AGGREGATE, GATE, AND GUIDE: TOWARDS EFFICIENT, BALANCED, AND SHARP MULTI-TASK DENSE SCENE UNDERSTANDING

Anonymous authors
Paper under double-blind review

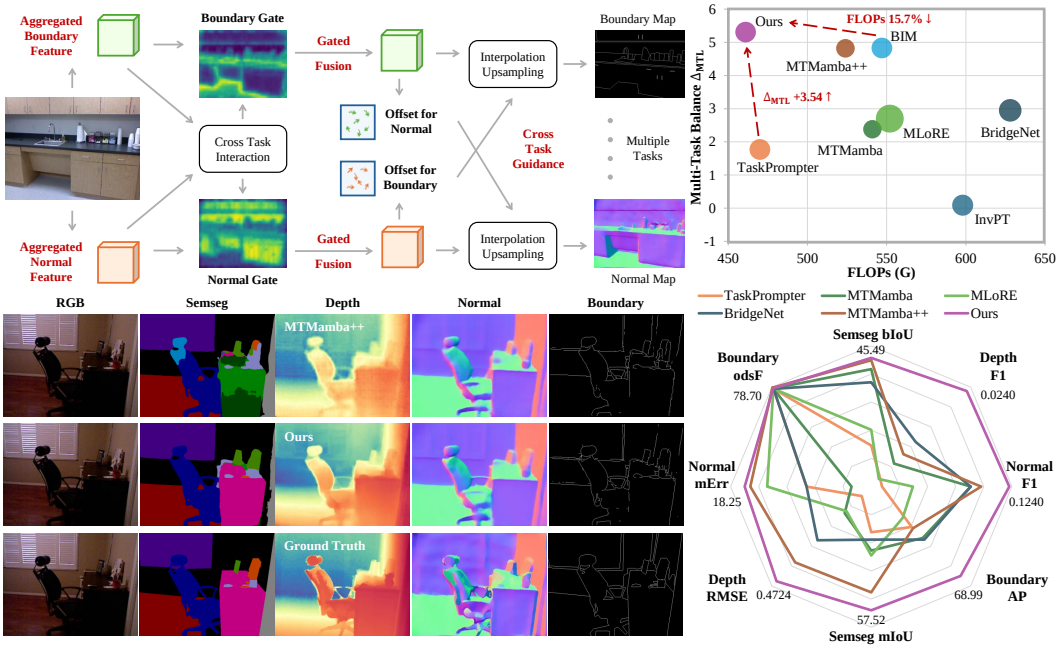

Figure 1: We present **AGG-MTL**, which advances the efficiency–balance–sharpness trade-off in multi-task dense prediction through three designs: (1) aggregate-then-interact, which reduces repeated cross-task interaction across scales; (2) adaptive cross-task interaction transformer, which preserves critical task-specific patterns via soft-to-hard gating; and (3) multi-task guided upsampling, which exploits complementary cues across tasks for sharp predictions with fine structures.

ABSTRACT

Multi-task dense scene understanding jointly solves multiple pixel-level tasks with a shared image encoder, aiming to exploit task correlations while maintaining computational efficiency. However, existing methods often perform cross-task interactions at multiple feature scales, introducing considerable computational overhead. Moreover, they typically exchange task information without explicitly preserving task-specific patterns, which may compromise balanced performance. Meanwhile, their upsampling operations are commonly performed independently for each task, overlooking complementary task cues that can facilitate the recovery of fine structures. To address these limitations, we propose AGG-MTL, a multi-task dense prediction framework that advances the efficiency–balance–sharpness trade-off. First, we aggregate multi-scale task features into compact representations and perform cross-task interaction only at a single scale, reducing repeated computation across resolutions. Second, an adaptive cross-task interaction transformer employs soft-to-hard gating to selectively incorporate cross-task cues while preserving critical task-specific patterns. Third, a multi-task guided upsampling module predicts task-specific interpolation offsets from complementary multi-task features, extending cross-task collaboration to fine-structure recovery.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Experiments on NYUDv2 and PASCAL-Context demonstrate a favorable trade-off among computational efficiency, balanced multi-task performance, and prediction sharpness. On NYUDv2, AGG-MTL improves Δ_{MTL} by $+3.54$ points over the most efficient prior method, reduces FLOPs by 15.7% relative to the previous best-balanced method, and achieves the best sharpness scores across all tasks.

1 INTRODUCTION

Multi-task dense scene understanding (Vandenhende et al., 2022), also referred to as multi-task dense prediction, is a fundamental yet challenging problem in computer vision. It aims to simultaneously solve multiple pixel-level tasks (*e.g.*, semantic segmentation, depth estimation, and surface normal estimation) (Cao et al., 2025; Zhang et al., 2025) using a shared backbone (Liu et al., 2021; Dosovitskiy et al., 2021). The core motivation behind this paradigm lies in improving computational efficiency while exploiting task correlations to achieve superior performance (Vandenhende et al., 2022). Such merits underpin its widespread applications, including autonomous driving (Ishihara et al., 2021), robotics (Ze et al., 2023), and healthcare (Hur et al., 2024).

Existing multi-task dense prediction methods often perform cross-task interaction at multiple feature scales (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025). However, repeatedly applying cross-task interaction modules at different resolutions introduces considerable computational overhead. This motivates us to rethink *where cross-task interaction should be performed*. Instead of fusing task features at every scale, a more efficient strategy is to aggregate multi-scale task features into a compact representation and conduct cross-task interaction only at a single scale. Such an aggregate-then-interact design can reduce computational cost while maintaining effective task collaboration.

Beyond deciding where tasks should interact, another important question is *what information should be exchanged*. Existing methods develop various mechanisms for propagating information among tasks (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025). However, they primarily focus on how to perform interaction, leaving it implicit which specific components within task features should be fused. For example, InvPT (Ye & Xu, 2022) and BIM (Cao et al., 2025) enable global interaction across both positions and tasks via attention and Mamba, respectively. However, the usefulness of cross-task cues is content-dependent: high-frequency structural cues from boundary detection can help estimate surface normals near object contours, yet may interfere with normal-specific patterns in textureless regions. Therefore, a desirable multi-task decoder should not simply fuse all task features without distinction. Instead, it should adaptively determine which components should be updated through cross-task interaction and which components should be directly preserved as task-specific patterns, thereby exploiting task correlations while mitigating cross-task interference.

Cross-task collaboration is valuable not only for feature interaction, but also for high-resolution feature recovery. Existing methods typically perform upsampling independently for each task, often using fixed operators such as bilinear interpolation (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025). This task-wise reconstruction overlooks the fact that the recovery of fine structures is itself correlated across tasks. For instance, object boundaries can help sharpen semantic delineations, semantic regions can provide layout priors for depth estimation, and human parsing features can help preserve body structures in saliency prediction. These complementary cues are particularly valuable for restoring spatial details that low-resolution representations tend to obscure. Upsampling should therefore be formulated as a multi-task collaborative process, in which the reconstruction of each task is guided by complementary information from the others, enabling sharp dense predictions.

To this end, we propose AGG-MTL (Aggregate, Gate, and Guide for Multi-Task Learning), a framework for efficient multi-task dense scene understanding with balanced performance and sharp predictions, as shown in Figure 1. We first aggregate multi-scale task features into compact representations and perform cross-task interaction only at a single scale, avoiding repeated computation across resolutions. Second, we introduce an Adaptive Cross-task Interaction Transformer (ACIT) with a soft-to-hard gating strategy. During training, soft gates enable flexible information exchange and encourage the model to explore rich cross-task correlations. During inference, the gates are binarized to perform hard selection, explicitly preserving critical task-specific patterns while filtering out noisy cross-task interactions. Finally, we propose a Multi-task Guided Upsampling Module (MGUM), which predicts task-specific interpolation offsets from multi-task features, allowing complementary cross-task cues to guide upsampling and faithfully recover fine structures.

