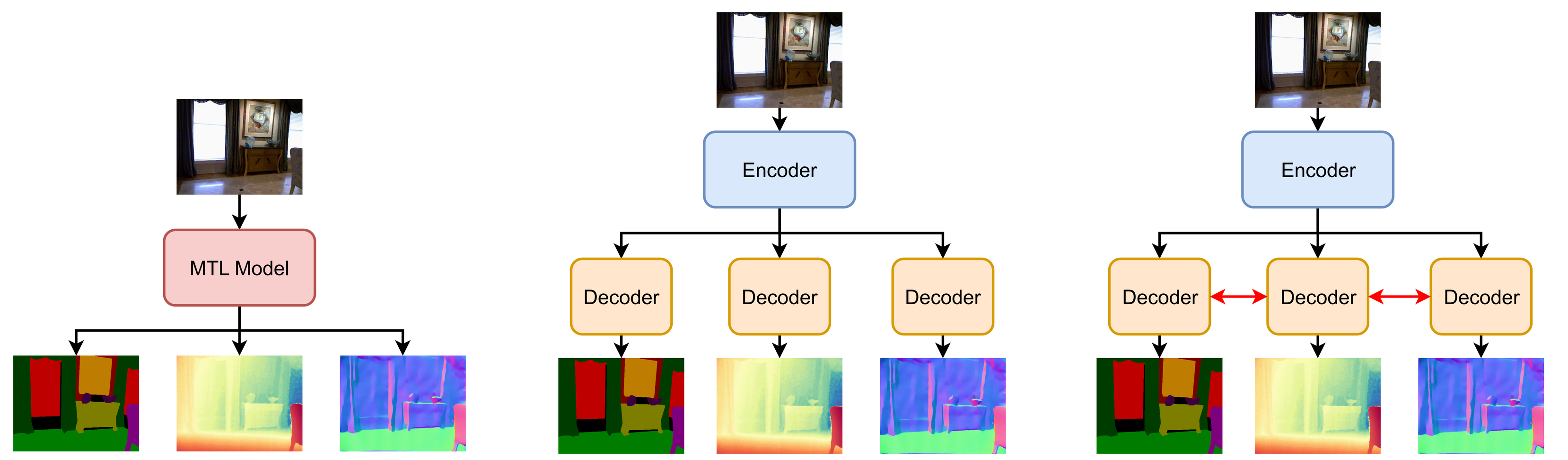
MTMamba: Enhancing Multi-Task Dense Scene Understanding by Mamba-Based Decoders

Baijiong Lin[1,4] Weisen Jiang[2,3] Pengguang Chen[5] Yu Zhang[3] Shu Liu[5] Ying-Cong Chen[1,2,4]

