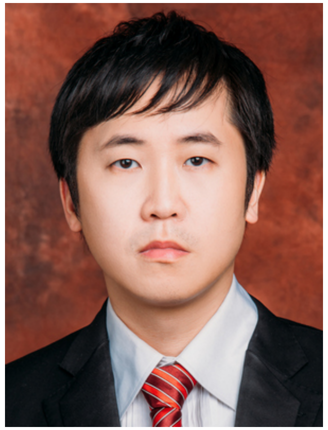
CVPR
JUNE 17-21, 2024
SEATTLE, WA

Unsupervised Deep Unrolling Networks for Phase Unwrapping

Zhile Chen

Yuhui Quan*

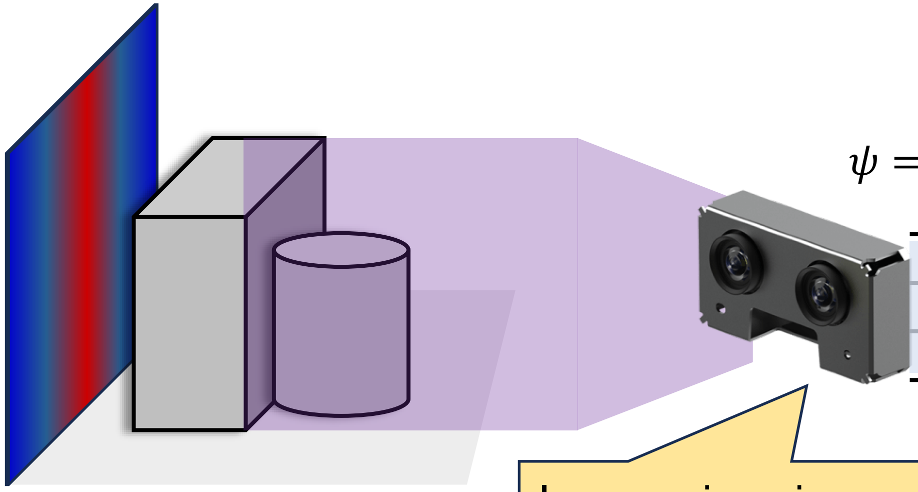


Hui Ji

华南理工大学
South China University of Technology

NUS
National University of Singapore

Phase Unwrapping (PU) in Imaging



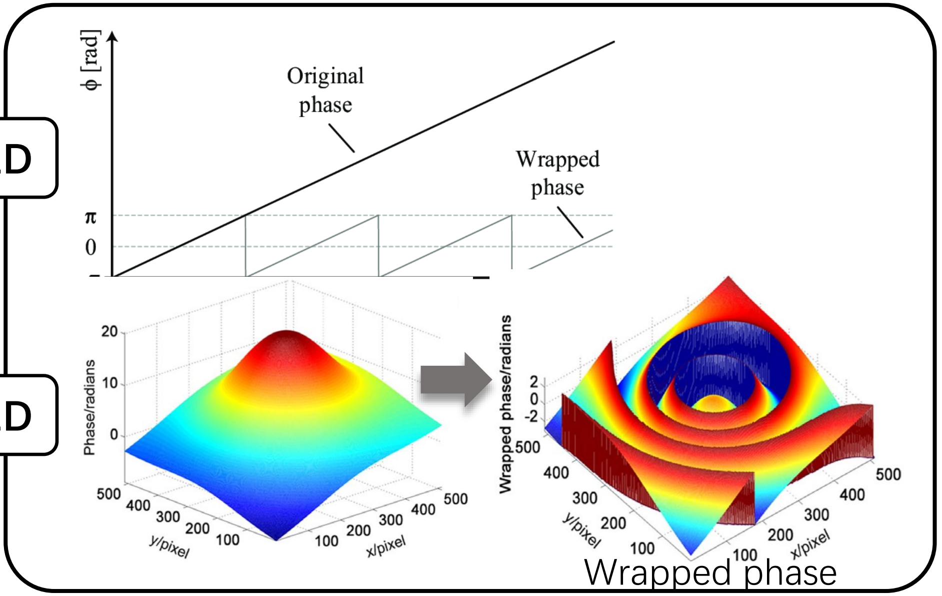
$$\psi = \arctan\frac{\sin(\phi)}{\cos(\phi)}$$

Measure

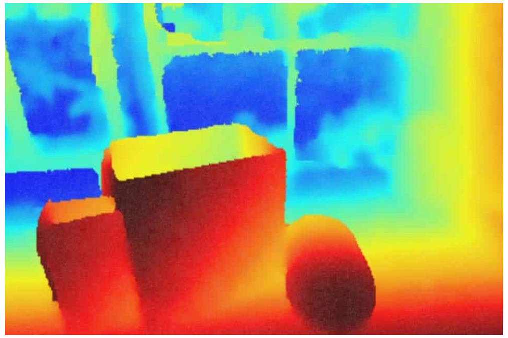
In many imaging systems, initial measurement yields a phase image wrapped in $[-\pi, \pi)$.