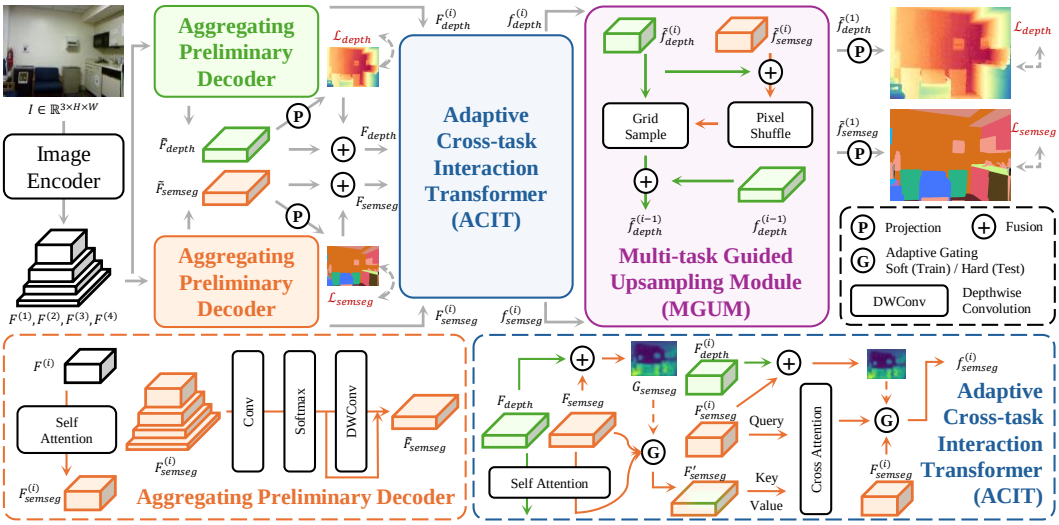

Figure 2: **Pipeline of AGG-MTL.** Only two tasks (depth estimation and semantic segmentation) are illustrated for simplicity. We first consolidate multi-scale features into compact representations F_{depth} and F_{semseg}, where ACIT performs cross-task interactions while preserving task-specific patterns. MGUM predicts task-specific offsets from multi-task features to recover sharp predictions.

We conduct comprehensive evaluations on two standard benchmarks, NYUDv2 (Silberman et al., 2012) and PASCAL-Context (Chen et al., 2014). We further evaluate prediction sharpness under a protocol tailored to various tasks. Specifically, on NYUDv2, AGG-MTL improves Δ_{MTL} by $+3.54$ points over the most computationally efficient prior method, while reducing FLOPs by 15.7% relative to the previous best method in overall multi-task balance. AGG-MTL further achieves the best sharpness scores among various methods across all tasks, recovering fine structures more faithfully and producing predictions that adhere more closely to the ground truth, as illustrated in Figure 1.

Our main contributions are summarized as follows:

- We introduce a multi-task dense prediction model that advances the efficiency–balance–sharpness trade-off. Its aggregate-then-interact design consolidates multi-scale task features before cross-task interaction, reducing computational cost.
- We develop an adaptive cross-task interaction transformer with soft-to-hard gating. It selectively determines which feature components should incorporate cross-task context and which should be preserved as critical task-specific patterns.
- We propose a multi-task guided upsampling module that predicts task-specific interpolation offsets from complementary multi-task features, extending cross-task collaboration to spatial detail recovery for sharp predictions with fine structures.

2 AGG-MTL

We first introduce the overall framework in Section 2.1. We then detail each component, including the image encoder (Section 2.2), the aggregating preliminary decoder (Section 2.3), the proposed ACIT (Section 2.4), the designed MGUM (Section 2.5), and the output head (Section 2.6).

2.1 Overview

Given a single image I and a set of tasks $\mathcal{T}$, our goal is to produce a dense prediction for each task $t \in \mathcal{T}$. As illustrated in Figure 2, AGG-MTL first aggregates multi-scale task features into compact representations F_t to reduce the computational cost of cross-task interaction. ACIT then performs selective interactions to incorporate complementary cross-task cues while preserving critical task-specific patterns. Finally, MGUM uses features from multiple tasks to guide the upsampling of each target task, producing sharp predictions that faithfully preserve fine structures.

First, after obtaining multi-scale features $\{F^{(i)}\}_{i=1}^{4}$ from the task-shared image encoder, we aggregate them into compact task representations $\{F_t\}_{t\in\mathcal{T}}$. Unlike previous methods (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025) that perform cross-task interaction at multiple scales, AGG-MTL performs it only at the aggregated scale. This design reduces computational cost while maintaining balanced task performance and prediction sharpness with the proposed ACIT and MGUM.

Second, the Adaptive Cross-task Interaction Transformer (ACIT) enables selective information exchange among $\{F_t\}_{t\in\mathcal{T}}$. Rather than implicitly fusing all task features (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025), ACIT employs a soft-to-hard gating strategy to explicitly determine which components should incorporate cross-task information and which should preserve task-specific patterns. Soft gates encourage the exploration of rich task correlations during training, while hard gates preserve critical task-specific patterns during inference.

Finally, the Multi-task Guided Upsampling Module (MGUM) recovers high-resolution task features through multi-task guidance. Instead of relying on bilinear interpolation (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025), MGUM predicts task-specific sampling offsets from multi-task features. In this way, complementary cues across tasks, such as geometric discontinuities that help localize semantic transitions, can guide the upsampling process and facilitate fine-structure recovery.

2.2 Image Encoder

Following previous works (Cao et al., 2025; Lin et al., 2025; 2024), we employ the Swin Transformer (Liu et al., 2021) as our image encoder. Given an input image $I \in \mathbb{R}^{3\times H\times W}$, where H and W denote the height and width, respectively, the encoder partitions it into non-overlapping patches and progressively processes the resulting tokens through Swin Transformer blocks and patch-merging layers. This hierarchical encoding process produces multi-scale image features:

$$\left\{F^{(i)}\right\}_{i=1}^{4} = \mathrm{ImageEncoder}(I), \tag{1}$$

where $F^{(i)} \in \mathbb{R}^{2^{i-1}C\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}$ and C denotes the channel dimension at the first stage.

2.3 Aggregating Preliminary Decoder

As motivated in Section 1, repeatedly performing cross-task feature interaction at multiple resolutions incurs considerable computational overhead. We therefore adopt an aggregate-then-interact strategy: multi-scale features are first consolidated into compact task representations, after which explicit cross-task feature interaction is performed only at the aggregated scale.

Given the multi-scale image features $F^{(i)}$, we first employ self-attention to obtain multi-scale task-specific features $F_t^{(i)}$. For each task t, we then project $\{F_t^{(i)}\}_{i=1}^{4}$ into a common spatial and channel space and aggregate them using learned weights, ultimately producing a compact task representation $F_t \in \mathbb{R}^{c\times h\times w}$. Specifically, the aggregation process can be formulated as:

$$\alpha_t^{(i)} = \frac{\exp(w_t^{(i)})}{\sum_{j=1}^{4}\exp(w_t^{(j)})} \in \mathbb{R}, \tag{2}$$

$$\tilde{F}_t = \sum_{i=1}^{4}\alpha_t^{(i)}\mathrm{Conv}_t^{(i)}(F_t^{(i)}) + \mathrm{DWConv}_t(\sum_{i=1}^{4}\alpha_t^{(i)}\mathrm{Conv}_t^{(i)}(F_t^{(i)})) \in \mathbb{R}^{c\times h\times w}, \tag{3}$$

$$P_t = \mathrm{PWConv}_t\left(\tilde{F}_t\right), \tag{4}$$

$$F_t = \mathrm{PWConv}_t\left(\mathrm{Concat}\left[\tilde{F}_t, P_t\right]\right) \in \mathbb{R}^{c\times h\times w}, \tag{5}$$

where $w_t^{(i)}$ is a learnable scalar, PWConv_t denotes pointwise convolution, and DWConv_t denotes depthwise convolution. $\mathrm{Conv}_t^{(i)}$ aligns $F_t^{(i)}$ with the common feature space $\mathbb{R}^{c\times h\times w}$.

In our implementation, we set $c = 192$, $h = \frac{H}{16}$, and $w = \frac{W}{16}$. The kernel sizes of $\mathrm{Conv}_t^{(i)}$ for $i \in \{1, 2, 3, 4\}$ are 4, 2, 1, and 2, respectively. $\mathrm{Conv}_t^{(4)}$ is implemented as a transposed convolution. For each $\mathrm{Conv}_t^{(i)}$, the stride equals its kernel size, and zero padding is used.

Following previous works (Ye & Xu, 2022; 2024; Zhang et al., 2025), we generate a preliminary prediction P_t from $\tilde{F}_t$ and supervise it using the same task-specific loss $\mathcal{L}_t$ as the final prediction. P_t is also concatenated with $\tilde{F}_t$ and projected to obtain the compact task representation F_t.