[1]HKUST(GZ) [2]HKUST [3]SUSTech [4]HKUST(GZ) - SmartMore Joint Lab [5]SmartMore

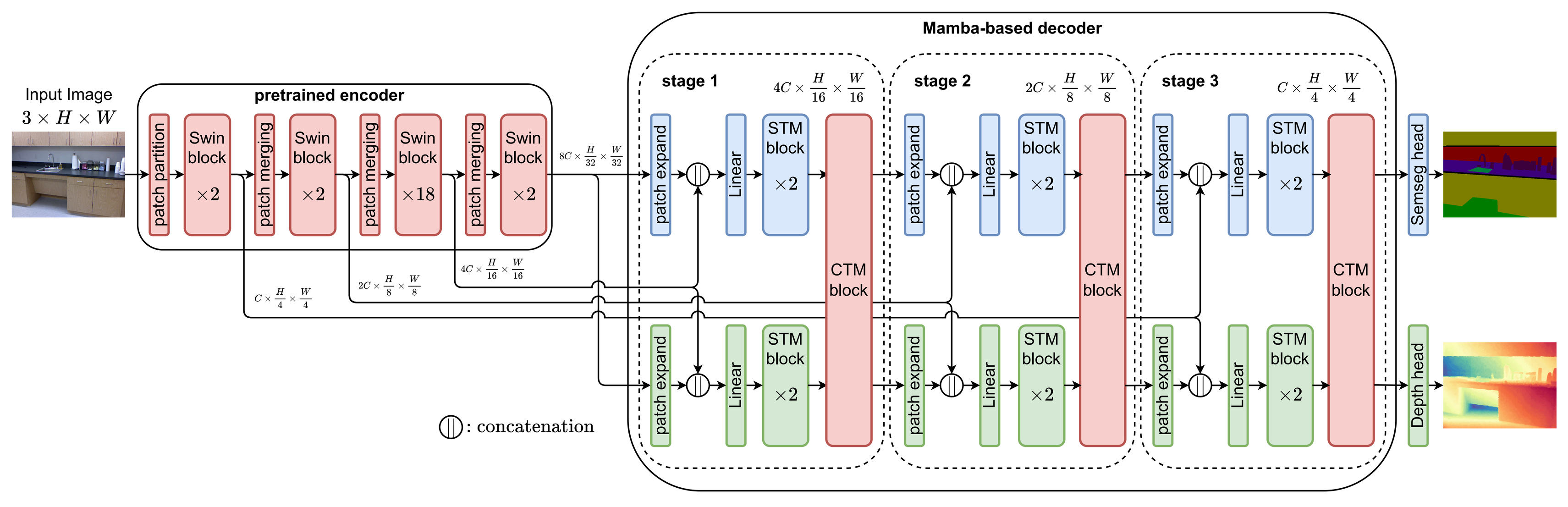
Background

- Multi-task dense scene understanding aims to train **a model for simultaneously handling multiple dense prediction tasks**, whose architecture is widely based on the encoder-decoder framework.



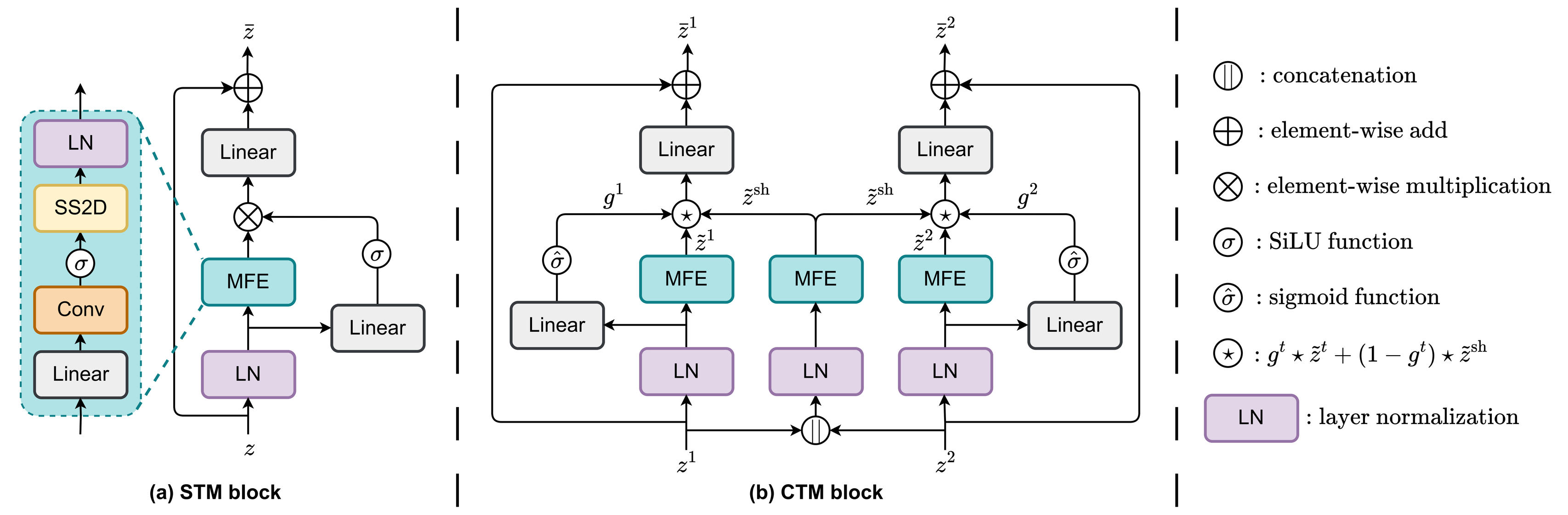
- Previous works have shown that
  1. **Enhancing cross-task correlation** in the task-specific decoders is crucial to achieving better performance;
  2. **Modeling long-range spatial relationships** plays an important role in Transformer-based methods to outperform CNN-based methods.
- Recently, **Mamba** has demonstrated **better capacity in long-range dependencies modeling** and **superior performance** than Transformers in various domains.
- **However,**
  1. Existing works on Mamba are limited to single-task learning scenarios, while **using Mamba to solve multi-task problems is still unexplored**;
  2. **Achieving cross-task correlation in Mamba remains under investigated**, which is critical for multi-task scene understanding.

