Plug-and-Play Mass Mapping with Fast Uncertainty Quantification

Hubert Leterme

GREYC, Ensicaen & CosmoStat, DAp, CEA Paris-Saclay

Joint work with Andreas Tersenov, Jalal Fadili and Jean-Luc Starck

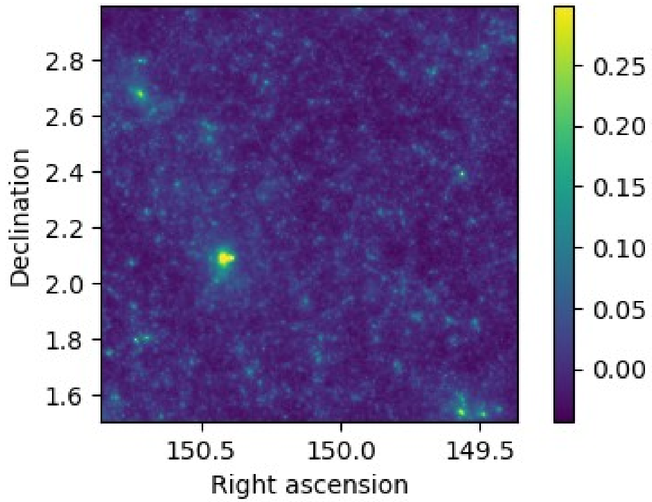
2025 Joint ARGOS-TITAN-TOSCA workshop

FORTH, Heraklion, 07th July 2025

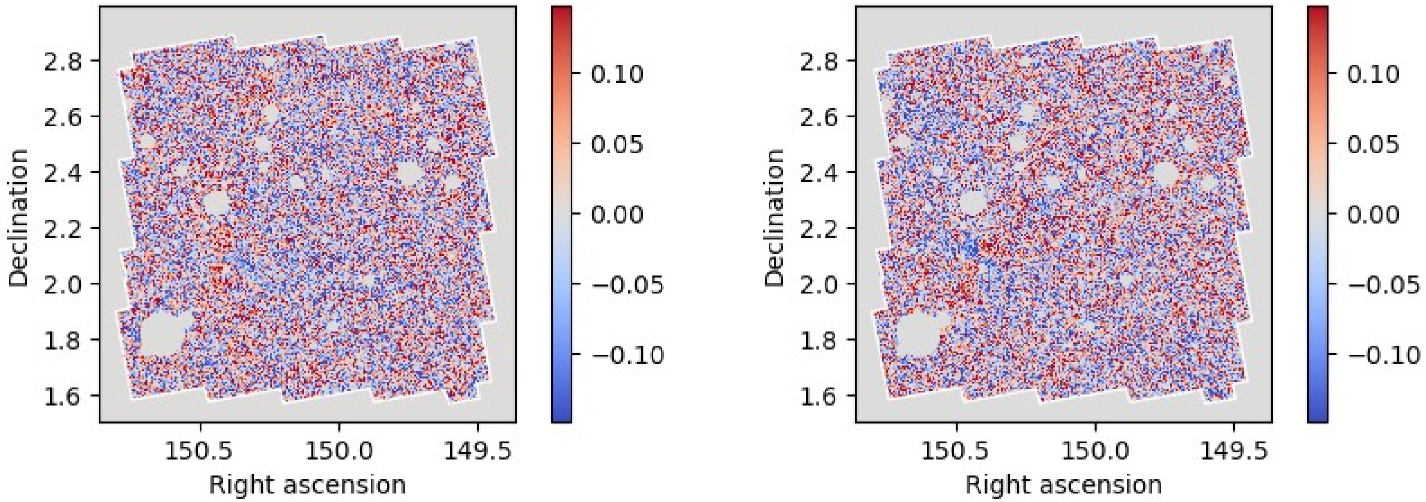
Context and objectives

Example with the κTNG simulated dataset[1]

Simulated convergence map



Corresponding shear map (real and imaginary parts)



- Relation between the noisy shear $\boldsymbol{\gamma}$ (observable) and the convergence $\boldsymbol{\kappa}$ (qty of interest):

$$\boldsymbol{\gamma} = \mathbf{A}\boldsymbol{\kappa} + \boldsymbol{n}$$

  with noise $\boldsymbol{n}$ assumed Gaussian, zero-centered and with diagonal covariance matrix .

- Noise level (standard deviation per pixel): $\boldsymbol{\Sigma}[k,\ k] = \sigma / N_k$
- **Objective**: get a point estimate $\hat{\boldsymbol{\kappa}}$ with error bars, with coverage guarantees.

