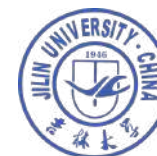


PointMAC: Meta-Learned Adaptation for Robust Test-Time Point Cloud Completion

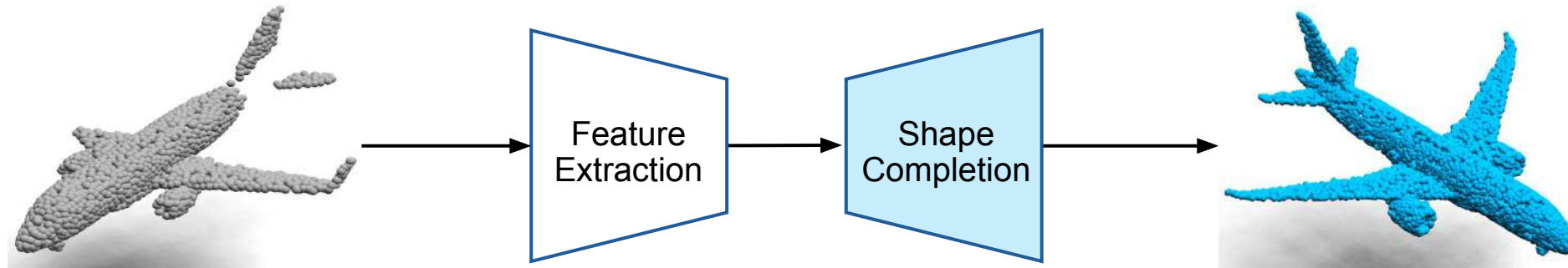
Linlian Jiang, Rui Ma, Li Gu, Ziqiang Wang, Xinxin Zuo , Yang Wang 

Concordia University, Jilin University, Mila - Quebec AI Institute,
Engineering Research Center of Knowledge-Driven Human-Machine Intelligence, MOE, China

NeurIPS 2025



Background: Point Cloud Completion



Input:
incomplete shape

Output:
complete shape

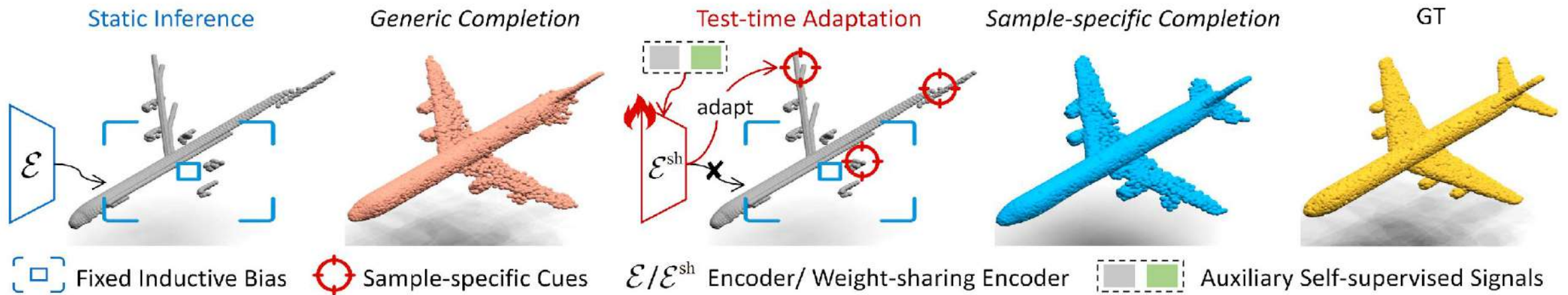
Goal: complete missing parts of a point cloud into a full 3D shape.

Motivation

What's the problem (and why do existing methods fail)?