1D

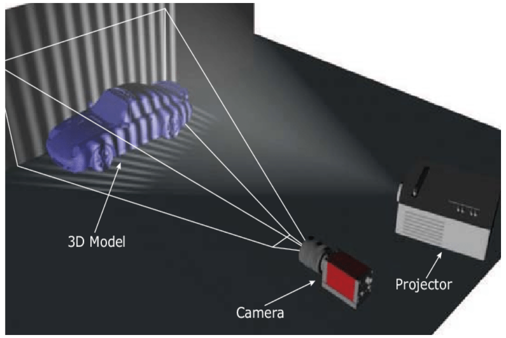
2D



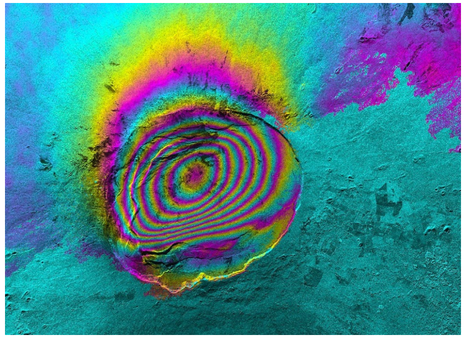
Applications



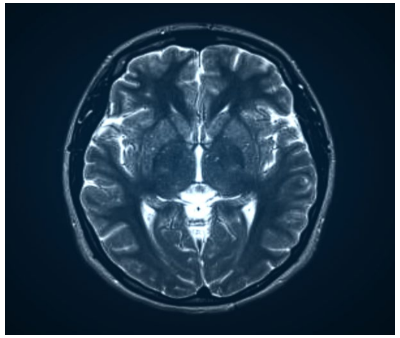
3D Depth Sensing



Fringe Projection

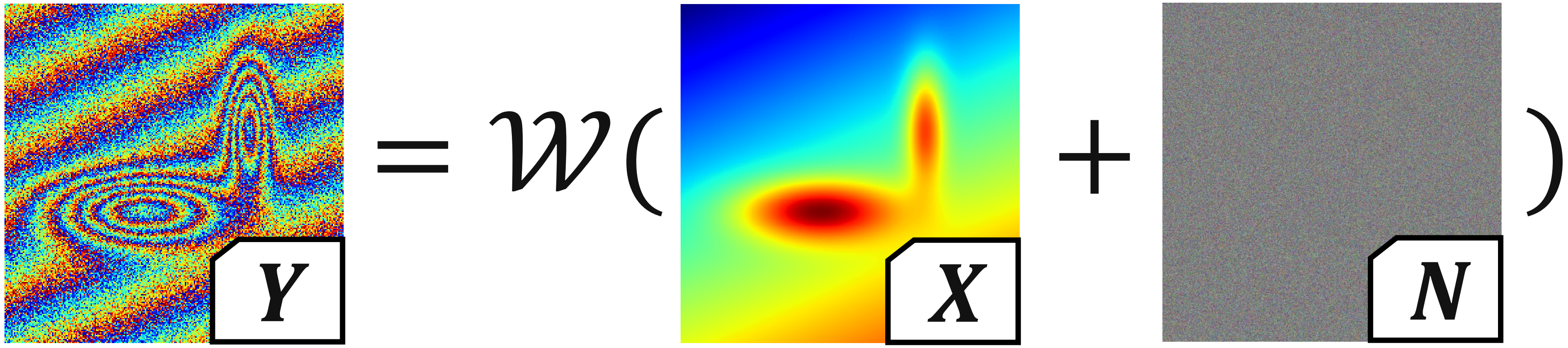


InSAR Imaging



MRI

Formulation of PU



$\boldsymbol{Y} \in \mathbb{R}^{M \times N}$: Wrapped phase image; $\boldsymbol{X} \in \mathbb{C}^{M \times N}$: GT Phase image;

$\boldsymbol{N} \in \mathbb{R}^{M \times N}$: Measurement noise;

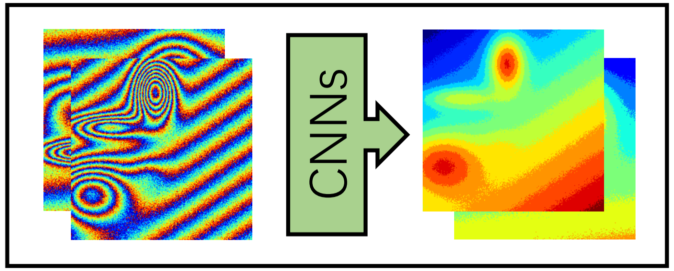
$\mathcal{W}$: Wrapping operator: $\mathcal{W}(\theta) = \big((\theta + \pi) \bmod 2\pi\big) - \pi$

PU needs to reconstruct $\boldsymbol{X}$ from $\boldsymbol{Y}$.

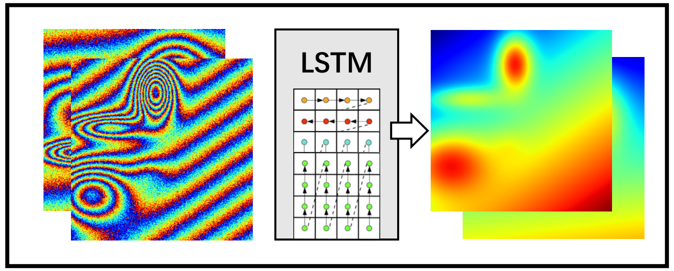
<u>Challenges</u>: 1) $\mathcal{W} \rightarrow$ Solution ambiguity; 2) $\boldsymbol{N} \rightarrow$ Noise corruption.

End-to-End Supervised Learning for PU

- Standard CNNs; Treat PU as pixel-wise classification, *e.g.*, PhaseNet 2.0 [1] and EESANet [2].
  - ※ **Struggle to scale to wide-range phase.**
  - ※ **Cannot capture long-range spatial dependencies.**



- RNN-based Method; Directly map wrapped phases to unwrapped ones, *e.g.*, SQD-LSTM [3].
  - ※ **A limited number of paths (due to cost constraint) cannot capture rich dependencies.**



Both are impractical !! → GT phase images and wrap counts are costly to collect.