[1] K. Osato, J. Liu, and Z. Haiman, “κTNG: effect of baryonic processes on weak lensing with IllustrisTNG simulations,” Monthly Notices of the Royal Astronomical Society, vol. 502, no. 4, pp. 5593–5602, Apr. 2021

Related work and proposed approach

	Accurate	Flexible	Fast rec.	Fast UQ
DeepMass	✓	✗*	✓	✓
DeepPosterior	✓	✓	✗	✗
PnPMass (ours)	✓	✓	✓	✓

*Requires specific retraining for each new observation

Proposed approach:

- Based on plug-and-play (PnP) forward-backward splitting.
- Deep denoiser trained on simulated convergence map, corrupted by Gaussian white noise.
- Error bars estimated with an order-2 moment network.
- Coverage guarantees: calibration with conformal predictions.

DeepMass : Jeffrey, N., Lanusse, F., Lahav, O. & Starck, J.-L. MNRAS 492, 5023–5029 (2020)
DeepPosterior: Remy, B. et al. A&A 672, A51 (2023)
Moment networks: Jeffrey, N. & Wandelt, B. D. 2020, in Third Workshop on ML and the Phys. Sc. (NeurIPS 2020)
Conformal predictions: Romano, Y., Patterson, E., & Candes, E. 2019, in NeurIPS

PnP: a fixed point perspective

$0 < \tau < \frac{2}{\|\mathbf{BA}\|}$

step size

noisy shear map (input)

$$T : \boldsymbol{\kappa}' \mapsto F_{\hat{\boldsymbol{\Theta}}}\left[\boldsymbol{\kappa}' + \tau\mathbf{B}(\boldsymbol{\gamma} - \mathbf{A}\boldsymbol{\kappa}')\right].$$

denoising operator
non-expansive
zero-mean outputs

linear operator
$\mathbf{BA} \geq 0$

forward operator

Fixed-point property

T admits a unique fixed point $\hat{\boldsymbol{\kappa}}$, and the series $\hat{\boldsymbol{\kappa}}^{(k+1)} := T\big(\hat{\boldsymbol{\kappa}}^{(k)}\big)$ converges toward $\hat{\boldsymbol{\kappa}}$.

Denoiser training

Hypothesis: $\hat{\boldsymbol{\kappa}} \approx \boldsymbol{\kappa}$. Under this condition, at fixed point:

$$\hat{\boldsymbol{\kappa}} = T(\hat{\boldsymbol{\kappa}}) \approx F_{\hat{\boldsymbol{\Theta}}}(\boldsymbol{\kappa} + \underline{\tau\mathbf{B}\boldsymbol{n}}).$$

Gaussian denoiser

$\boldsymbol{n}_0 \sim \mathcal{N}(\mathbf{0},\, \tau^2\mathbf{B}\boldsymbol{\Sigma}\mathbf{B}^*)$

PnP: a fixed point perspective

$0 < \tau < \frac{2}{\|\mathbf{BA}\|}$

step size

noisy shear map (input)

$$T : \boldsymbol{\kappa}' \mapsto F_{\hat{\boldsymbol{\Theta}}}\left[\boldsymbol{\kappa}' + \tau \mathbf{B}(\boldsymbol{\gamma} - \mathbf{A}\boldsymbol{\kappa}')\right].$$

denoising operator
non-expansive
zero-mean outputs

linear operator
$\mathbf{BA} \geq 0$

forward operator

Noise-whitening operator

Flexibility: We want $\boldsymbol{n}_0$ to be independent of $\boldsymbol{\Sigma}$.

Proposed solution: $\mathbf{B} := \mathbf{A}^*\boldsymbol{\Sigma}^{-1/2}$. Then,

$$\boldsymbol{n}_0 \sim \mathcal{N}(\mathbf{0},\, \tau^2\mathbf{A}^*\mathbf{A}) = \mathcal{N}(\mathbf{0},\, \tau^2).$$

PnP: a fixed point perspective

$0 < \tau < \frac{2}{\|\mathbf{BA}\|}$

step size

noisy shear map (input)

$$T : \boldsymbol{\kappa}' \mapsto F_{\hat{\boldsymbol{\Theta}}}\left[\boldsymbol{\kappa}' + \tau \mathbf{B}({\color{blue}\boldsymbol{\gamma}} - \mathbf{A}\boldsymbol{\kappa}')\right].$$

denoising operator
non-expansive
zero-mean outputs

linear operator
$\mathbf{BA} \geq 0$

forward operator

Analogy with proximal-based optimization

Hyp. (probably wrong): $F_{\hat{\boldsymbol{\Theta}}} = \mathrm{prox}_{g_\tau}$ for some convex reg. function g_τ. Then:

$$\hat{\boldsymbol{\kappa}}^{(k+1)} := \mathrm{prox}_{g_\tau}\left(\hat{\boldsymbol{\kappa}}^{(k)} - \tau \nabla f_{\color{blue}\boldsymbol{\gamma}}(\hat{\boldsymbol{\kappa}}^{(k)})\right), \qquad \text{with} \qquad f_{\color{blue}\boldsymbol{\gamma}} : \boldsymbol{\kappa}' \mapsto \frac{1}{2}\left\|{\color{blue}\boldsymbol{\gamma}} - \mathbf{A}\boldsymbol{\kappa}'\right\|^2_{\boldsymbol{\Sigma}^{-1/2}}.$$