Static Inference → *Generic Completion*

- Completion models **frozen at deployment** — infer but never learn
- Relying on **training priors** — **CANNOT** adapt to unseen geometry / noise
- Predicts *plausible* but *non-specific* shapes



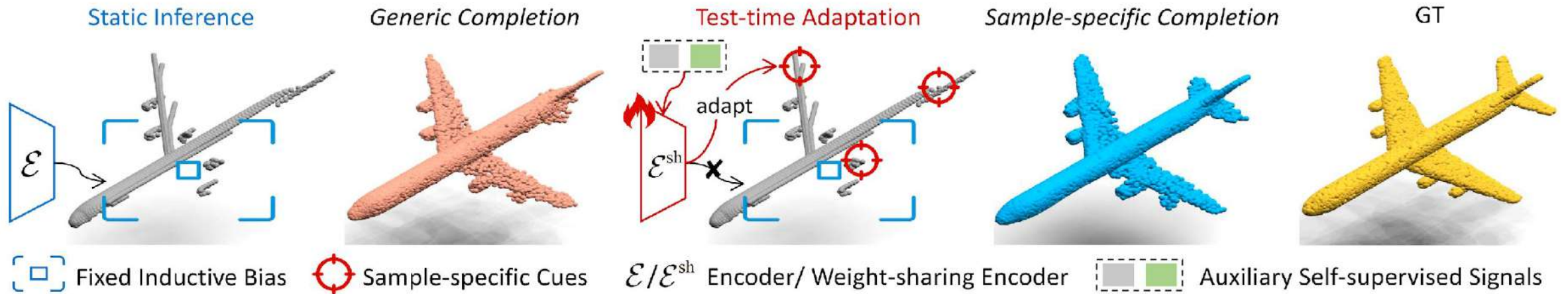
Limitation: (1) static inference and (2) training-learned inductive biases

Motivation

What's our solution?

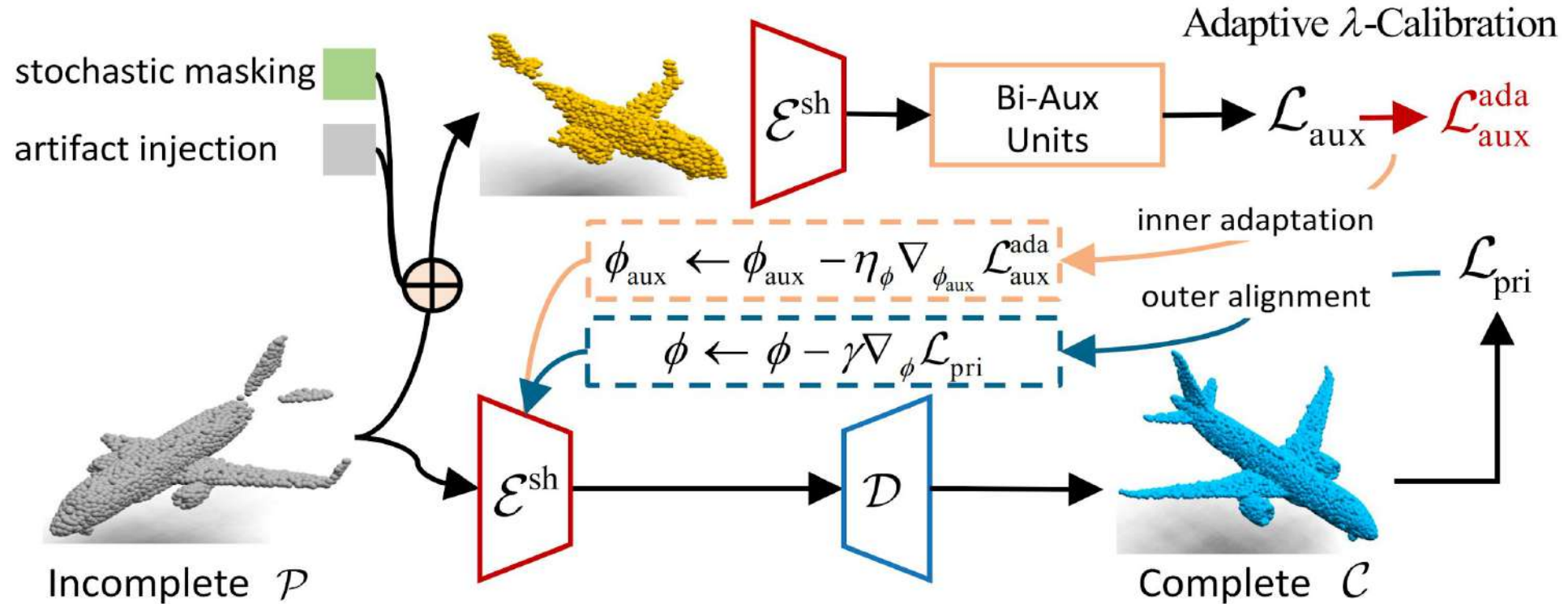
Adapt at Test Time!