[1] Spoorthi G E, Gorthi R K S S, Gorthi S. PhaseNet 2.0: Phase unwrapping of noisy data based on deep learning approach. IEEE TIP, 2020.
[2] Zhang J, Li Q. EESANet: edge-enhanced self-attention network for two-dimensional phase unwrapping. Optics Express, 2022.
[3] Perera M V, De Silva A. A joint convolutional and spatial quad-directional LSTM network for phase unwrapping. ICASSP, 2021.

Dataset-free Unsupervised Learning for PU

Re-parameterize a phase image via a CNN and optimize it with wrapped fidelity.

✓ No need for GT phase images.
※ Slow due to per-sample training.
※ Ignore knowledge from external data.

$\text{argmin}_{\boldsymbol{\theta}} \|\mathcal{W}(f_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})) - \boldsymbol{y}\|_p^p$

For efficient inference & releasing the use of GT
→ **End-to-end unsupervised deep learning for PU**

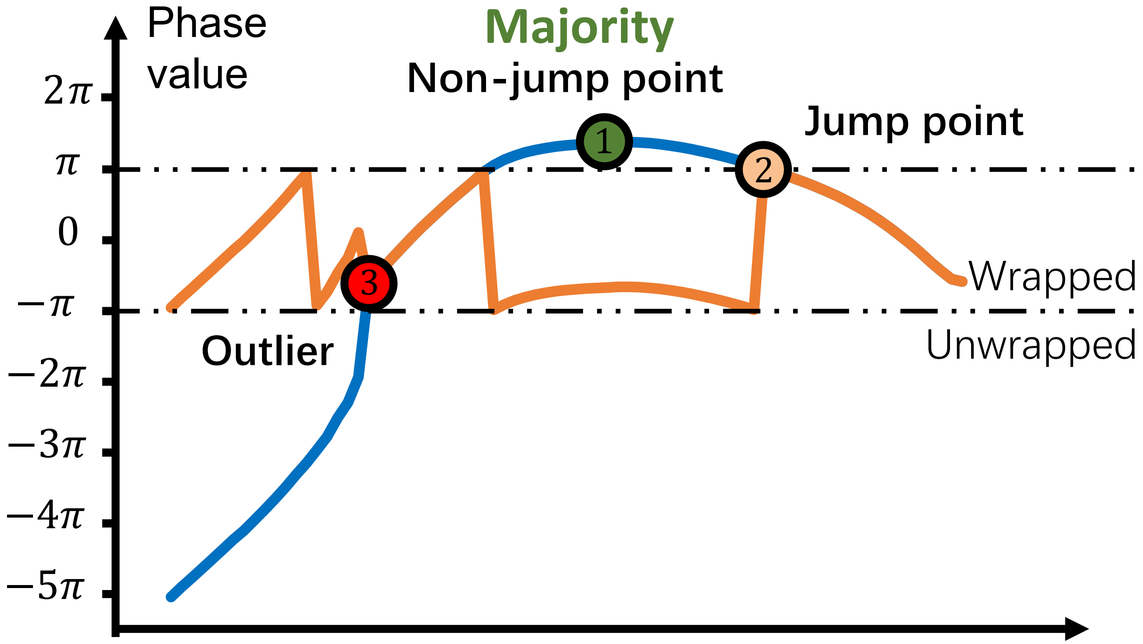
[4] Yang F, Pham T-a, Brandenberg N, et al. Robust phase unwrapping via deep image prior for quantitative phase imaging. IEEE TIP, 2021

Contributions

The 1st external unsupervised DL approach for end-to-end PU.

- The first exploration of deep unrolling network for PU, founded on a variational model utilizing wrapped gradients and perceiving outliers.
- A re-corruption-based self-reconstruction loss function with noise tolerance to leverage Itoh's continuity condition, and a self-distillation loss function for improved generalization.
- Better than existing unsupervised methods & competitive against the supervised ones.

Core Ideas



For those whose adjacent points share the same wrap count, $\nabla \boldsymbol{Y}[m,n] = \nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n]$. (1)

Due to majority of non-jump points, $\nabla \boldsymbol{Y}$ can be viewed as noisy label of $\nabla \boldsymbol{X}$, for self-supervised loss function.

2D Itoh's Condition

In **noisy** case: $\boldsymbol{Y} = \mathcal{W}(\boldsymbol{X} + \boldsymbol{N})$, for points satisfying $\|\nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n]\|_\infty < \pi$,

$$\mathcal{W}(\nabla \boldsymbol{Y}[m,n]) = \nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n]. \quad (2)$$

Utilized for the unfolded regularization model.

For those share different wrap count and satisfy $\|\nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n]\|_\infty \geq \pi$,

$$\mathcal{W}(\nabla \boldsymbol{Y}[m,n]) = \nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n] + 2\pi \boldsymbol{K}. \quad (3)$$

Variational Regularization Model

- Unfold the proximal gradient descend solver of:

$$\min_{\boldsymbol{X},\boldsymbol{E}} \|\nabla\boldsymbol{X} - \mathcal{W}(\nabla\boldsymbol{Y}) + \boldsymbol{E}\|_{\mathrm{F}}^2 + \phi(\boldsymbol{X}) + \psi(\boldsymbol{E}).$$

Regularizing $\boldsymbol{X}$

Regularizing $\boldsymbol{E}$ for sparsity

- Leveraging an $\boldsymbol{E}$ for absorbing the $2\pi\boldsymbol{K}$ in $\mathcal{W}(\nabla\boldsymbol{Y}) = \nabla\boldsymbol{X} + \nabla\boldsymbol{N} + 2\pi\boldsymbol{K}$.