$\Longrightarrow$ *Forward-backward algorithm* [CW05]: the fixed point satisfies:

$$\hat{\boldsymbol{\kappa}} \in \underset{\boldsymbol{\kappa}'}{\mathrm{argmin}}\, f_{\color{blue}\boldsymbol{\gamma}}(\boldsymbol{\kappa}') + g_\tau(\boldsymbol{\kappa}').$$

PnPMass: noise level

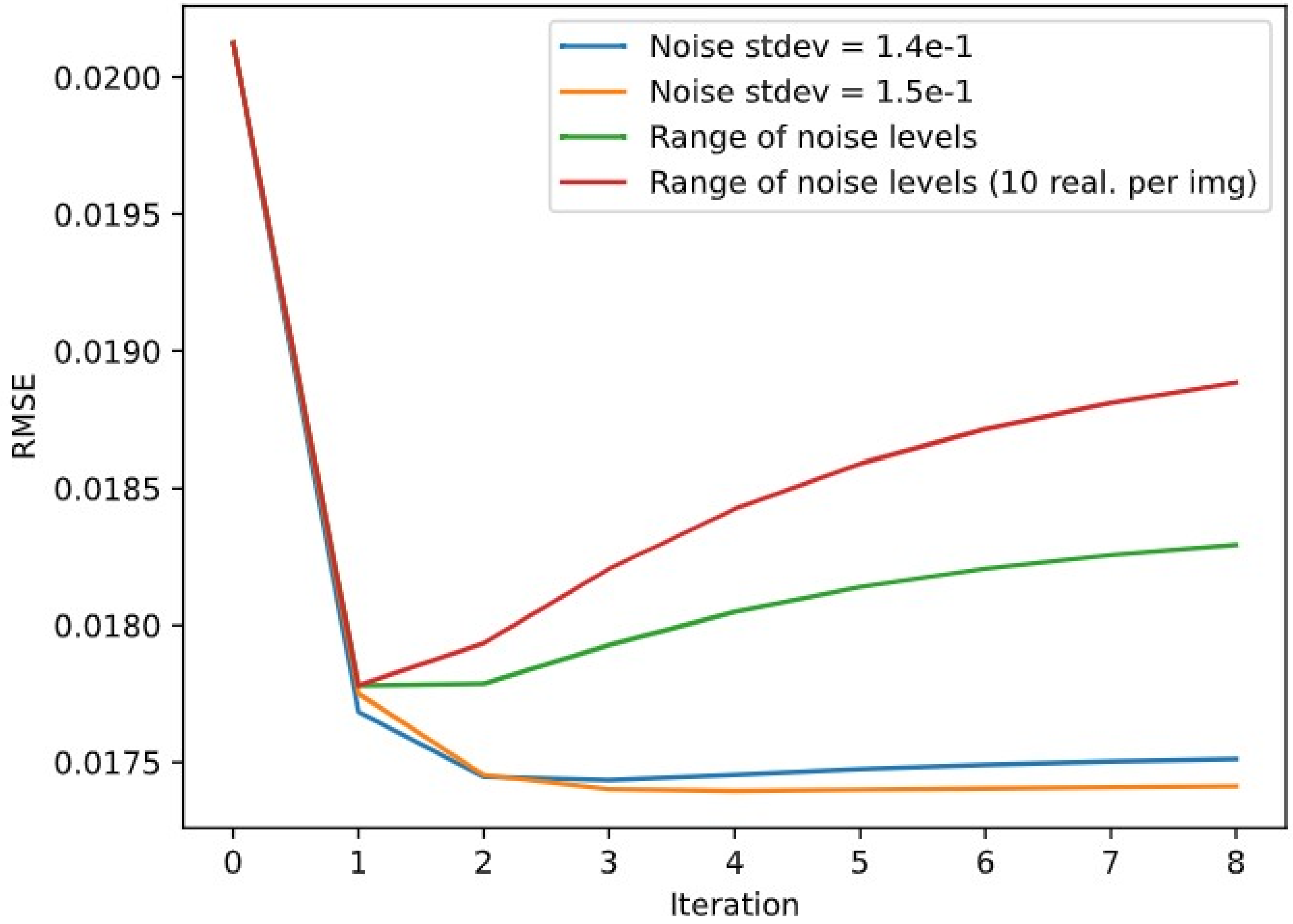
- Noise stdev on which to train the denoiser: equal to the step size.
- However, the step size is conditioned by the noise covariance matrix.