- Test-time adaptation (TTA) + meta-auxiliary learning → on-the-fly shared encoder adaptation



Core Challenge: from static inference → adaptive, self-improving completion

PointMAC: Meta-Learned Test-Time Training



(i) Inner Auxiliary Adaptation

(ii) Outer Primary Alignment

$$\phi_{aux}^{smr(t+1)} \leftarrow \phi_{aux}^{smr(t)} - \alpha \cdot \nabla_{\phi_{aux}^{smr}} \mathcal{L}_{aux}^{smr} \left(\tilde{\mathcal{P}}^{(t)}, \mathcal{P}^{(t)}; \phi_{pri}^{sh(t)}, \phi_{aux}^{sh(t)}, \phi_{aux}^{smr(t)} \right),$$

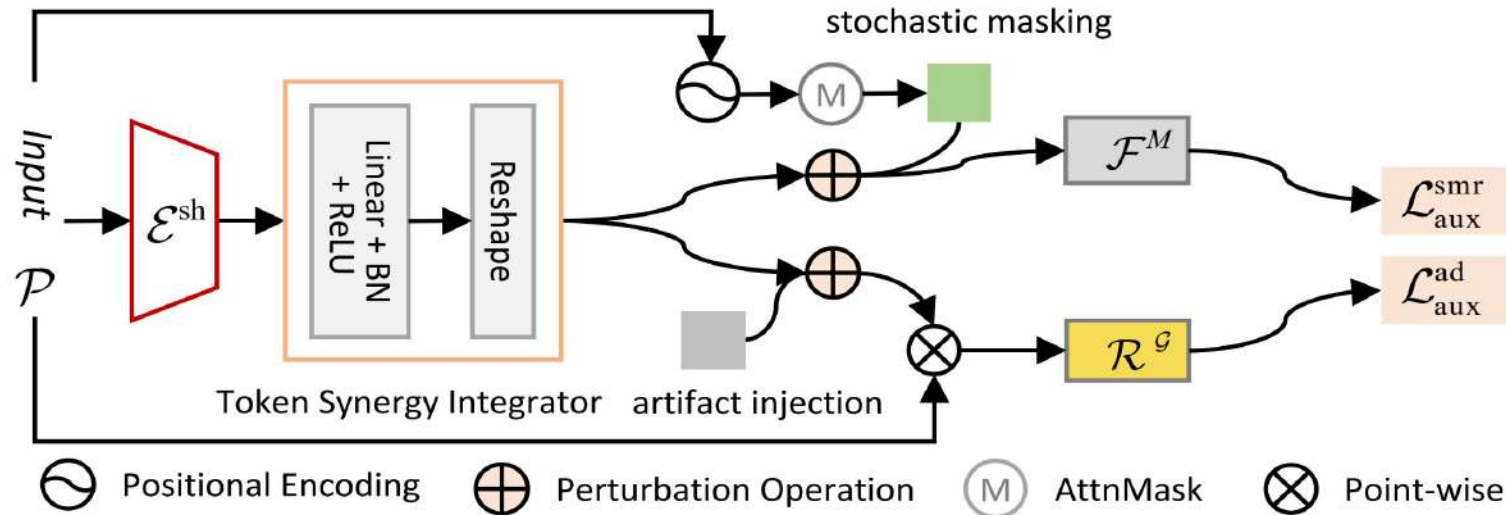
$$\phi_{aux}^{ad(t+1)} \leftarrow \phi_{aux}^{ad(t)} - \beta \cdot \nabla_{\phi_{aux}^{ad}} \mathcal{L}_{aux}^{ad} \left(\hat{\mathcal{P}}^{(t)}, \mathcal{P}^{(t)}; \phi_{pri}^{sh(t)}, \phi_{aux}^{sh(t)}, \phi_{aux}^{ad(t)} \right),$$

$$\phi_{pri}^{(t+1)} \leftarrow \phi_{pri}^{(t)} - \gamma \cdot \nabla_{\phi_{pri}^{(t)}} \left(\frac{1}{T} \sum_{i=1}^T \mathcal{L}_{pri}(\mathcal{C}^{(i)}, \mathcal{P}^{(i)}; \phi_{pri}^{(t)}) \right).$$