For j from 1 to J,

$$\boldsymbol{X}_{(j)}^{(t)} = \boldsymbol{V}_{(j-1)}^{(t)} + \lambda^{(t)}\mathrm{div}(\nabla\boldsymbol{V}_{(j-1)}^{(t)} - (\mathcal{W}(\nabla\boldsymbol{Y}) - \boldsymbol{E}^{(t-1)})),$$

$$\alpha_{(j)} = 1/2 \cdot (1 + \sqrt{1 + 4\alpha_{(j-1)}^2}),$$

$$\boldsymbol{V}_{(j)}^{(t)} = \boldsymbol{X}_{(j)}^{(t)} + \frac{\alpha_{(j-1)} - 1}{\alpha_{(j)}} \cdot (\boldsymbol{X}_{(j)}^{(t)} - \boldsymbol{X}_{(j-1)}^{(t)}),$$

Accelerated gradient descent for solving data fidelity term

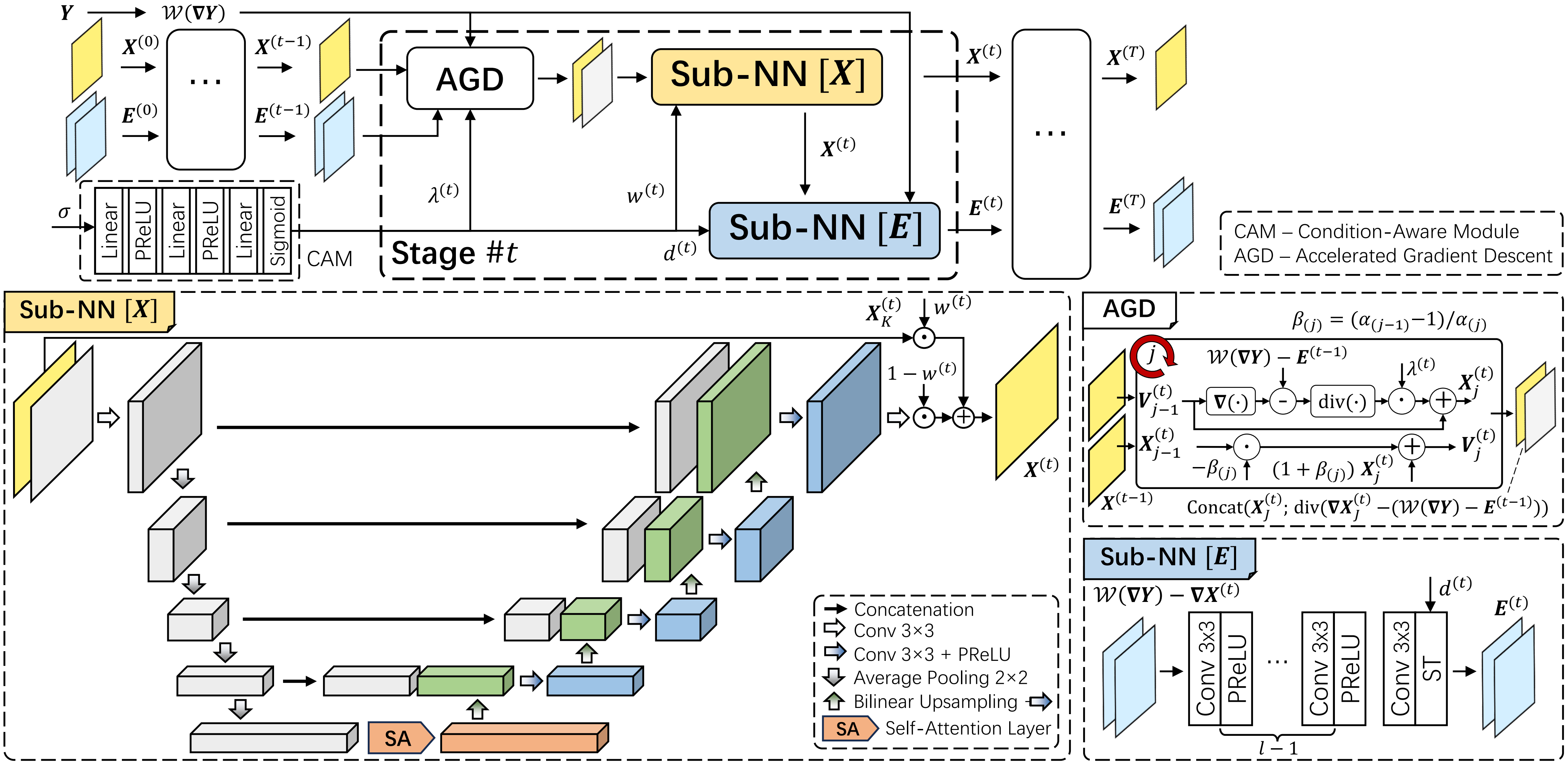
$$\boldsymbol{X}^{(t)} = \mathrm{NN}_{\phi}\left(\boldsymbol{X}_{(J)}^{(t)}, \mathcal{G}_t\left(\boldsymbol{X}_{(J)}^{(t)}\right), w^{(t)}\right),$$

$$\boldsymbol{E}^{(t)} = \mathrm{NN}_{\psi}(\boldsymbol{E}^{(t-1)}, d^{(t)}),$$

Regularization terms replaced by two sub-NNs

where $\lambda^{(t)}, w^{(t)}, d^{(t)}$ are learned from condition (noise strength, etc.) via CAM module.