→ **How to keep flexibility?**

→ **Solution:** train the network on a range of noise levels.

→ Converges to a suboptimal fixed point!

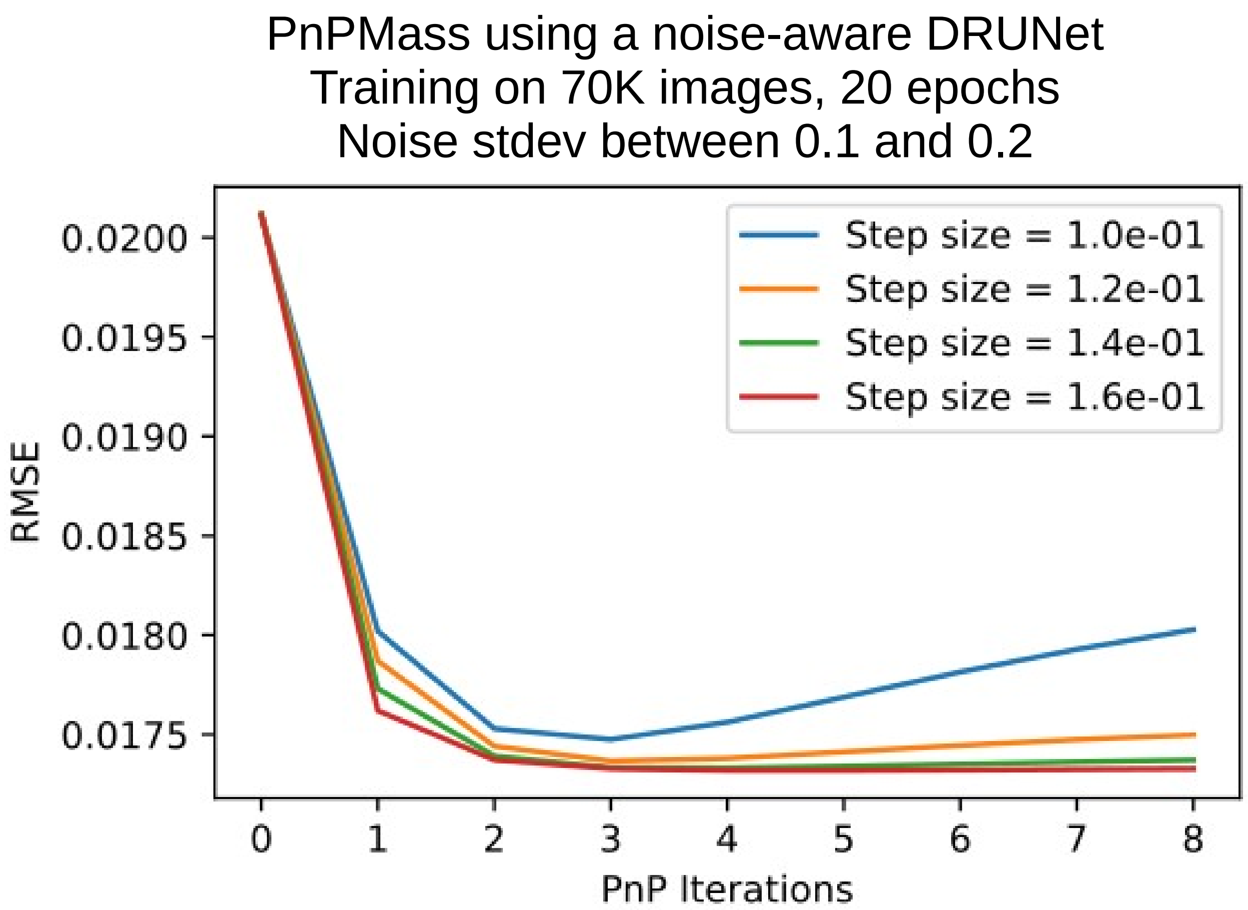
PnPMass using a standard Unet
Training on 70K images, 20 epochs



Range of noise stdev: between 0.1 and 0.2

PnPMass: noise level

→ **Solution:** use a noise-aware model such as DRUNet.