PointMAC: Meta Auxiliary Learning — Bi-Aux Units

- **Bi-Aux Units:** Two Self-Supervised Adaptation Signals

Purpose: on-the-fly adaptation under structural & sensor incompleteness



(1) Stochastic Masked Reconstruction:

randomly mask regions \rightarrow recover geometry
 \rightarrow robust to missing parts

$$\mathcal{L}_{\text{aux}}^{\text{smr}} = \mathcal{L}_{\text{CD}}(\tilde{\mathcal{P}}, \mathcal{P}).$$

(2) Artifact Denoising:

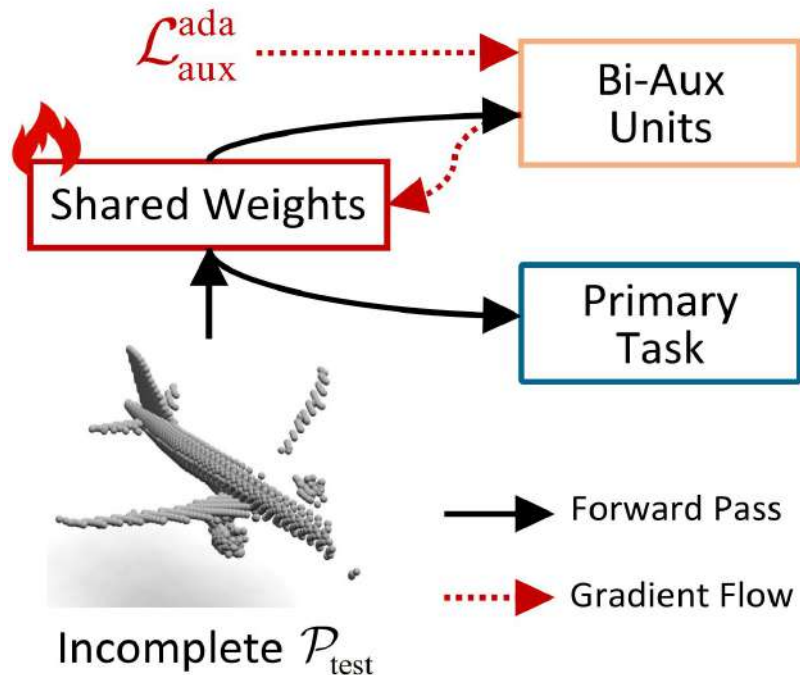
add realistic noise \rightarrow restore clean structure
 \rightarrow suppress artifacts

a noisy partial input $\bar{\mathcal{P}} = \mathcal{P} + \mathcal{N}(0, \sigma^2)$

a clean and dense point cloud $\hat{\mathcal{P}} = \Upsilon_{\epsilon}^{\text{ad}}(\bar{\mathcal{P}})$

$$\mathcal{L}_{\text{aux}}^{\text{ad}} = \mathcal{L}_{\text{CD}}(\hat{\mathcal{P}}, \mathcal{P})$$

PointMAC: Sample-Specific Test-Time Adaptation



Algorithm 1: Sample-Specific Test-Time Adaptation

Input: Trained parameters $\phi = \{\phi_{\text{pri}}^{\text{sh}}, \phi_{\text{pri}}, \phi_{\text{aux}}^{\text{sh}}, \phi_{\text{aux}}^{\text{smr}}, \phi_{\text{aux}}^{\text{ad}}\}$;

test input $\mathcal{P}_{\text{test}}$; step size η ; number of steps K

Output: Adapted encoder $\phi_{\text{pri}}^{\text{sh}}$

Apply stochastic masking and artifact injection using \mathcal{M} and σ via Aux^{smr} and Aux^{ad} ;

for $t = 0$ **to** $K - 1$ **do**

/* Inner-loop adaptation */

$\tilde{\mathcal{P}} \leftarrow Aux^{\text{smr}}_Forward(\mathcal{P}_{\text{test}}, \mathcal{M}; \phi_{\text{aux}}^{\text{smr}});$