U3Net (U3 = Unsupervised, Unrolling, Unwrapping)



Unsupervised Loss functions

- Noise-resistant self-reconstruction loss:

$$\mathcal{L}_{\mathrm{sr}} := \mathbb{E}_{\boldsymbol{U}} \left\| \mathcal{W}\left[\nabla \mathcal{F}\left(\mathcal{W}(\nabla \boldsymbol{Y} + \nabla \boldsymbol{U})\right) - (\nabla \boldsymbol{Y} - \nabla \boldsymbol{U})\right] \right\|_{\mathrm{F}}^{2}$$

Unsupervised loss approaximates supervised loss.

Proposition. Let $\boldsymbol{Y} = \mathcal{W}(\boldsymbol{X} + \boldsymbol{N})$. Suppose $\nabla \boldsymbol{Y}[m,n] = \nabla \boldsymbol{X}[m,n] + \nabla \boldsymbol{N}[m,n]$ is satisfied at all points. Assume that $\boldsymbol{N}, \boldsymbol{U} \sim \mathcal{P}$ are independent. Then, we have that

$$\mathbb{E}_{\boldsymbol{Y},\boldsymbol{U}} \left\| \nabla \mathcal{F}\left(\mathcal{W}(\nabla \boldsymbol{Y} + \nabla \boldsymbol{U})\right) - (\nabla \boldsymbol{Y} - \nabla \boldsymbol{U}) \right\|_{\mathrm{F}}^{2} = \mathbb{E}_{\boldsymbol{X},\boldsymbol{N},\boldsymbol{U}} \left\| \nabla \mathcal{F}\left(\mathcal{W}(\nabla \boldsymbol{Y} + \nabla \boldsymbol{U})\right) - \nabla \boldsymbol{X} \right\|_{\mathrm{F}}^{2} + C_0,$$

where C_0 is a constant.

- Once we sample $\boldsymbol{U}$ from the distribution of $\boldsymbol{N}$, the training with $\mathcal{L}_{\mathrm{sr}}$ is equivalent to learning noiseless spatial gradient, supervised by $\nabla \boldsymbol{X}$.
- Introducing an outer $\mathcal{W}$ for counteracting the impact of outliers.
- Inductive bias of unrolling CNNs helps reduce the ambiguity of outliers.

Unsupervised Loss functions

- Self-distillation loss:

$$\mathcal{L}_{\mathrm{sd}} := \mathbb{E}_{\boldsymbol{U}} \left\| \nabla \mathcal{F}\big(\mathcal{W}(\nabla \boldsymbol{Y})\big) - \nabla \bar{\mathcal{F}}\big(\mathcal{W}(\nabla \boldsymbol{Y} + \nabla \boldsymbol{U})\big) \right\|_{\mathrm{F}}^{2},$$

$\bar{\mathcal{F}}$ denotes the NN detached from the previous iteration with stopped gradient.

➢ Reducing the NN's prediction variance, enhancing the PU accuracy.

➢ Reconciling the input of unrolling network, *e.g.* $\mathcal{W}(\nabla \boldsymbol{Y} + \nabla \boldsymbol{U}) \rightarrow \mathcal{W}(\nabla \boldsymbol{Y})$, improving generalization ability.

- Total loss:

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{sr}} + \eta \mathcal{L}_{\mathrm{sd}}, \qquad \eta \in \mathbb{R}^{+}$$

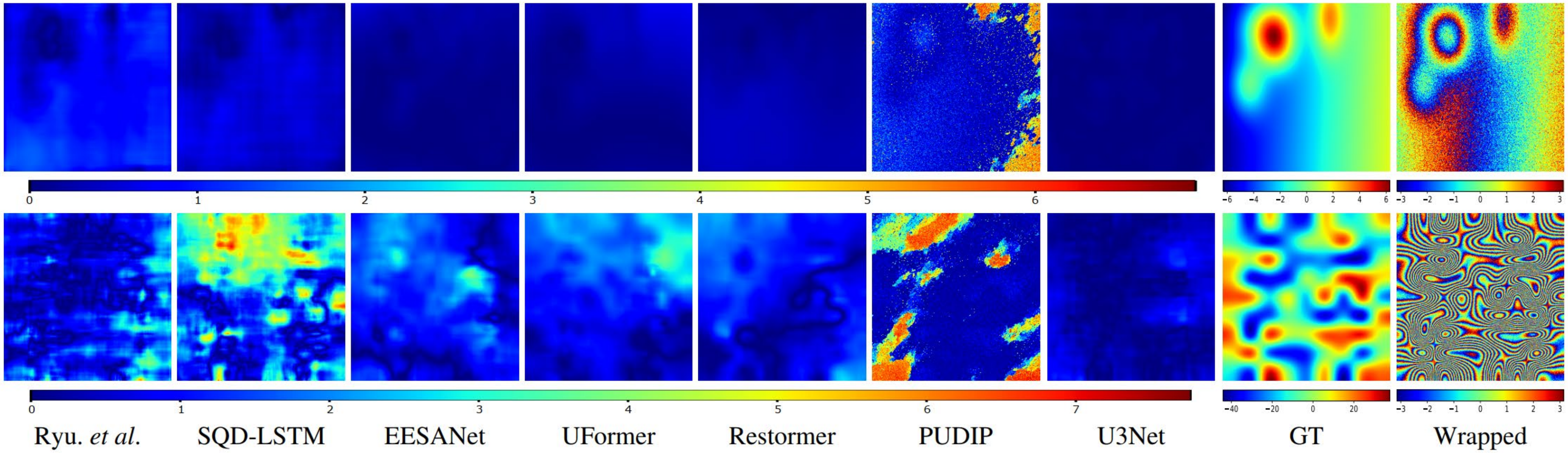
Evaluation on Simulated Phase Patterns

Boldfaced: best results; Underlined: second best results at each column. NRMSE is used for evaluation.