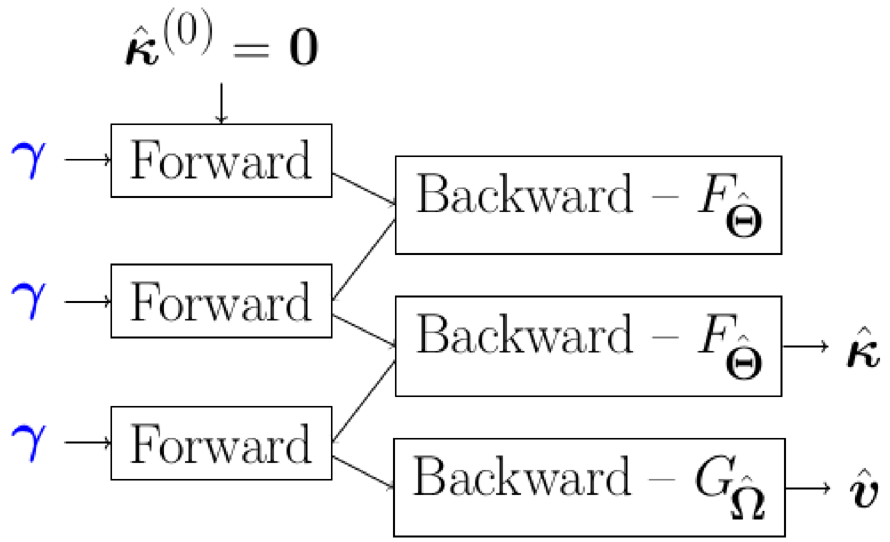


→ Better solutions are obtained for larger step sizes.

Fast uncertainty quantification

Training set $(\boldsymbol{x}_i, \boldsymbol{y}_i)_{i=1}^{n}$, drawn i.i.d. from $(\mathbf{X}, \mathbf{Y})$.

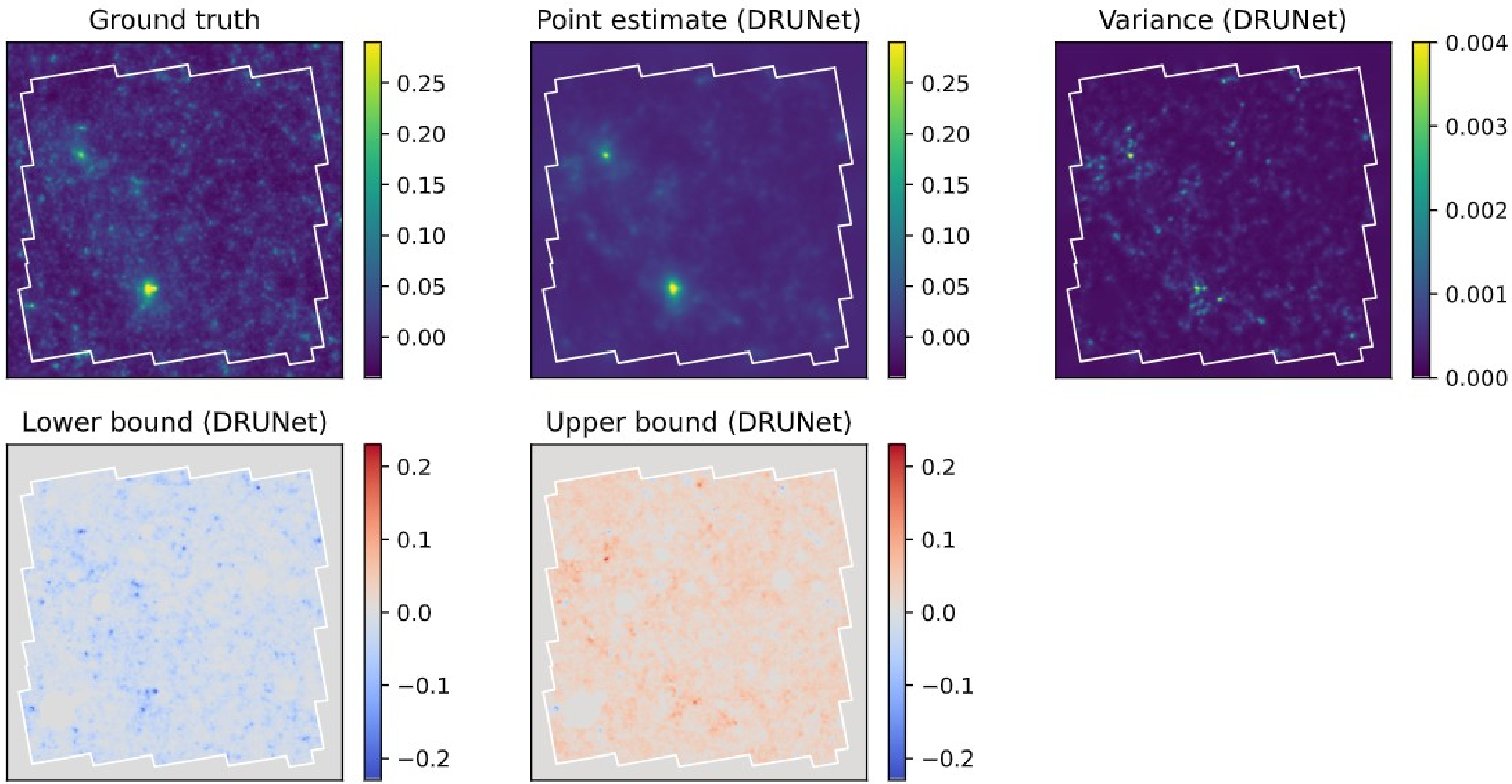
	$F_{\hat{\Theta}}$ (order-1)	$G_{\hat{\Omega}}$ (order-2)
Ground truth	$\boldsymbol{x}_i$	$(\boldsymbol{x}_i - F_{\hat{\Theta}}(\boldsymbol{y}_i))^2$
Approximates	Posterior mean $\mathbb{E}[\mathbf{X} \mid \mathbf{Y} = \boldsymbol{y}_i]$	Posterior variance $\mathbb{V}[\mathbf{X} \mid \mathbf{Y} = \boldsymbol{y}_i]$



