barbalau A, Ionescu RT, Khan FS, Popescu M, Shah M (2021) Anomaly detection in video via self-supervised and multi-task learning. CVPR

Ghiasi G, Zoph B, Cubuk ED, Le QV, Lin TY (2021) Multi-task self-training for learning general representations. In: ICCV

Guo MH, Xu TX, Liu JJ, Liu ZN, Jiang PT, Mu TJ, Zhang SH, Martin RR, Cheng MM, Hu SM (2021) Attention mechanisms in computer vision: A survey. arXiv preprint arXiv:211107624

Hehe F, Yi Y, Mohan K (2021) Point 4d transformer networks for spatio-temporal modeling in point cloud videos. CVPR

Hu R, Singh A (2021) Unit: Multimodal multitask learning with a unified transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1439–1449

Huang S, Lu Z, Cheng R, He C (2021) Fapn: Feature-aligned pyramid network for dense image prediction. In: ICCV

Hugo T, Matthieu C, Matthijs D, Francisco M, Alexandre S, Jégou H (2021) Training data-efficient image transformers & distillation through attention. ICML

Jack L, Tianlu W, Vicente O, Yanjun Q (2021) General multi-label image classification with transformers. CVPR

Jalali A, Sanghavi S, Ruan C, Ravikumar P (2010) A dirty model for multi-task learning. NeurIPS

Kanakis M, Bruggemann D, Saha S, Georgoulis S, Obukhov A, Van Gool L (2020) Reparameterizing convolutions for incremental multi-task learning without task interference. In: ECCV

Kendall A, Gal Y, Cipolla R (2018) Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: CVPR

Kundu JN, Lakkakula N, Babu RV (2019) Um-adapt: Unsupervised multi-task adaptation using adversarial cross-task distillation. In: ICCV

Li K, Wang S, Zhang X, Xu Y, Xu W, Tu Z (2021) Pose recognition with cascade transformers. CVPR

Li X, You A, Zhu Z, Zhao H, Yang M, Yang K, Tan S, Tong Y (2020) Semantic flow for fast and accurate scene parsing. In: ECCV

Li X, Xu S, Cheng YY, Tong Y, Tao D, et al. (2022a) Panoptic-partformer: Learning a unified model for panoptic part segmentation. arXiv preprint arXiv:220404655

Li X, Zhang W, Pang J, Chen K, Cheng G, Tong Y, Loy CC (2022b) Video k-net: A simple, strong, and unified baseline for video segmentation. In: CVPR

Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: CVPR

Ling Z, Zhen C, Chunyan X, Zhenyu Z, Chaoqun W, Tong Z, Jian Y (2020) Pattern-structure diffusion for multi-task learning. CVPR

Liu S, Johns E, Davison AJ (2019) End-to-end multi-task learning with attention. In: CVPR

Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: Hierarchical vision transformer using shifted windows. ICCV

Misra I, Shrivastava A, Gupta A, Hebert M (2016) Cross-stitch networks for multi-task learning. In: CVPR

Phillips J, Martinez J, Bârsan IA, Casas S, Sadat A, Urtasun R (2021) Deep multi-task learning for joint localization, perception, and prediction. In: CVPR

Prakash A, Chitta K, Geiger A (2021) Multi-modal fusion transformer for end-to-end autonomous driving. In: CVPR

Ranftl R, Lasinger K, Hafner D, Schindler K, Koltun V (2020) Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. IEEE TPAMI

Ranftl R, Bochkovskiy A, Koltun V (2021) Vision transformers for dense prediction. ArXiv preprint

Raychaudhuri DS, Suh Y, Schulter S, Yu X, Faraki M, Roy-Chowdhury AK, Chandraker M (2022) Controllable dynamic multi-task architectures. In: CVPR, pp 10955–10964

Shu C, Liu Y, Gao J, Yan Z, Shen C (2021) Channel-wise knowledge distillation for dense prediction. ICCV

Silberman N, Hoiem D, Kohli P, Fergus R (2012) Indoor segmentation and support inference from rgbd images. In: ECCV

Strezoski G, Noord Nv, Worring M (2019) Many task learning with task routing. ICCV

Sun K, Xiao B, Liu D, Wang J (2019) Deep high-resolution representation learning for human pose estimation. In: CVPR

Sun X, Panda R, Feris R, Saenko K (2020) Adashare: Learning what to share for efficient deep multi-task learning. NeurIPS

Takahashi N, Mitsufuji Y (2021) Densely connected multi-dilated convolutional networks for dense prediction tasks. In: CVPR

Tateno K, Navab N, Tombari F (2018) Distortion-aware convolutional filters for dense prediction in panoramic images. ECCV

Vandenhende S, Georgoulis S, Van Gool L (2020) Mti-net: Multi-scale task interaction networks for multi-task learning. In: ECCV

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. NIPS

Wang W, Xie E, Li X, Fan DP, Song K, Liang D, Lu T, Luo P, Shao L (2021a) Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. ICCV

Wang Y, Xu Z, Wang X, Shen C, Cheng B, Shen H, Xia H (2021b) End-to-end video instance segmentation with transformers. CVPR

Xin C, Bin Y, Jiawen Z, Dong W, Xiaoyun Y, Lu H (2021) Transformer tracking. CVPR

Xu D, Ouyang W, Wang X, Sebe N (2018) Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In: CVPR

Xu S, Li X, Wang J, Cheng G, Tong Y, Tao D (2022) Fashionformer: A simple, effective and unified baseline for human fashion segmentation and recognition. In: arxiv

Xu Y, Zhang Q, Zhang J, Tao D (2021) Vitae: Vision transformer advanced by exploring intrinsic inductive bias. NeurIPS 34:28522–28535

Yang Y, Li H, Li X, Zhao Q, Wu J, Lin Z (2020) Sognet: Scene overlap graph network for panoptic segmentation. In: AAAI

Yang Y, You S, Li H, Wang F, Qian C, Lin Z (2021) Towards improving the consistency, efficiency, and flexibility of differentiable neural architecture search. In: CVPR

Ye H, Xu D (2022) Inverted pyramid multi-task transformer for dense scene understanding. arXiv preprint arXiv:220307997

Yuan H, Li X, Yang Y, Cheng G, Zhang J, Tong Y, Zhang L, Tao D (2021a) Polyphonicformer: Unified query learning for depth-aware video panoptic segmentation. arXiv preprint arXiv:211202582

Yuan L, Chen Y, Wang T, Yu W, Shi Y, Jiang Z, Tay FE, Feng J, Yan S (2021b) Tokens-to-token vit: Training vision transformers from scratch on imagenet. arXiv preprint arXiv:210111986

Zhang Q, Xu Y, Zhang J, Tao D (2022a) Vitaev2: Vision transformer advanced by exploring inductive bias for image recognition and beyond. arXiv preprint arXiv:220210108

Zhang Q, Xu Y, Zhang J, Tao D (2022b) Vsa: Learning varied-size window attention in vision transformers. arXiv preprint arXiv:220408446

Zhang Z, Cui Z, Xu C, Yan Y, Sebe N, Yang J (2019) Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In: CVPR

Zhenyu Z, Zhen C, Chunyan X, Zequn J, Xiang L, Jian Y (2018) Joint task-recursive learning for semantic segmentation and depth estimation. ECCV

Zhu X, Su W, Lu L, Li B, Wang X, Dai J (2021) Deformable DETR: deformable transformers for end-to-end object detection. ICLR