Dataset		MoGR					RME					#Param. (M)	#FLOPs (G)	Time (msec.)
SNR(dB)		0	5	10	20	30	0	5	10	20	30			
Non-Learning	LS	7.91	2.28	1.20	0.37	0.15	7.85	2.50	1.25	0.40	0.12	-	-	8.39
	QGPU	17.12	2.12	1.15	0.36	0.11	17.26	2.29	1.20	0.39	0.12	-	-	14.94
Supervised	Ryu. *et al.*	1.34	1.11	1.09	1.05	1.05	1.51	1.05	1.01	0.98	0.93	1.07	21.85	303.96
	PhaseNet2.0	8.35	8.19	8.07	8.06	8.04	9.29	8.53	7.53	6.95	6.86	1.15	11.93	20.87
	SQD-LSTM	0.87	0.71	0.70	0.69	0.68	1.57	1.13	1.12	1.10	1.09	0.90	4.07	13.04
	EESANet	0.78	0.77	0.76	0.76	0.74	1.31	1.31	1.25	1.24	1.06	61.68	75.27	9.85
	TriNet	5.65	5.55	5.46	5.40	5.34	5.40	5.33	5.21	5.10	5.05	13.61	65.48	11.16
	UFormer	0.50	0.47	0.47	0.46	0.45	0.58	0.51	0.49	0.49	0.48	20.60	40.98	42.00
	Restormer	**0.45**	0.38	0.36	0.36	0.35	**0.50**	0.43	0.42	0.41	0.41	3.02	17.23	44.69
Unsupervised	PUDIP	17.53	15.16	7.87	0.34	0.11	13.10	7.22	2.62	0.38	0.12	2.33	21.58	99695.01
	U3Net	0.69	**0.25**	**0.19**	**0.16**	**0.10**	1.12	**0.38**	**0.27**	**0.17**	**0.12**	0.74	8.77	10.28

Our U3Net achieves the best results in **8/10** settings, using a lightest-weight unrolling network.

Visualization on Simulated Phase Patterns

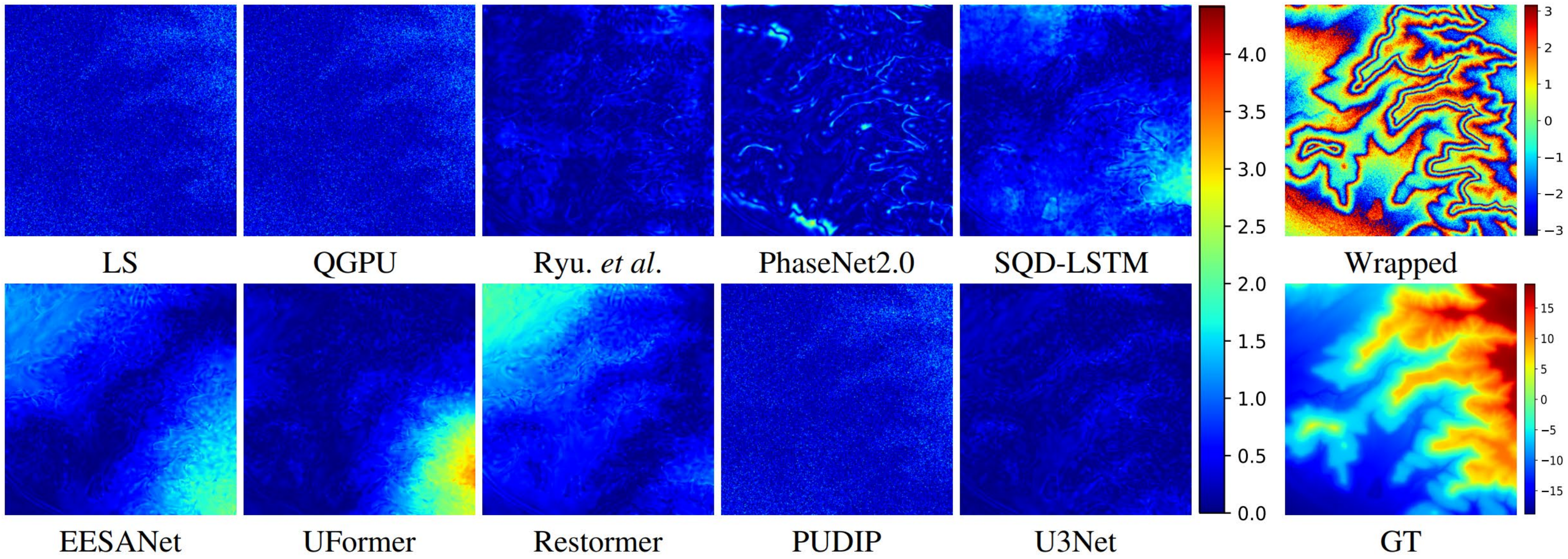


Residual visualizations of PU results on MoGR (top) and RME (bottom)

Our U3Net provides the best residual image results in both datasets.