Adaptation to PnPMass: one additional iteration for UQ.

Fast uncertainty quantification

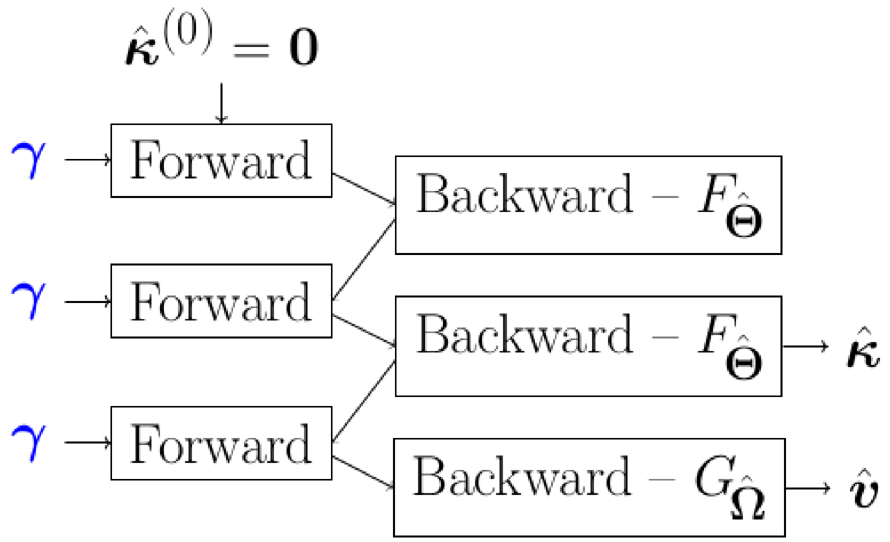
Example : PnPMass run with 3 iterations



Fast uncertainty quantification

Training set $(\boldsymbol{x}_i,\, \boldsymbol{y}_i)_{i=1}^{n}$, drawn i.i.d. from $(\mathbf{X},\, \mathbf{Y})$.

	$F_{\hat{\mathbf{\Theta}}}$ (order-1)	$G_{\hat{\mathbf{\Omega}}}$ (order-2)
Ground truth	$\boldsymbol{x}_i$	$(\boldsymbol{x}_i - F_{\hat{\mathbf{\Theta}}}(\boldsymbol{y}_i))^2$
Approximates	Posterior mean $\mathbb{E}\left[\mathbf{X} \mid \mathbf{Y} = \boldsymbol{y}_i\right]$	Posterior variance $\mathbb{V}\left[\mathbf{X} \mid \mathbf{Y} = \boldsymbol{y}_i\right]$



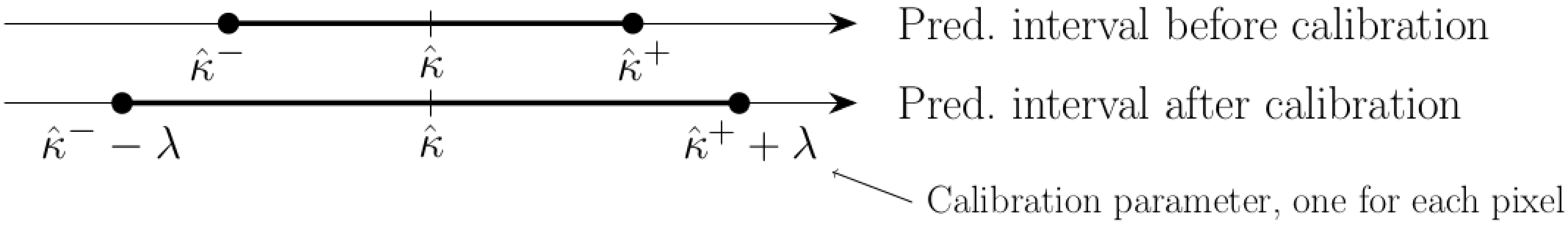
Adaptation to PnPMass: one additional iteration for UQ.