2.4 Adaptive Cross-task Interaction Transformer

As discussed in Section 1, existing methods (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025) primarily focus on how to exchange information across tasks, while leaving it implicit whether each feature component should incorporate cross-task cues or preserve task-specific patterns. We therefore formulate cross-task interaction as a selective update, allowing each task to adaptively incorporate complementary cross-task information while retaining critical task-specific patterns.

Given the compact task representations $\{F_t\}_{t\in\mathcal{T}}$, we first stack them:

$$F_{\text{stack}} = \text{Stack}\left(\{F_t\}_{t\in\mathcal{T}}\right) \in \mathbb{R}^{c\times|\mathcal{T}|\times h\times w}, \tag{6}$$

and then we rearrange F_{stack} into $X \in \mathbb{R}^{|\mathcal{T}|hw\times c}$ and apply self-attention:

$$Y = \text{Softmax}\left(\frac{XW^Q(XW^K)^\top}{\sqrt{c}}\right)XW^V \in \mathbb{R}^{|\mathcal{T}|hw\times c}, \tag{7}$$

where $W^Q, W^K, W^V \in \mathbb{R}^{c\times c}$ are learnable projections. We reshape Y back to the task-wise spatial representations $\{Y_t\}_{t\in\mathcal{T}}$, where $Y_t \in \mathbb{R}^{c\times h\times w}$. To determine whether each feature component should preserve its task-specific representation or incorporate the interacted representation, we construct a spatial-channel gate for each task. We first concatenate $\{F_t\}_{t\in\mathcal{T}}$:

$$F_{\text{cat}} = \text{Concat}\left(\{F_t\}_{t\in\mathcal{T}}\right) \in \mathbb{R}^{c|\mathcal{T}|\times h\times w}, \tag{8}$$

and then generate the task-specific gate as:

$$G_t = \text{Sigmoid}\left(\text{PWConv}_t(F_{\text{cat}})\right) \in [0,1]^{c\times h\times w}. \tag{9}$$

During training, we retain the continuous gate to enable smooth optimization and flexible exploration of cross-task correlations. During inference, the gate is binarized to explicitly select between task-specific patterns and cross-task representations:

$$\tilde{G}_t = \begin{cases} G_t, & \text{during training,} \\ \mathbf{1}[G_t > \delta], & \text{during inference,} \end{cases} \tag{10}$$

where δ is the gating threshold and $\mathbf{1}[\cdot]$ denotes the indicator function. The selectively interacted task representation is then given by:

$$F'_t = F_t \odot \tilde{G}_t + Y_t \odot \left(1 - \tilde{G}_t\right), \tag{11}$$

where $\odot$ denotes element-wise multiplication. A larger gate value preserves more of the task-specific pattern, whereas a smaller value introduces more information from cross-task interaction.

After completing cross-task interaction at the aggregated scale, we fuse F'_t into the multi-scale task features. As illustrated in Figure 2, we perform cross-attention using $F_t^{(i)}$ as the query and F'_t as both the key and value. We then apply a task-specific gate following the selective fusion principle in Equations (9)–(11). The resulting feature $f_t^{(i)}$ is fed into the upsampling stage.

2.5 Multi-task Guided Upsampling Module

As highlighted in Section 1, cross-task collaboration is valuable not only for feature interaction, but also for high-resolution feature recovery. However, existing methods (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025) typically upsample each task independently, overlooking complementary cues across tasks that can facilitate the recovery of fine structures. We therefore predict task-specific interpolation offsets from multi-task features to guide high-resolution reconstruction, ultimately producing sharp predictions with faithful fine structures.

Given the updated multi-scale task features $\{f_t^{(i)}\}_{i=1}^4$, let $\tilde{f}_t^{(i)}$ denote the current upsampled feature, with $\tilde{f}_t^{(4)} = f_t^{(4)}$. At each stage $i \in \{3, 2, 1\}$, we first concatenate the upsampled features:

$$f_{\text{cat}}^{(i+1)} = \text{Concat}\left(\left\{\tilde{f}_t^{(i+1)}\right\}_{t\in\mathcal{T}}\right) \in \mathbb{R}^{2^i C|\mathcal{T}|\times\frac{H}{2^{i+2}}\times\frac{W}{2^{i+2}}}. \tag{12}$$

For each task t, a pointwise convolution followed by pixel shuffle predicts an interpolation offset:

$$o_t^{(i)} = \text{PixelShuffle}\left(\text{PWConv}_t\left(f_{\text{cat}}^{(i+1)}\right)\right) \in \mathbb{R}^{2\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}. \tag{13}$$

We then deform the standard sampling grid using the predicted offsets and fuse the sampled feature with the corresponding feature at the target scale:

$$\tilde{f}_t^{(i)} = \text{PWConv}_t\left(\text{GridSample}\left(\tilde{f}_t^{(i+1)}, o_t^{(i)} + g\right)\right) + \text{Conv}_t\left(f_t^{(i)}\right) \in \mathbb{R}^{2^{i-1}C\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}, \tag{14}$$

where g denotes the standard sampling grid. Unlike fixed task-independent interpolation, complementary multi-task cues are used to adapt the sampling locations for each target task, facilitating fine-structure recovery. The resulting feature $\tilde{f}_t^{(1)}$ is fed into the task-specific output head.

2.6 Output Head

Following common practice in multi-task dense prediction (Ye & Xu, 2022; Lin et al., 2024; Cao et al., 2025), we employ a lightweight task-specific output head comprising a 3×3 convolution, batch normalization, ReLU activation, and a final linear projection. For each task, both the final prediction and the preliminary prediction P_t in Section 2.3 are supervised using the same task-specific loss function adopted in prior works (Ye & Xu, 2022; Lin et al., 2024; Cao et al., 2025).

3 Experiments

We first detail the datasets, implementation details, and evaluation metrics in Section 3.1. Then, we present comparisons with state-of-the-art methods in Section 3.2. Finally, we conduct ablation studies in Section 3.3 to validate the effectiveness of our main contributions.

3.1 Experimental Setups

Datasets. Following previous works (Ye & Xu, 2022; Lin et al., 2024; Cao et al., 2025), we evaluate on NYUDv2 and PASCAL-Context. NYUDv2 (Silberman et al., 2012) contains 795 training and 654 test images, with $\mathcal{T}_{\text{NYUDv2}} = \{\text{Semseg}, \text{Depth}, \text{Normal}, \text{Boundary}\}$. PASCAL-Context (Chen et al., 2014) contains 4998 training and 5105 test images, with $\mathcal{T}_{\text{PASCAL}-\text{Context}} = \{\text{Semseg}, \text{Parsing}, \text{Saliency}, \text{Normal}, \text{Boundary}\}$. Here, Semseg, Depth, Normal, Boundary, Parsing, and Saliency denote semantic segmentation, depth estimation, surface normal estimation, object boundary detection, human parsing, and saliency detection, respectively.

Implementation details. Following previous works (Ye & Xu, 2022; Lin et al., 2024; Cao et al., 2025), we employ Swin-Large (Liu et al., 2021) as our image encoder and adopt the same data preprocessing, task-specific losses, and loss-weighting schemes. We train the model for 50000 iterations with a batch size of 4 on NYUDv2 and 75000 iterations with a batch size of 6 on PASCAL-Context. Optimization is performed using AdamW with a learning rate of 5×10^{-5}.

Evaluation metrics. Following previous works (Ye & Xu, 2022; Lin et al., 2024; Cao et al., 2025), we evaluate Semseg and Parsing using mean intersection over union (mIoU), Depth using root mean squared error (RMSE), Normal using mean error (mErr), Boundary using the optimal dataset-scale F-measure (odsF), Saliency using the maximum F-measure (maxF), and overall task balance using the average relative multi-task performance (Δ_{MTL}) (Vandenhende et al., 2022). We report floating-point operations (FLOPs) and the number of parameters (Params) for computational cost and model size, respectively. To evaluate prediction sharpness, we report boundary intersection over union (bIoU) (Cheng et al., 2021) for Semseg, discontinuity F1 (Bochkovskiy et al., 2025) for Depth, its angular-discontinuity adaptation for Normal, and average precision (AP) (Xie & Tu, 2015) for Boundary. More details are provided in the supplementary material.