$\hat{\mathcal{P}} \leftarrow Aux^{\text{ad}}_Forward(\mathcal{P}_{\text{test}}, \sigma; \phi_{\text{aux}}^{\text{ad}});$

$\mathcal{L}_{\text{aux}}^{\text{ada}} \leftarrow \lambda_{\text{smr}} \cdot \mathcal{L}_{\text{aux}}^{\text{smr}}(\tilde{\mathcal{P}}, \mathcal{P}_{\text{test}}) + \lambda_{\text{ad}} \cdot \mathcal{L}_{\text{aux}}^{\text{ad}}(\hat{\mathcal{P}}, \mathcal{P}_{\text{test}});$

$\phi_{\text{pri}}^{\text{sh}(t+1)} \leftarrow \phi_{\text{pri}}^{\text{sh}(t)} - \eta \cdot \nabla_{\phi_{\text{pri}}^{\text{sh}}} \mathcal{L}_{\text{aux}}^{\text{ada}};$ /* Gradient descent on shared encoder */

return $\phi_{\text{pri}}^{\text{sh}(K)}$

Adaptive λ -Calibration. dynamically adjusts the auxiliary weights λ_{smr} and λ_{ad}

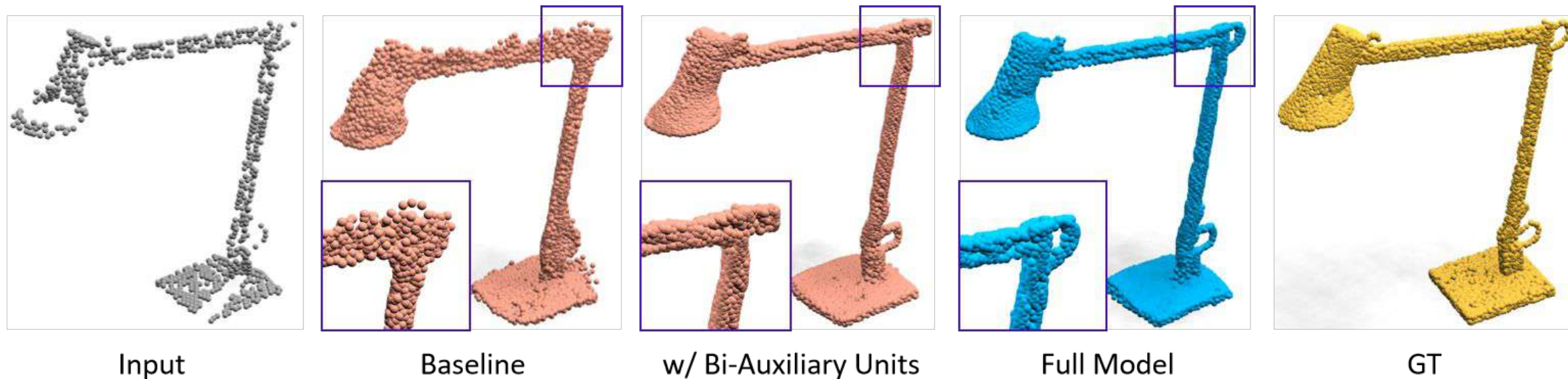
$$w_{\text{smr}} = (\log(1 + \lambda_{\text{smr}}^2) / [\log(1 + \lambda_{\text{smr}}^2) + \log(1 + \lambda_{\text{ad}}^2)]), \quad w_{\text{ad}} = 1 - w_{\text{smr}}, \quad (6)$$

$$\mathcal{L}_{\text{aux}}^{\text{ada}} = w_{\text{smr}} \cdot \mathcal{L}_{\text{aux}}^{\text{smr}} + w_{\text{ad}} \cdot \mathcal{L}_{\text{aux}}^{\text{ad}}. \quad (7)$$