- Non-iterative UQ method, initially designed for end-to-end methods such as DeepMass
- Assumes perfect data knowledge → **aleatoric uncertainty** (irreducible)

→ **Model (or epistemic) uncertainty overlooked** → error bars must be adjusted

→ **Solution:** calibration with conformal prediction

Conformal prediction

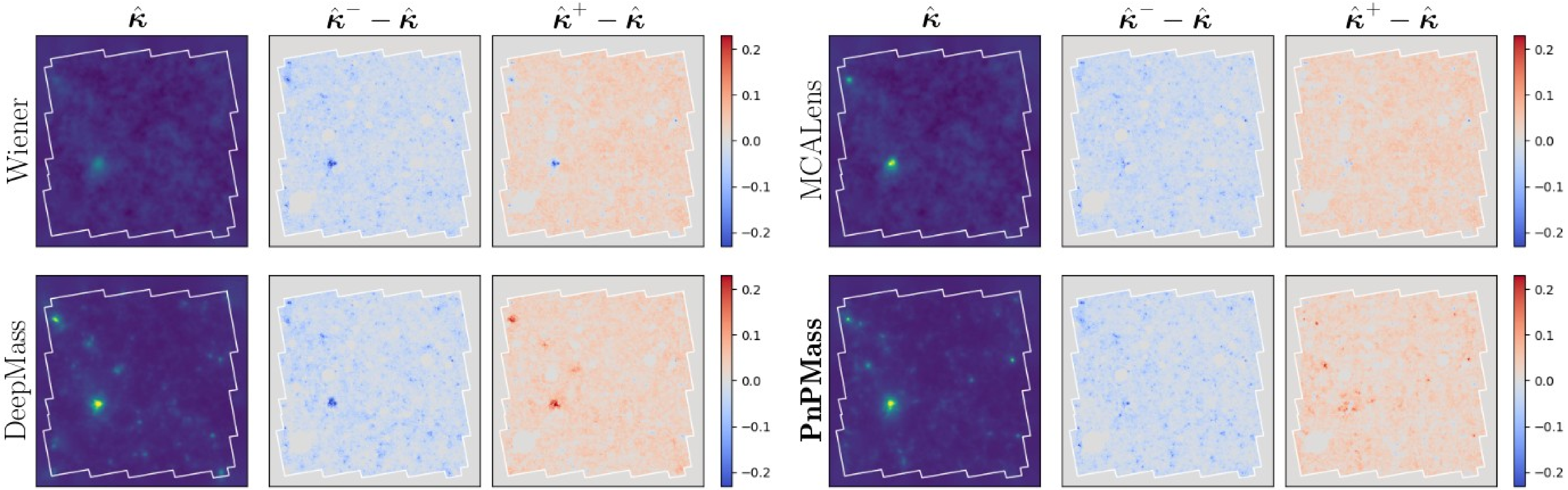


– Based on a simulated **calibration set** $(\boldsymbol{\gamma}_i, \boldsymbol{\kappa}_i)_{i=1}^m$;
– Compute **conformity scores** $\boldsymbol{\lambda}_i$, which depend on α (target **miscoverage rate**).
– Get the $(1-\alpha)(1+1/m)$-th **empirical quantile** of $(\boldsymbol{\lambda}_i)_{i=1}^m$, denoted by $\boldsymbol{\lambda}$.

$$\alpha - \frac{1}{m} \leq \mathbb{P}\left\{\mathbf{K}[k] \notin \left[\hat{\mathbf{K}}^-[k], \hat{\mathbf{K}}^+[k]\right]\right\} \leq \alpha$$