Table 1: **Comparisons of multi-task balance with state-of-the-art methods** on NYUDv2 (Silberman et al., 2012) (left) and PASCAL-Context (Chen et al., 2014) (right). On NYUDv2, Δ_{MTL} is computed relative to the standard set of independently trained single-task models adopted in prior works (Lin et al., 2024; 2025; Cao et al., 2025), whose Semseg, Depth, Normal, and Boundary scores are 54.32, 0.5166, 19.21, and 77.30, respectively. On PASCAL-Context, since prior works (Lin et al., 2024; 2025; Cao et al., 2025) do not establish a common reference baseline for computing Δ_{MTL}, we report scores relative to AGG-MTL for ease of comparison.

Method	Semseg	Depth	Normal	Boundary	Balance
	mIoU↑	RMSE↓	mErr↓	odsF↑	Δ_{MTL} ↑
Cross-Stitch (Misra et al., 2016)	36.34	0.6290	20.88	76.38	-16.19
PAP (Zhang et al., 2019)	36.72	0.6178	20.82	76.42	-15.38
PSD (Zhou et al., 2020)	36.69	0.6246	20.87	76.42	-15.79
PAD-Net (Xu et al., 2018)	36.61	0.6270	20.85	76.38	-15.93
MTI-Net (Vandenhende et al., 2020)	45.94	0.5365	20.27	77.86	-6.02
ATRC (Brüggemann et al., 2021)	46.33	0.5363	20.18	77.94	-5.69
MQTransformer (Xu et al., 2024)	54.84	0.5325	19.67	78.20	-0.84
InvPT (Ye & Xu, 2022)	53.66	0.5183	19.04	78.10	+0.09
TaskPrompter (Ye & Xu, 2023)	55.30	0.5152	18.47	78.20	+1.77
InvPT++ (Ye & Xu, 2024)	53.85	0.5096	18.67	78.10	+1.08
MTMamba (Lin et al., 2024)	55.82	0.5066	18.63	78.70	+2.38
MLoRE (Yang et al., 2024)	55.96	0.5076	18.33	78.43	+2.70
BridgeNet (Zhang et al., 2025)	55.51	0.4930	18.47	78.22	+2.95
MTMamba++ (Lin et al., 2025)	57.01	0.4818	18.27	**79.40**	+4.82
BIM (Cao et al., 2025)	57.40	0.4733	18.55	78.72	+4.83
AGG-MTL (Ours)	**57.52**	**0.4724**	**18.25**	78.70	**+5.31**

Method	Semseg	Parsing	Saliency	Normal	Boundary	Balance
	mIoU↑	mIoU↑	maxF↑	mErr↓	odsF↑	Δ_{MTL} ↑
PAD-Net (Xu et al., 2018)	53.60	59.60	65.80	15.30	72.50	-18.34
ASTMT (Maninis et al., 2019)	68.00	61.10	65.70	14.70	72.40	-13.60
MTI-Net (Vandenhende et al., 2020)	61.70	60.18	84.78	14.23	70.80	-10.63
ATRC (Brüggemann et al., 2021)	62.69	59.42	84.70	14.20	70.93	-10.54
ATRC-ASPP (Brüggemann et al., 2021)	63.60	60.23	83.91	14.30	70.86	-10.44
ATRC-BMTAS (Brüggemann et al., 2021)	67.67	62.93	82.29	14.24	72.42	-8.61
MQTransformer (Xu et al., 2024)	78.93	67.41	83.58	14.21	73.90	-3.90
InvPT (Ye & Xu, 2022)	79.03	67.61	84.81	14.15	73.00	-3.67
TaskPrompter (Ye & Xu, 2023)	80.89	68.89	84.83	13.72	73.50	-2.13
InvPT++ (Ye & Xu, 2024)	80.22	69.12	84.74	13.73	74.20	-2.09
MTMamba (Lin et al., 2024)	81.11	72.62	84.14	14.14	78.80	-0.47
MLoRE (Yang et al., 2024)	81.41	70.52	84.90	**13.51**	75.42	-0.75
BridgeNet (Zhang et al., 2025)	80.64	70.06	84.64	13.82	72.96	-2.19
MTMamba++ (Lin et al., 2025)	**81.94**	72.87	**85.56**	14.29	78.60	-0.13
BIM (Cao et al., 2025)	81.25	73.15	84.14	14.13	**79.05**	-0.21
AGG-MTL (Ours)	81.59	**73.24**	84.73	14.11	78.80	**+0.00**

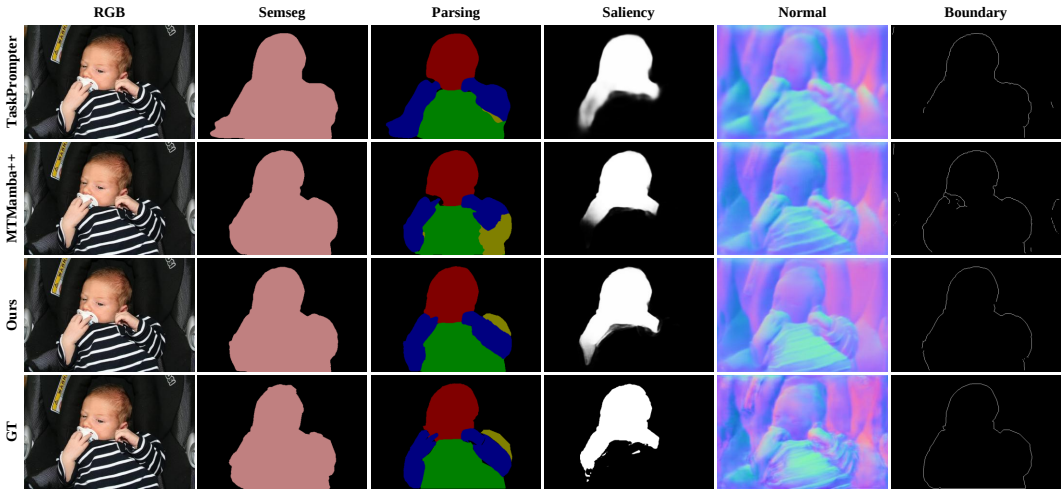

Figure 3: **Visual comparisons with state-of-the-art methods** on PASCAL-Context (Chen et al., 2014). AGG-MTL produces sharper predictions that more closely match the ground truth (GT). Results on NYUDv2 (Silberman et al., 2012) are provided in the supplementary material.

3.2 Comparisons with State-of-the-art Methods

Multi-task balance. Table 1 presents comprehensive comparisons with state-of-the-art methods on NYUDv2 (Silberman et al., 2012) and PASCAL-Context (Chen et al., 2014). AGG-MTL achieves the best overall multi-task balance on both benchmarks. Specifically, on NYUDv2, AGG-MTL improves Δ_{MTL} from $+4.83$ for the second-best method, BIM (Cao et al., 2025), to $+5.31$, corresponding to a gain of 0.48 points. On PASCAL-Context, AGG-MTL also ranks first, outperforming the second-best method, MTMamba++ (Lin et al., 2025), by 0.13 points. We further provide visual comparisons in Figure 3. Compared with state-of-the-art methods (Ye & Xu, 2023; Lin et al., 2025), AGG-MTL produces predictions that more closely adhere to the ground truth across multiple tasks, further supporting its superior multi-task performance.

Prediction sharpness. Table 2 (left) evaluates prediction sharpness on NYUDv2 (Silberman et al., 2012) using the task-appropriate metrics described in Section 3.1. AGG-MTL achieves the best result on every task. Notably, it improves Depth F1 from 0.0112 to 0.0240 over the second-best method, a relative gain of 114.29%, and Normal F1 from 0.1130 to 0.1240, a relative gain of 9.73%. Visual comparisons in Figure 3 further show that AGG-MTL delineates the person's right upper arm more accurately in Parsing, produces a sharper and more complete human silhouette in Saliency, and suppresses spurious responses to the black-and-white clothing texture in Normal, recovering a smoother surface orientation rather than imprinting the stripe pattern. Together, these results demonstrate consistently sharper and more structurally faithful predictions across various tasks.

Table 2: **Comparisons of prediction sharpness** (left) **and computational efficiency** (right) **with state-of-the-art methods** on NYUDv2 (Silberman et al., 2012).