Both the auxiliary branch parameters $\phi_{\text{aux}} = \{\phi_{\text{aux}}^{\text{smr}}, \phi_{\text{aux}}^{\text{ad}}\}$ and the weighting coefficients $\lambda \in \{\lambda_{\text{smr}}, \lambda_{\text{ad}}\}$ are jointly updated via gradient descent:

$$\phi_{\text{aux}} \leftarrow \phi_{\text{aux}} - \eta_{\phi} \nabla_{\phi_{\text{aux}}} \mathcal{L}_{\text{aux}}^{\text{ada}}, \quad \lambda \leftarrow \lambda - \eta_{\lambda} \nabla_{\lambda} \mathcal{L}_{\text{aux}}^{\text{ada}}. \quad (8)$$

PointMAC: Ablation



Generic completion
(One-size-fits-all)

Sample-specific completion
(Test-time Adaptation)

PointMAC: Main Experiment Results

PointMAC is among the **most thoroughly evaluated methods**, covering [synthetic](#) (ShapeNet, PCN), [simulated](#) (MVP), and [real-world](#) (KITTI) benchmarks, and it achieves **new SOTA** performance.

Table 1: Synthetic experiments on the ShapeNet dataset.

CD- ℓ_2 ($\times 1000$)	Table	Chair	Plane	Car	Sofa	Bird House	Bag	Remote	Key board	Rocket	CD-S	CD-M	CD-H	CD-Avg	F1
FoldingNet [18]	2.53	2.81	1.43	1.98	2.48	4.71	2.79	1.44	1.24	1.48	2.67	2.66	4.05	3.12	0.082
PCN [4]	2.13	2.29	1.02	1.85	2.06	4.50	2.86	1.33	0.89	1.32	1.94	1.96	4.08	2.66	0.133
PoinTr [7]	0.81	0.95	0.44	0.91	0.79	1.86	0.93	0.53	0.38	0.57	0.58	0.88	1.79	1.09	0.464
SnowflakeNet [5]	0.75	0.84	0.42	0.88	0.72	1.74	0.81	0.48	0.36	0.51	0.52	0.80	1.62	0.98	0.477
SeedFormer [6]	0.72	0.81	0.40	0.89	0.71	-	-	-	-	-	0.50	0.77	1.49	0.92	0.472
ProxyFormer [8]	0.70	0.83	0.34	0.78	0.69	-	-	-	-	-	0.49	0.75	1.55	0.93	0.483
EINet [39]	0.66	0.79	0.41	0.84	0.69	1.49	0.73	0.42	0.33	0.49	0.49	0.75	1.46	0.90	0.432
CRA-PCN [19]	0.66	0.74	0.37	0.85	0.66	1.36	0.73	0.43	0.35	0.50	0.48	0.71	1.37	0.85	-
Ours	0.65	0.72	0.34	0.80	0.64	1.34	0.72	0.40	0.31	0.47	0.47	0.69	1.34	0.83	0.490