Evaluation on InSAR Data

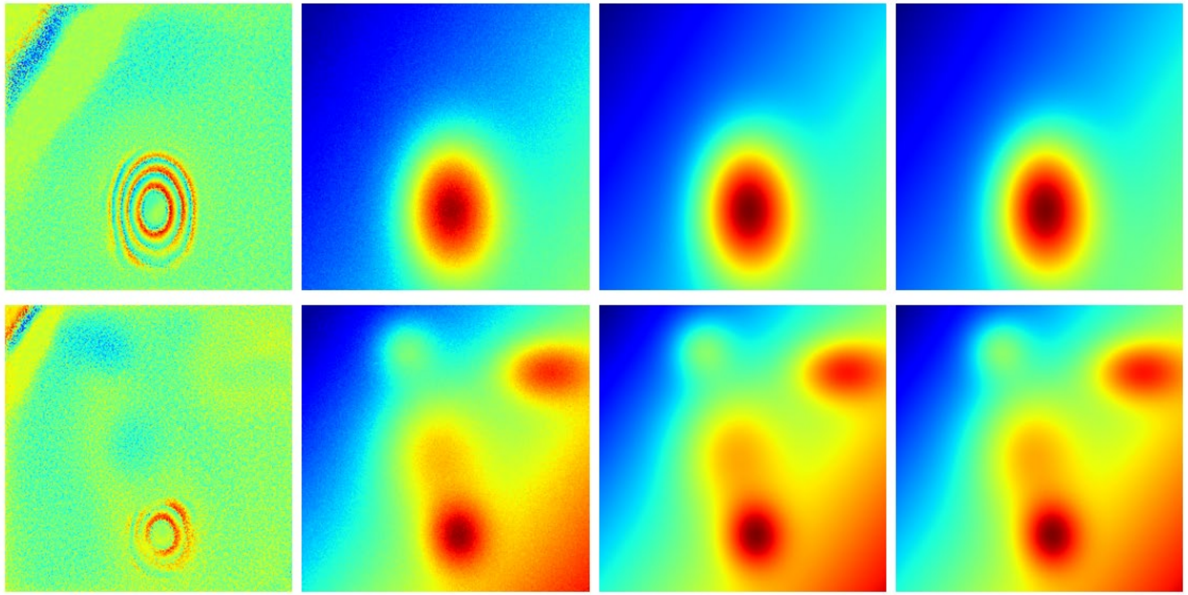
SNR(dB)	LS	QGPU	Ryu. *et al.*	PhaseNet2.0	SQD-LSTM	EESANet	TriNet	PU-GAN	PUNet	UFormer	Restormer	PUDIP	U3Net
5	3.31	3.13	1.41	2.28	1.76	2.45	5.04	13.73	9.59	1.46	<u>1.06</u>	9.99	**1.00**
10	1.84	1.96	1.27	1.69	1.52	1.99	4.69	11.84	9.20	1.28	<u>0.93</u>	5.30	**0.82**
20	0.94	1.13	1.24	1.43	1.48	1.79	4.46	11.62	9.01	0.97	0.91	<u>0.47</u>	**0.46**



Our U3Net ranks the first in all settings and shows minimum residual.