Method	Semseg bIoU↑	Depth F1↑	Normal F1↑	Boundary AP↑
TaskPrompter (Ye & Xu, 2023)	41.75	0.0022	0.0742	64.07
MTMamba (Lin et al., 2024)	45.01	0.0058	0.1089	65.17
MLoRE (Yang et al., 2024)	42.42	0.0019	0.0863	63.16
BridgeNet (Zhang et al., 2025)	44.45	0.0112	0.1089	65.34
MTMamba++ (Lin et al., 2025)	45.39	0.0081	0.1130	64.20
AGG-MTL (Ours)	**45.49**	**0.0240**	**0.1240**	**68.99**

Method	FLOPs (G)↓	Params (M)↓	Balance Δ_{MTL} ↑
InvPT (Ye & Xu, 2022)	598	402	+0.09
TaskPrompter (Ye & Xu, 2023)	470	392	+1.77
InvPT++ (Ye & Xu, 2024)	–	402	+1.08
MTMamba (Lin et al., 2024)	541	**308**	+2.38
MLoRE (Yang et al., 2024)	552	706	+2.70
BridgeNet (Zhang et al., 2025)	628	463	+2.95
MTMamba++ (Lin et al., 2025)	524	315	+4.82
BIM (Cao et al., 2025)	547	388	+4.83
AGG-MTL (Ours)	**461**	386	**+5.31**

Computational efficiency. Table 2 (right) compares the computational costs of AGG-MTL with state-of-the-art methods on NYUDv2 (Silberman et al., 2012). AGG-MTL achieves the lowest FLOPs and maintains a relatively low parameter count, while delivering the best overall multi-task balance. Specifically, compared with BIM (Cao et al., 2025), the second-best method in terms of Δ_{MTL}, AGG-MTL reduces FLOPs by 15.7%. Together, Tables 1 and 2 demonstrate that AGG-MTL provides a stronger efficiency–balance–sharpness trade-off in multi-task dense prediction.

3.3 Ablation Studies

Table 3: **Efficacy of AGG-MTL.**

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑	FLOPs (G) ↓
STL	54.32	0.5166	19.21	77.30	+0.00	1075
MTL	53.72	0.5239	19.97	76.50	-1.88	466
Ours	57.52	0.4724	18.25	78.70	+5.31	461
w/o Aggregation	57.58	0.4718	18.24	78.70	+5.38	583
w/o Gating	57.43	0.4745	18.26	78.70	+5.16	445
w/o Guidance	57.11	0.4836	18.29	78.60	+4.50	457

Efficacy of AGG-MTL. Table 3 evaluates our three main contributions on NYUDv2 (Silberman et al., 2012). STL and MTL are the single-task and multi-task learning baselines, respectively, following prior settings (Cao et al., 2025; Lin et al., 2025; 2024). "w/o Aggregation" replaces aggregate-then-interact with interaction at every scale; "w/o Gating" removes the gate in Equation (11); and "w/o Guidance" removes the learned sampling offsets in Equation (14). Aggregation substantially reduces computational cost, while gating and guided upsampling both improve overall multi-task balance. More ablations are provided in the supplementary material.

Table 4: **Analysis of ACIT.**

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑
Soft	57.43	0.4729	18.28	78.70	+5.21
$\delta = 1.00$	57.43	0.4745	18.26	78.70	+5.16
$\delta = 0.75$	57.44	0.4732	18.26	78.70	+5.23
$\delta = 0.50$	57.52	0.4724	18.25	78.70	+5.31
$\delta = 0.25$	57.42	0.4728	18.30	78.60	+5.15
$\delta = 0.00$	57.26	0.4835	18.41	78.60	+4.42

Analysis of ACIT. In Figure 5, the gate maps highlight task-relevant structures, suggesting that ACIT preserves task-specific patterns while adaptively controlling cross-task interaction. We also evaluate soft gating and various hard-gating thresholds in Table 4. Hard gating at $\delta = 0.50$ achieves the best multi-task balance, indicating that an appropriate threshold helps retain useful task-specific information.

Table 5: **Analysis of MGUM.**

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑
FeatUp	57.18	0.4835	18.28	78.60	+4.55
AnyUp	57.27	0.4811	18.27	78.60	+4.72
Ours	57.52	0.4724	18.25	78.70	+5.31

Analysis of MGUM. We first replace MGUM with the single-task learnable upsamplers FeatUp (Fu et al., 2024) and AnyUp (Wimmer et al., 2026). As shown in Table 5, MGUM achieves higher multi-task balance, supporting the value of cross-task cues during upsampling. To analyze task relationships within MGUM, we compute the ℓ_1 norm of the weights associated with each task in the target-specific PWConv_t of Equation (13), and average the resulting norms. As shown in Figure 4, Boundary features receive substantial weight when predicting offsets for multiple target tasks. This suggests cross-task contributions to offset prediction. The relationship is particularly intuitive for semantic segmentation, where object outlines provide spatial cues for delineating semantic regions. Together, these observations suggest that MGUM draws on complementary cues across tasks to guide feature reconstruction and produce sharp predictions that preserve fine structures.

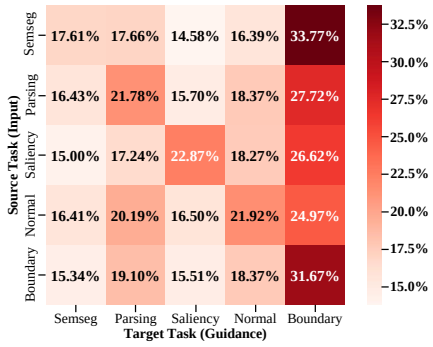

Figure 4: **Analysis of MGUM.**

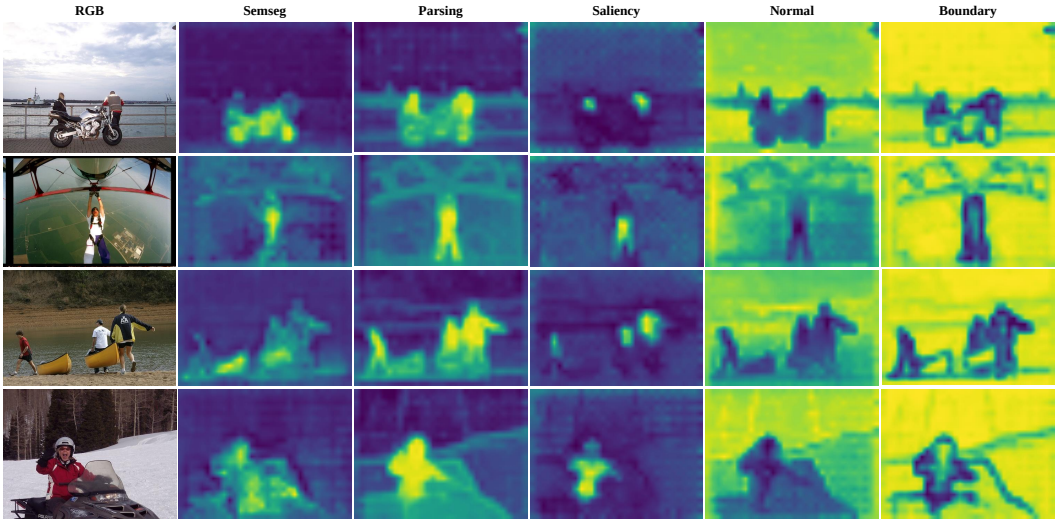

Figure 5: **Analysis of ACIT.** We visualize the spatial-channel gate in Equation (9) by averaging it across channels. Higher values indicate stronger preservation of task-specific patterns.

4 Related Works

Balance in multi-task dense prediction. Prior methods (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025) focus on *how* tasks interact; for example, BIM (Cao et al., 2025) exchanges information through bidirectional scanning. They leave it implicit *which* feature components should absorb cross-task cues and which should remain task-specific. High-frequency structural cues from boundary detection may help surface-normal estimation near object contours but interfere with normal-specific patterns in textureless regions. Our ACIT addresses this distinction through soft-to-hard gating, selectively incorporating cross-task cues while preserving critical task-specific patterns.

Efficiency in multi-task dense prediction. Existing methods improve efficiency from different perspectives. TaskPrompter (Ye & Xu, 2023) uses compact task prompts to reduce task-specific modeling cost, although their limited capacity may constrain rich cross-task interaction (Cao et al., 2025). Mamba-based methods, including MTMamba (Lin et al., 2024) and BIM (Cao et al., 2025), reduce the cost of sequence modeling through state-space models. Nevertheless, applying task interaction at multiple feature scales still incurs considerable computation. Our aggregate-then-interact strategy instead consolidates multi-scale features into compact representations and performs cross-task interaction at a single aggregated scale, avoiding repeated interaction across resolutions.