Overall Architecture of MTMamba



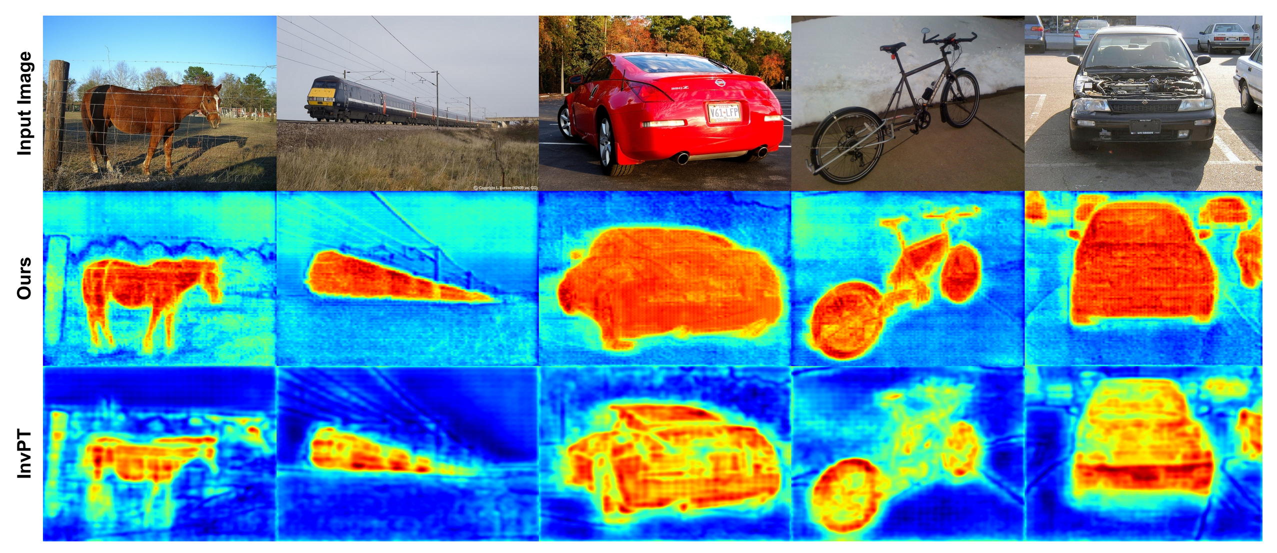
- The **pretrained encoder** (Swin-Large Transformer is used here) extracts multi-scale generic visual representations from the input RGB image;
- The **decoder** consists of three stages. Each stage contains **task-specific STM blocks to capture the long-range spatial relationship for each task** and **a shared CTM block to enhance each task's feature by exchanging knowledge across tasks**. Note that the structures of STM and CTM blocks in the decoder are Mamba-based;
- Each task has its own **prediction head** to generate the final predictions.

Two Types of Core Blocks

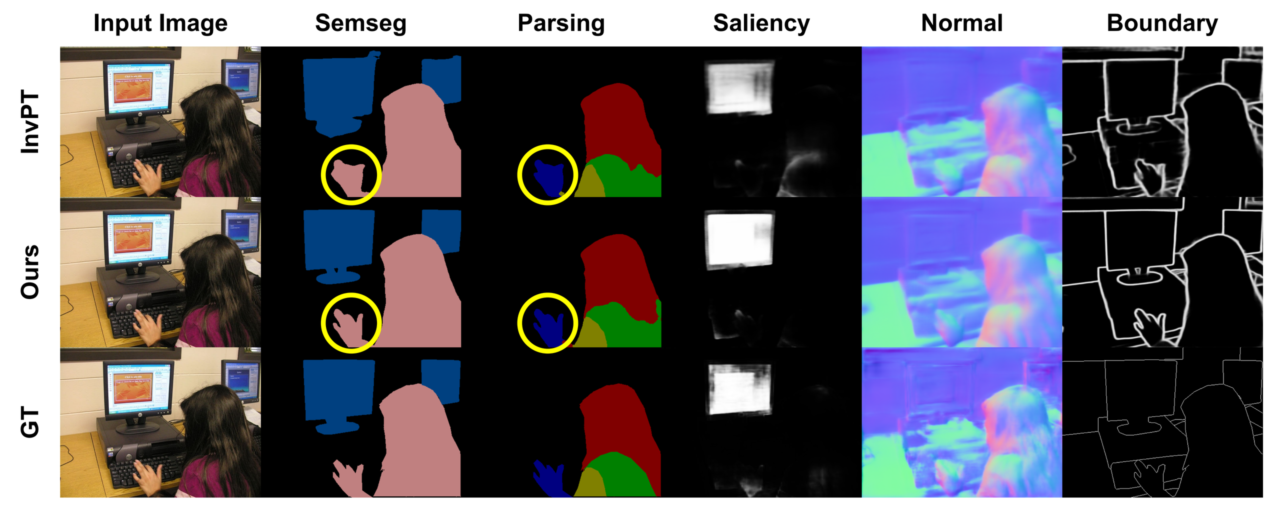


- The self-task Mamba (**STM**) block is responsible for learning task-specific features. Its core module is the Mamba-based feature extractor (MFE), where 1D SSM operation is extended on 2D images, namely SS2D. **MFE learns discriminant features** and an **input-dependent gate** $\sigma(\text{Linear}(\text{LN}(z)))$ further refines the learned features.
- The proposed cross-task Mamba (**CTM**) block contains $T+1$ MFE modules to exchange information across T task-specific input features. One module is used to generate a **global feature** $\tilde{z}^{sh}$ and the other T modules is to obtain the **task-specific feature** $\tilde{z}^t$. **Each task-specific output feature is the aggregation of task-specific feature $\tilde{z}^t$ and global feature $\tilde{z}^{sh}$ weighted by a task-specific and input-dependent gate g^t.**