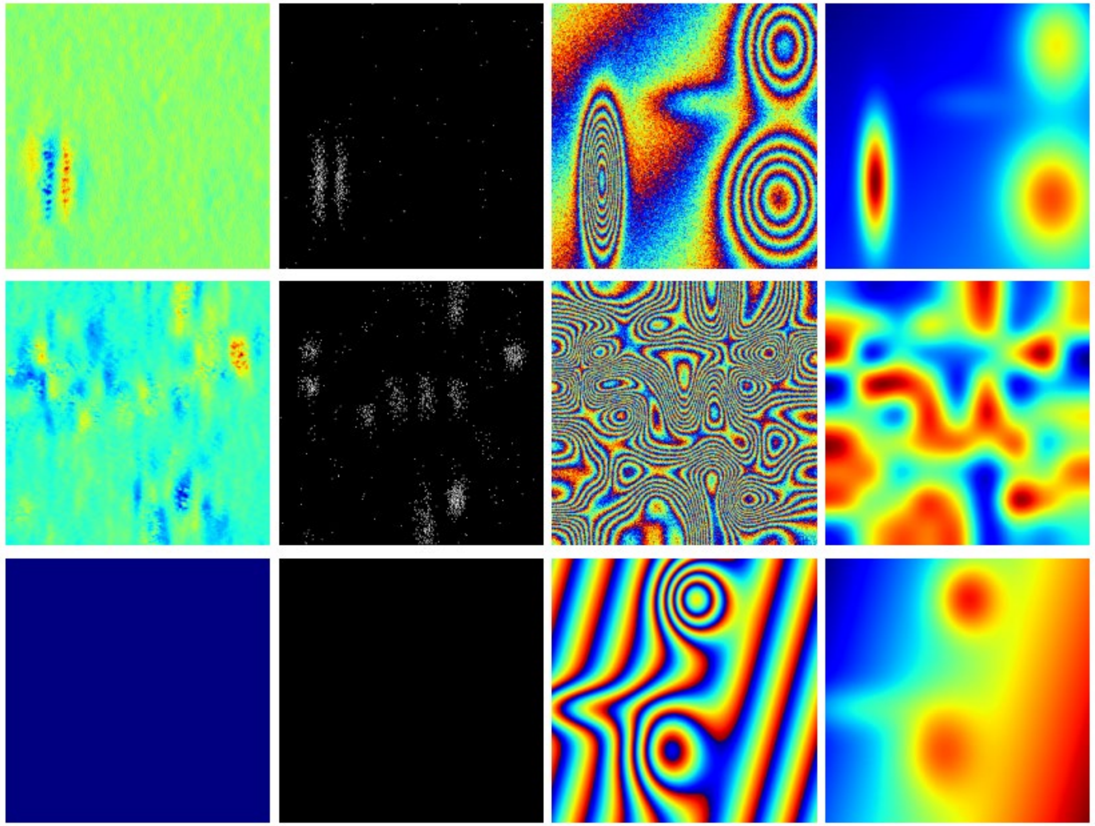
Ablation Analysis

- Loss function



$\mathcal{L}_{sr} \rightarrow \mathcal{L}$ | w/o $\boldsymbol{\nabla U}$ | Original | GT

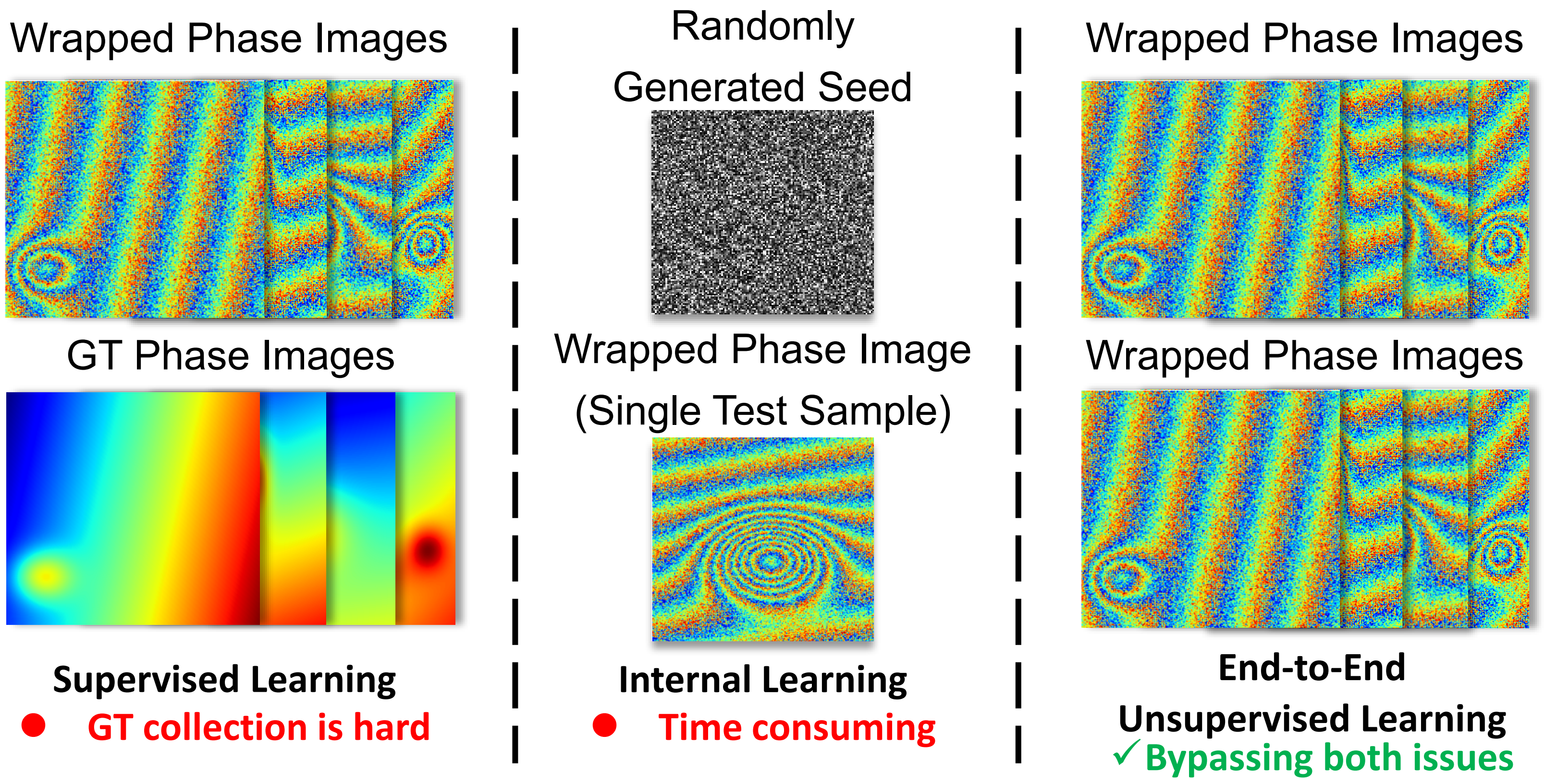
- Visualization of $\boldsymbol{E}$



Error Matrix $\boldsymbol{E}$ | Outlier Map | Wrapped Input | GT

Conclusion and Future Work

- To conclude



☆Our work

- In future
  - Improving the perceiving schemes for outlier points.
  - Enhancing the model robustness to noise inconsistency.

Take home messages

- PU can be solved in unsupervised learning manner by utilizing the gradient or wrapped gradient information of wrapped phase images.
- Well designed physic-encoded NN yields better performance and less complexity.

Thank you

For more, please see https://csyhquan.github.io