Sharpness in multi-task dense prediction. Previous methods (Cao et al., 2025; Lin et al., 2025; Zhang et al., 2025) commonly upsample features for each task independently, often using bilinear interpolation. This operation is neither learned nor informed by complementary cues from other tasks. General feature upsampling methods such as FeatUp (Fu et al., 2024) and AnyUp (Wimmer et al., 2026) recover spatial detail from low-resolution features, but are not designed to exploit cross-task relationships during the reconstruction of each target task. To extend task collaboration to high-resolution feature recovery, our MGUM predicts task-specific interpolation offsets from multi-task features, facilitating sharp predictions that preserve fine structures.

5 Conclusion and Limitations

We present AGG-MTL, advancing the efficiency–balance–sharpness trade-off through aggregated-scale interaction, adaptive soft-to-hard gating, and multi-task guided upsampling. Experiments show that AGG-MTL achieves the best overall multi-task balance, the lowest FLOPs, and the best sharpness scores. However, AGG-MTL does not lead every individual task or achieve the smallest parameter count, reflecting the inherently multi-objective nature of multi-task dense prediction, where improvements in task balance, computational cost, model size, and individual-task accuracy may involve different trade-offs. Future work could further compress the model, improve individual-task performance without sacrificing overall balance, or extend the framework to dynamic video inputs.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools for [assist with translation]. We have not used generative AI tools for [help develop theoretical models or conceptual frameworks, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset, support qualitative and thematic data analysis, interpret results], and [generate synthetic data sets, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses] are not applicable to this work. Additionally, we used generative AI tools for [edit a research paper to improve readability]. We have reviewed all AI-assisted work. [We repeatedly reviewed the LLM-generated wording for readability and translation accuracy.] We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Alexey Bochkovskiy, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R. Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *ICLR*, 2025.

David Brüggemann, Menelaos Kanakis, Anton Obukhov, Stamatios Georgoulis, and Luc Van Gool. Exploring relational context for multi-task dense prediction. In *ICCV*, 2021.

Mang Cao, Sanping Zhou, Yizhe Li, Ye Deng, Wenli Huang, and Le Wang. Enhancing mamba decoder with bidirectional interaction in multi-task dense prediction. In *ICCV*, 2025.

Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan L. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014.

Bowen Cheng, Ross B. Girshick, Piotr Dollár, Alexander C. Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *CVPR*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *ICLR*, 2024.

Kyunghoon Hur, Jungwoo Oh, Junu Kim, Jiyoun Kim, Min Jae Lee, Eunbyeol Cho, Seong-Eun Moon, Young-Hak Kim, Louis Atallah, and Edward Choi. Genhpf: General healthcare predictive framework for multi-task multi-source learning. *IEEE JBHI*, 2024.

Keishi Ishihara, Anssi Kanervisto, Jun Miura, and Ville Hautamäki. Multi-task learning with attention for end-to-end autonomous driving. In *CVPRW*, 2021.

Baijiong Lin, Weisen Jiang, Pengguang Chen, Yu Zhang, Shu Liu, and Ying-Cong Chen. Mtmamba: Enhancing multi-task dense scene understanding by mamba-based decoders. In *ECCV*, 2024.

Baijiong Lin, Weisen Jiang, Pengguang Chen, Shu Liu, and Ying-Cong Chen. Mtmamba++: Enhancing multi-task dense scene understanding via mamba-based decoders. *IEEE TPAMI*, 2025.

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. In *ICCV*, 2021.

Kevis-Kokitsi Maninis, Ilija Radosavovic, and Iasonas Kokkinos. Attentive single-tasking of multiple tasks. In *CVPR*, 2019.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *CVPR*, 2016.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012.

Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *ECCV*, 2020.

Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE TPAMI*, 2022.

Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. Anyup: Universal feature upsampling. In *ICLR*, 2026.

Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, 2015.

Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *CVPR*, 2018.

Yangyang Xu, Xiangtai Li, Haobo Yuan, Yibo Yang, and Lefei Zhang. Multi-task learning with multi-query transformer for dense prediction. *IEEE TCSVT*, 2024.

Yuqi Yang, Peng-Tao Jiang, Qibin Hou, Hao Zhang, Jinwei Chen, and Bo Li. Multi-task dense prediction via mixture of low-rank experts. In *CVPR*, 2024.

Hanrong Ye and Dan Xu. Inverted pyramid multi-task transformer for dense scene understanding. In *ECCV*, 2022.

Hanrong Ye and Dan Xu. Taskprompter: Spatial-channel multi-task prompting for dense scene understanding. In *ICLR*, 2023.

Hanrong Ye and Dan Xu. Invpt++: Inverted pyramid multi-task transformer for visual scene understanding. *IEEE TPAMI*, 2024.

Yanjie Ze, Ge Yan, Yueh-Hua Wu, Annabella Macaluso, Yuying Ge, Jianglong Ye, Nicklas Hansen, Li Erran Li, and Xiaolong Wang. Gnfactor: Multi-task real robot learning with generalizable neural feature fields. In *CoRL*, 2023.

Jingdong Zhang, Jiayuan Fan, Peng Ye, Bo Zhang, Hancheng Ye, Baopu Li, Yancheng Cai, and Tao Chen. Bridgenet: Comprehensive and effective feature interactions via bridge feature for multi-task dense predictions. *IEEE TPAMI*, 2025.

Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *CVPR*, 2019.

Ling Zhou, Zhen Cui, Chunyan Xu, Zhenyu Zhang, Chaoqun Wang, Tong Zhang, and Jian Yang. Pattern-structure diffusion for multi-task learning. In *CVPR*, 2020.

Supplementary Material

We provide additional details and analyses of AGG-MTL. Section A describes more implementation details (Section A.1) and provides further details on the evaluation metrics (Section A.2). Section B presents additional visual comparisons. Section C reports further ablation studies of the overall framework (Section C.1), ACIT (Section C.2), and the image encoder (Section C.3).

A Additional Experimental Setups

A.1 Implementation Details

Following previous works (Cao et al., 2025; Lin et al., 2024; Zhang et al., 2025), we use the AdamW optimizer with a weight decay of 1×10^{-5} and a polynomial learning-rate schedule. Images from NYUDv2 (Silberman et al., 2012) and PASCAL-Context (Chen et al., 2014) are resized to 448×576 and 512×512, respectively. For both datasets, we apply color jitter, random cropping, random

scaling, and random horizontal flipping, following the augmentation protocols of previous works (Cao et al., 2025; Lin et al., 2024; Zhang et al., 2025). We adopt the task-specific loss functions and weights used in previous works (Cao et al., 2025; Lin et al., 2024; Zhang et al., 2025). On NYUDv2, the loss weights for semantic segmentation, depth estimation, surface normal estimation, and object boundary detection are 1, 1, 10, and 50, respectively. On PASCAL-Context, the corresponding weights for semantic segmentation, human parsing, saliency estimation, object boundary detection, and surface normal estimation are 1, 2, 5, 50, and 10. The combined preliminary loss is weighted by 0.5 relative to the final prediction loss.

A.2 Evaluation Metrics

Task-specific metrics. Following established multi-task dense-prediction protocols (Cao et al., 2025; Lin et al., 2024; Zhang et al., 2025), we evaluate Semseg and Parsing using mean intersection over union (mIoU). For class c, let TP_c, FP_c, and FN_c denote the numbers of true-positive, false-positive, and false-negative pixels. Averaging over the C evaluated classes gives

$$\mathrm{mIoU} = \frac{1}{C}\sum_{c=1}^{C}\frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c}. \tag{A.1}$$

We evaluate Depth using root mean squared error (RMSE) over the set Ω of valid pixels:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\Omega|}\sum_{p\in\Omega}\left(\hat{d}_p - d_p\right)^2}, \tag{A.2}$$

where $\hat{d}_p$ and d_p are the predicted and ground-truth depths.