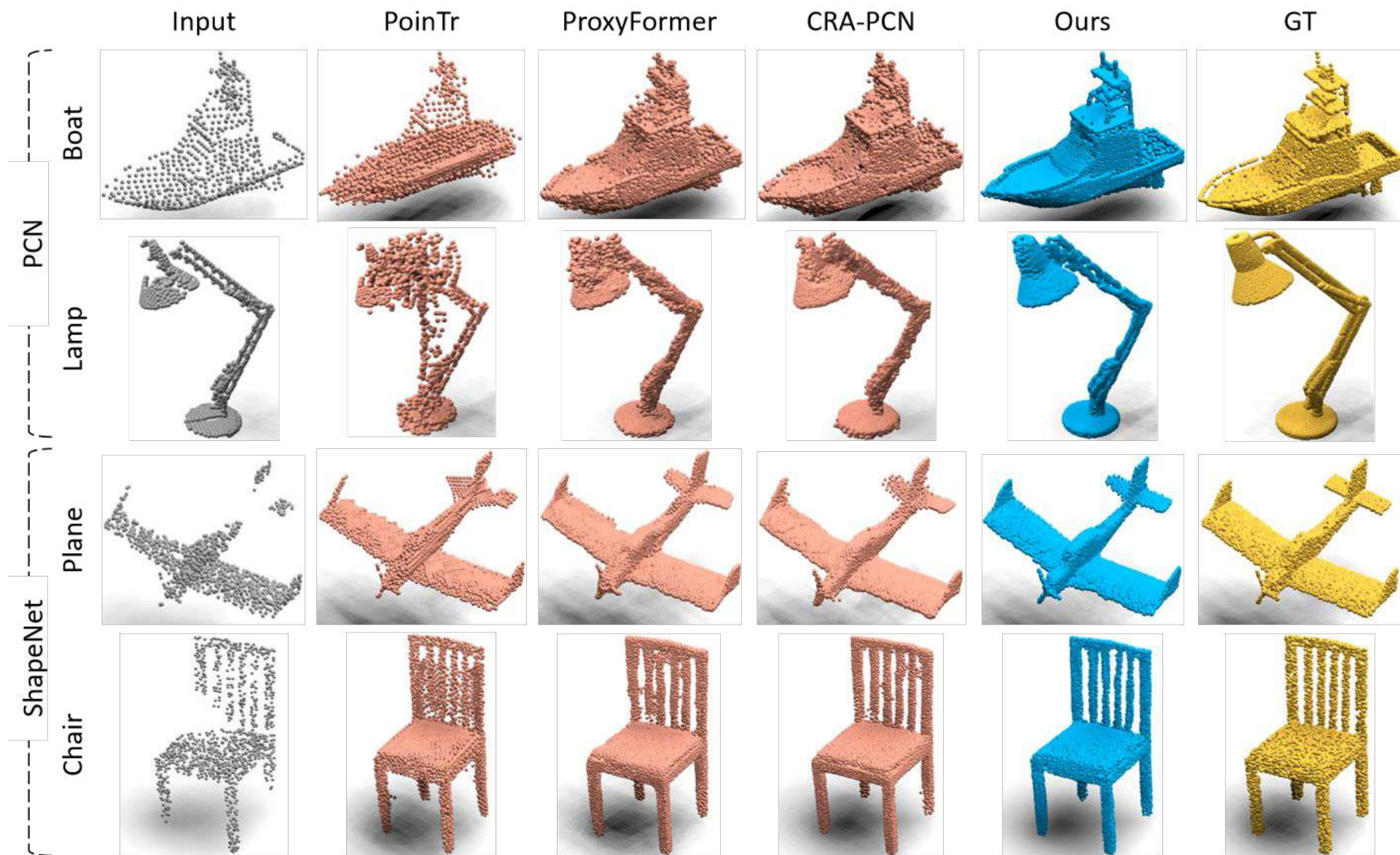
Table 2: Simulated experiments on the MVP dataset.

	PCN [4]	TopNet [41]	MSN [42]	CDN [43]	ECG [44]	VRCNet [38]	CRA-PCN [19]	Ours
CD- ℓ_2 \downarrow	9.77	10.11	7.90	7.25	6.64	5.96	5.33	5.24
F1 \uparrow	0.320	0.308	0.432	0.434	0.476	0.499	0.529	0.537

Table 3: Real-world experiments on the KITTI dataset.

	PCN [4]	FoldingNet [18]	TopNet [41]	GRNet [45]	PoinTr [7]	SeedFormer [6]	ProxyFormer [8]	EINet [39]	Ours
Fidelity \downarrow	2.235	7.467	5.354	0.816	0.000	0.151	0.000	1.48	0.135
MMD \downarrow	1.366	0.537	0.636	0.568	0.526	0.516	0.508	0.512	0.477

PointMAC: Main Experiment Results



PointMAC: Conclusion

This work overcomes the limits of *static inference* and *training priors* through meta-learned *test-time adaptation*, enabling robust point cloud completion.

- A *paradigm shift* from traditional static, one-size-fits-all inference.
- Robust to *domain shift* in real-world deployment.
- A *generalizable, plug-and-play* test-time adaptation framework for diverse 3D tasks.

Paper: <https://arxiv.org/abs/2510.10365>



Linlian Jiang



Rui Ma



Li Gu



Ziqiang Wang



Xinxin Zuo 



Yang Wang 



Project Page