Qualitative Results



- Visualization of the final decoder feature of semantic segmentation. Compared with the baseline, our method generates **more discriminative features**.



- Visualization of predictions on the PASCAL-Context dataset. Our method generates **better predictions with more accurate details** as marked in yellow circles.

Quantitative Results

Table 1: Comparison with state-of-the-art methods on NYUDv2 (**left**) and PASCAL-Context (**right**) datasets.

Method	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑
CNN-based decoder				
Cross-Stitch	36.34	0.6290	20.88	76.38
PAP	36.72	0.6178	20.82	76.42
PSD	36.69	0.6246	20.87	76.42
PAD-Net	36.61	0.6270	20.85	76.38
MTI-Net	45.97	0.5365	20.27	77.86
ATRC	46.33	0.5363	20.18	77.94
Transformer-based decoder				
InvPT	53.56	0.5183	19.04	78.10
MQTransformer	54.84	0.5325	19.67	78.20
Mamba-based decoder				
MTMamba (**ours**)	**55.82**	**0.5066**	**18.63**	**78.70**

Method	Semseg mIoU↑	Parsing mIoU↑	Saliency maxF↑	Normal mErr↓	Boundary odsF↑
CNN-based decoder					
ASTMT	68.00	61.10	65.70	14.70	72.40
PAD-Net	53.60	59.60	65.80	15.30	72.50
MTI-Net	61.70	60.18	84.78	14.23	70.80
ATRC	62.69	59.42	84.70	14.20	70.96
ATRC-ASPP	63.60	60.23	83.91	14.30	70.86
ATRC-BMTAS	67.67	62.93	82.29	14.24	72.42
Transformer-based decoder					
InvPT	79.03	67.61	**84.81**	14.15	73.00
MQTransformer	78.93	67.41	83.58	14.21	73.90
Mamba-based decoder					
MTMamba (**ours**)	**81.11**	**72.62**	84.14	**14.14**	**78.80**

- MTMamba achieves **superior performance** over CNN- and Transformer-based methods on both datasets.

Table 2: Effectiveness of the STM and CTM blocks on NYUDv2.

Method	Each Decoder Stage	Semseg mIoU↑	Depth RMSE↓	Normal mErr↓	Boundary odsF↑	Δ_m[%] ↑	#Param MB↓	FLOPs GB↓
Single-task	2*Swin	54.32	0.5166	19.21	77.30	0.00	888.77	1074.79
Multi-task	2*Swin	53.72	0.5239	19.97	76.50	-1.87	303.18	466.35
MTMamba	♦1*STM	54.61	0.5059	19.00	77.40	+0.95	252.51	354.13
	♠2*STM	54.66	**0.4984**	18.81	78.20	+1.84	276.48	435.47
	■3*STM	54.75	0.5054	18.81	78.20	+1.55	300.45	516.82
	★2*STM+1*CTM	**55.82**	0.5066	**18.63**	**78.70**	**+2.38**	307.99	540.81

- ♠ vs. "Multi-task": **STM** achieves **better performance** and is **more efficient** than the **Swin** Transformer block;
- ★ vs. ♠/■: Simply increasing the number of STM blocks from two to three fails to boost the performance. However, **when the CTM is used, MTMamba has a significantly better performance** in terms of Δ_m;
- ★ vs. "Single-task": **MTMamba significantly outperforms "Single-task"** on all tasks.

Summary

- We propose **MTMamba**, a novel multi-task architecture with a Mamba-based decoder for multi-task dense scene understanding, which can effectively **model long-range dependency** and **achieve cross-task interaction**;
- We design a novel **CTM** block to **enhance information exchange across tasks** in multi-task dense prediction;
- Experiments on two benchmark datasets demonstrate the **superiority** of MTMamba on multi-task dense prediction over previous CNN-based and Transformer-based methods;
- Qualitative evaluations show that MTMamba **captures discriminative features** and **generates precise predictions**;
- **We extend MTMamba to MTMamba++ by developing a new CTM block and achieve better performance.**

MTMamba Paper MTMamba++ Paper Code