Normal is measured by mean error (mErr). Let $\tilde{\mathbf{n}}_p$ and $\mathbf{n}_p$ denote the unit-normalized predicted and ground-truth normals, respectively, and let Ω contain pixels whose ground-truth normal is valid.

$$\mathrm{mErr} = \frac{1}{|\Omega|}\sum_{p\in\Omega}\frac{180}{\pi}\,2\,\mathrm{atan2}(\|\tilde{\mathbf{n}}_p - \mathbf{n}_p\|_2,\ \|\tilde{\mathbf{n}}_p + \mathbf{n}_p\|_2)\,. \tag{A.3}$$

Saliency is evaluated using the maximum F-measure (maxF). For a threshold τ, let $P_{\mathrm{sal}}(\tau)$ and $R_{\mathrm{sal}}(\tau)$ be the dataset-level precision and recall of the binarized saliency maps. We compute

$$\mathrm{maxF} = \max_{\tau}\frac{2P_{\mathrm{sal}}(\tau)R_{\mathrm{sal}}(\tau)}{P_{\mathrm{sal}}(\tau) + R_{\mathrm{sal}}(\tau)}. \tag{A.4}$$

For Boundary, after non-maximum suppression, predicted and ground-truth boundary pixels are matched one-to-one within the prescribed spatial tolerance. Precision $P_{\mathrm{edge}}(\tau)$ and recall $R_{\mathrm{edge}}(\tau)$ are computed from counts accumulated over the test set at each threshold τ. We compute

$$\mathrm{odsF} = \max_{\tau}\frac{2P_{\mathrm{edge}}(\tau)R_{\mathrm{edge}}(\tau)}{P_{\mathrm{edge}}(\tau) + R_{\mathrm{edge}}(\tau)}. \tag{A.5}$$

Multi-task balance. Following prior multi-task evaluations (Vandenhende et al., 2022; Lin et al., 2024; Cao et al., 2025), we report the mean relative improvement over a reference model:

$$\Delta_{\mathrm{MTL}} = 100\% \times \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} s_t\frac{M_t - M_t^{\mathrm{ref}}}{M_t^{\mathrm{ref}}}, \qquad s_t = \begin{cases} +1, & \text{if higher is better for task } t,\\ -1, & \text{if lower is better for task } t. \end{cases} \tag{A.6}$$

Here M_t and M_t^{ref} are the task scores of the evaluated method and the reference model, respectively.

Prediction sharpness. For Semseg, we use Boundary IoU (bIoU) (Cheng et al., 2021). For predicted and ground-truth masks P and G,

$$\mathrm{bIoU} = \frac{|(G_d \cap G) \cap (P_d \cap P)|}{|(G_d \cap G) \cup (P_d \cap P)|}, \tag{A.7}$$

where G_d and P_d denote regions within distance d of the respective contours.

For Depth, we use the discontinuity F1 (Bochkovskiy et al., 2025). At each threshold r_j, discontinuities are identified from ratios of depths at adjacent pixels:

$$\text{Depth F1} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{10}\frac{r_j}{\sum_k r_k}\,\frac{2P_{i,j}R_{i,j}}{P_{i,j}+R_{i,j}}, \qquad r_j \in \text{linspace}(1.05, 1.25, 10). \tag{A.8}$$

For Normal, we adapt the same construction Bochkovskiy et al. (2025) by identifying discontinuities when the angular difference between adjacent normals exceeds θ_j:

$$\text{Normal F1} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{10}\frac{\theta_j}{\sum_k \theta_k}\,\frac{2P_{i,j}R_{i,j}}{P_{i,j}+R_{i,j}}, \qquad \theta_j \in \text{linspace}(5^\circ, 25^\circ, 10). \tag{A.9}$$

For Boundary, we follow the HED contour-matching protocol (Xie & Tu, 2015) and summarize its dataset-level precision–recall curve by average precision:

$$\text{AP} = \frac{1}{100}\sum_{k=0}^{100}\widetilde{P}\left(\frac{k}{100}\right), \tag{A.10}$$

where $\widetilde{P}(r)$ is precision interpolated at recall r.

B Additional Visual Comparisons

Figures A and B provide more comparisons on NYUDv2 (Silberman et al., 2012). In the indoor scenes, AGG-MTL recovers more distinct depth transitions around furniture and other scene structures than TaskPrompter (Ye & Xu, 2023) and MTMamba++ (Lin et al., 2025), while producing semantic and surface normal predictions that are visually closer to the ground truth. These examples illustrate the model's ability to retain spatial detail across dense prediction tasks. Figures C and D show results on PASCAL-Context (Chen et al., 2014). In the person-centered example, AGG-MTL better preserves the separation of human parts in parsing and yields a more complete saliency silhouette. In the outdoor example, it more clearly separates the car and nearby people in semantic segmentation and parsing, with a saliency prediction that follows the car's shape. Together, these complement the evaluation of multi-task balance and prediction sharpness in the main paper.

C Additional Ablation Studies

C.1 Efficacy of AGG-MTL.

Table A: **Efficacy of AGG-MTL** on PASCAL-Context (Chen et al., 2014).

Method	**Semseg** mIoU↑	**Parsing** mIoU↑	**Saliency** maxF↑	**Normal** mErr↓	**Boundary** odsF↑	**Balance** Δ_{MTL} ↑
Ours	81.59	73.24	84.73	14.11	78.80	+0.00
w/o Guidance	80.24	71.89	82.43	14.97	78.70	-2.49
w/o Gating	80.63	72.11	82.93	14.69	78.70	-1.82

Table A extends the component analysis to PASCAL-Context (Chen et al., 2014). Removing either multi-task guided upsampling or adaptive gating degrades all five task metrics. These results complement the NYUDv2 (Silberman et al., 2012) ablation in Table 3 and show that both components contribute to multi-task performance.

Table B examines their effects on prediction sharpness. Without guidance, Semseg bIoU decreases from 45.49 to 38.67, Depth F1 from 0.0240 to 0.0001, and Normal F1 from 0.1240 to 0.0312. Removing gating also reduces all four sharpness metrics.

Table C studies the spatial size of the aggregated representation. Increasing its resolution from $(H/16, W/16)$ to $(H/8, W/8)$ yields only a 0.04-point gain in Δ_{MTL}, while increasing FLOPs

Table B: **Efficacy of AGG-MTL** on prediction sharpness.

Method	Semseg bIoU↑	Depth F1↑	Normal F1↑	Boundary AP↑
Ours	45.49	0.0240	0.1240	68.99
w/o Guidance	38.67	0.0001	0.0312	68.09
w/o Gating	45.31	0.0213	0.1084	68.72

Table C: **Efficacy of AGG-MTL** on various h and w.

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑	FLOPs (G)↓	Params (M)↓
$(H/8, W/8)$	57.52	0.4721	18.23	78.70	+5.35	501	436
$(H/16, W/16)$	57.52	0.4724	18.25	78.70	+5.31	461	386
$(H/32, W/32)$	57.44	0.4745	18.37	78.60	+4.99	451	419

from 461G to 501G and parameters from 386M to 436M. Reducing the resolution to $(H/32, W/32)$ saves only 10G FLOPs, lowers Δ_{MTL} by 0.32 points, and increases the parameter count to 419M. We therefore use $(H/16, W/16)$ as the operating point.

C.2 Analysis of ACIT.

Table D: **Analysis of ACIT.**

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑
Hybrid	57.42	0.4735	18.27	78.60	+5.16
Ours	57.52	0.4724	18.25	78.70	+5.31

We compare our hard gating with a hybrid strategy. The hybrid strategy makes a hard decision only for high-confidence values and retains the continuous gate otherwise:

$$\tilde{G}_t^{\mathrm{hybrid}} = \begin{cases} G_t, & G_t \le 0.5, \\ 1, & G_t > 0.5. \end{cases} \tag{C.1}$$

As shown in Table D, hard gating achieves a Δ_{MTL} of $+5.31$, compared with $+5.16$ for one-sided hybrid gating, with slightly better results on all four tasks. This result favors hard selection over retaining a soft mixture for gate values at or below 0.5 in this comparison.

C.3 Analysis of Image Encoder.

Table E: **Analysis of image encoder.**

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Balance Δ_{MTL} ↑
MTMamba-B	53.62	0.5126	19.28	77.70	-0.09
BIM-B	54.32	0.4915	18.89	77.74	+1.77
Ours-B	54.86	0.4897	18.74	77.60	+2.26
MTMamba-L	55.82	0.5066	18.63	78.70	+2.38
BIM-L	57.40	0.4733	18.55	78.72	+4.83
Ours-L	57.52	0.4724	18.25	78.70	+5.31

We evaluate the effect of image encoders with various capacities (*e.g.*, Swin-Transformer Base and Large, denoted as B and L) on model performance. As shown in Table E, results demonstrate a consistent performance gain as the image encoder capacity increases.

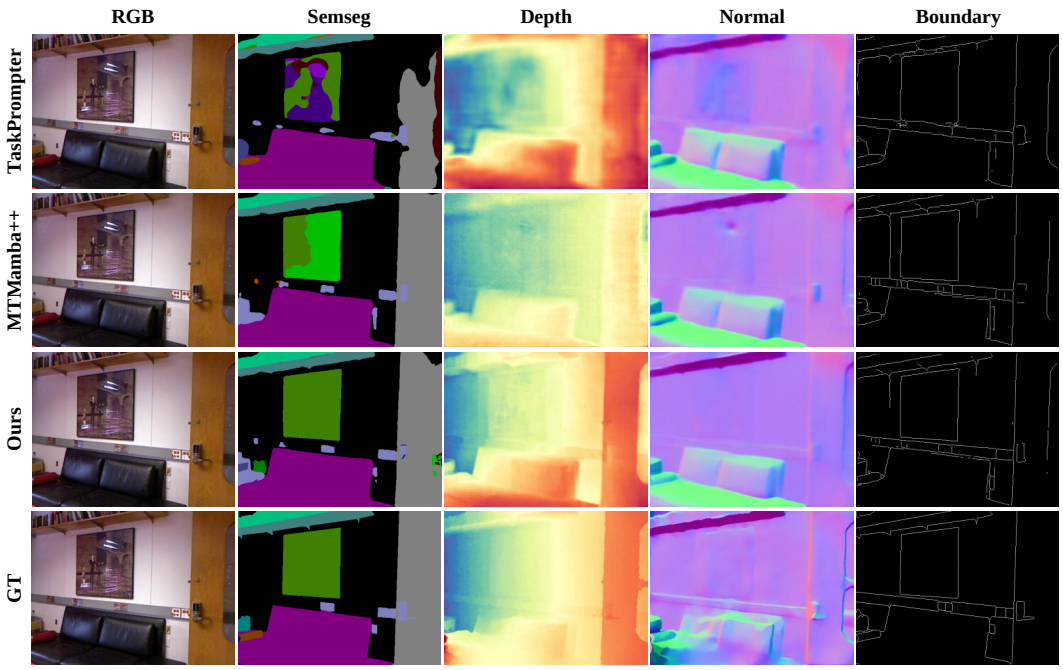

Figure A: **Additional visual comparisons** on NYUDv2 (Silberman et al., 2012).

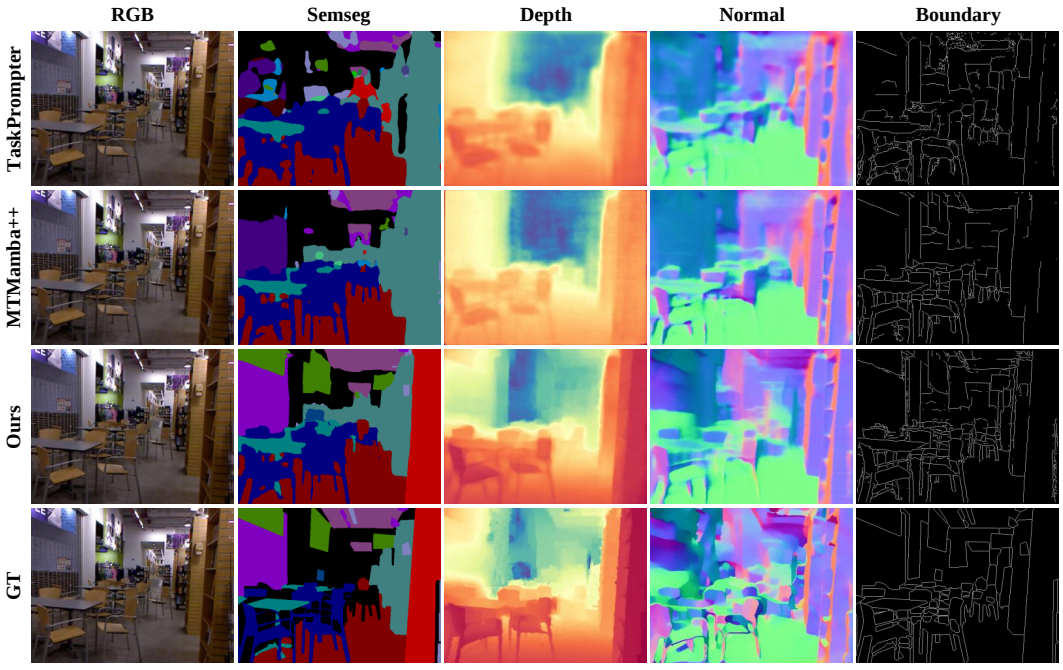

Figure B: **Additional visual comparisons** on NYUDv2 (Silberman et al., 2012).

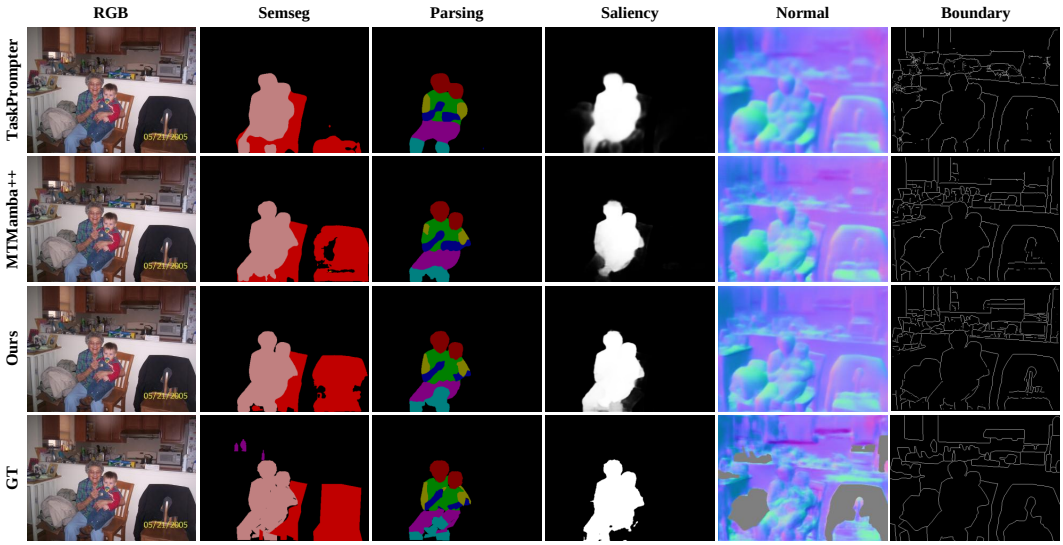

Figure C: **Additional visual comparisons** on PASCAL-Context (Chen et al., 2014).

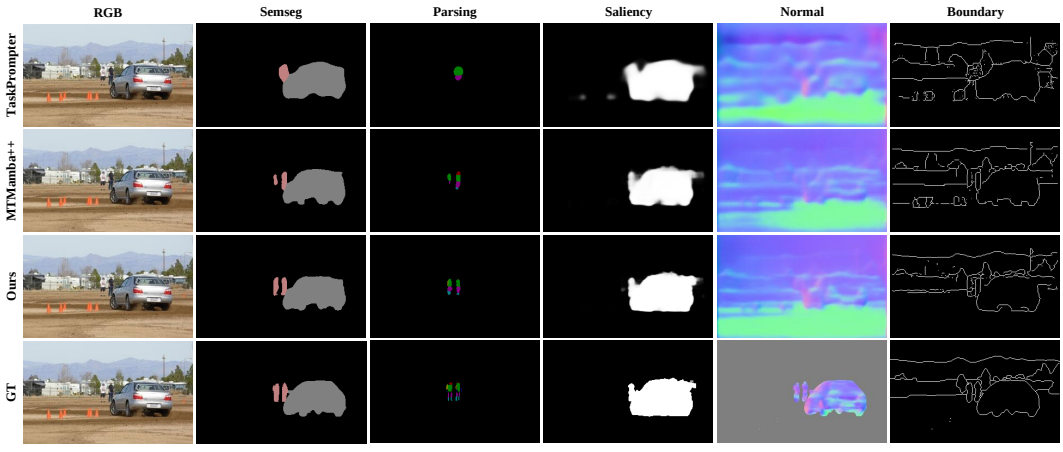

Figure D: **Additional visual comparisons** on PASCAL-Context (Chen et al., 2014).