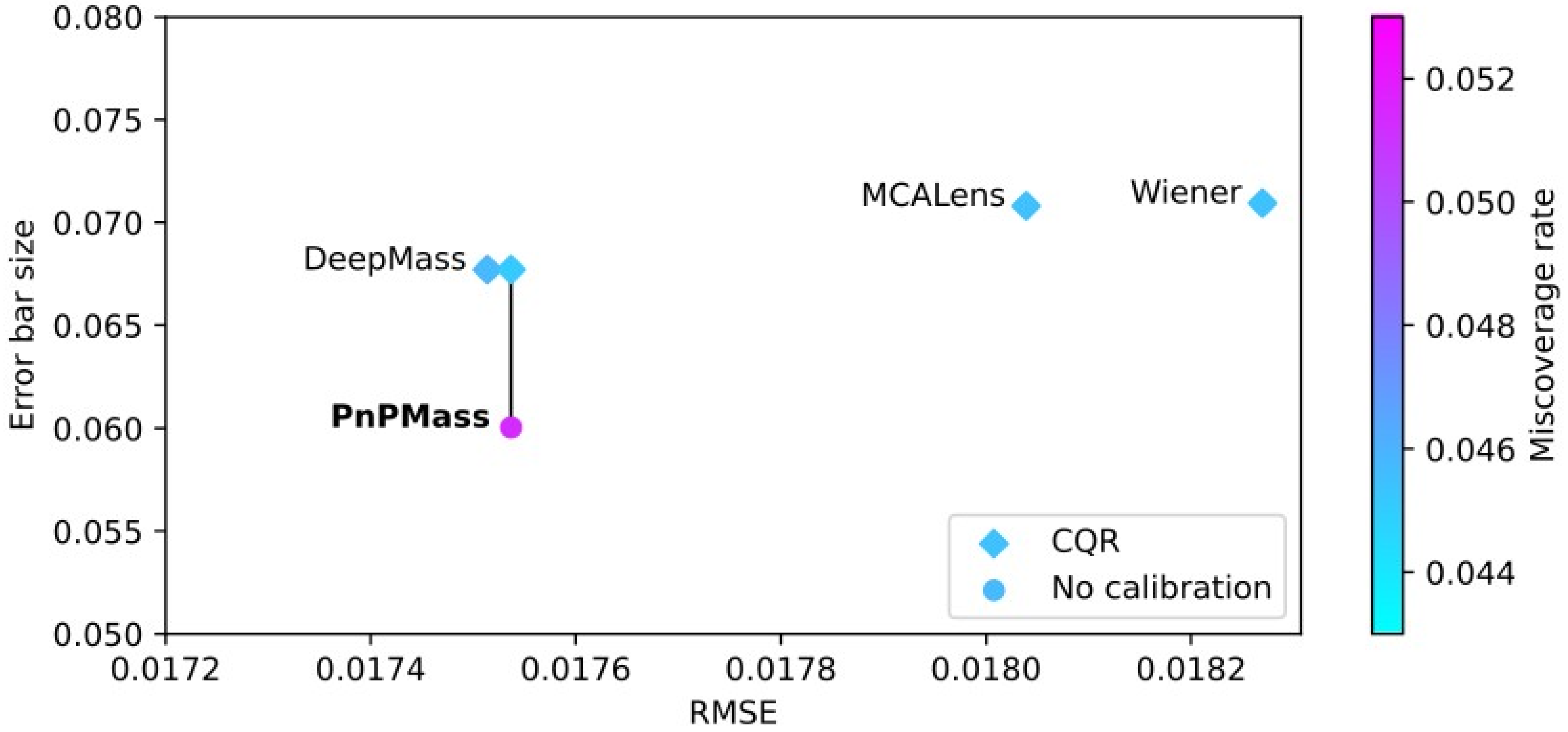
- Target error rate: 4.6% (2-sigma confidence).
- Minimal size for the calibration set: 21 images.
- In our experiments: 100 images.

Benchmark with other approaches



All methods have been calibrated using conformal prediction.

Benchmark with other approaches



- Wiener, MCALens: error bars before calibration initialized to 0.
- DeepMass, PnPMass: error bars before calibration obtained with moment networks.
- DeepMass: miscoverage rate at target (4.6%), even before calibration.
- Results for PnPMass degraded if we train for more noise levels → **increase nb of training epochs**

Robustness to cosmology

- In practice, the true cosmological parameters are unknown. On which dataset to train our model?

→ We need to test the robustness of PnPMass to the choice of cosmology.

- Training on the CosmoSLICS dataset → simulations generated from 25 sets of cosmological parameters.
- Results expected soon (Andreas).

To be continued
Conditional calibration

- Coverage guarantee obtained in average.
- What if we focus on high-density regions (peaks)?

→ No longer valid, even after calibration.

→ **Possible solution to explore:** conditional conformal prediction.

- Note however that calibration wrt to arbitrary conditions is impossible!

Gibbs, Isaac, John J. Cherian, and Emmanuel J. Candès, 2024

To be continued
PnPMass with Wiener initialization

- Wiener, MCALens and DeepMass incorporate physics in their pipeline: mass distributions originate from a Gaussian field, with known power spectrum.
- Can we take advantage of this for PnPMass?

→ **Possible solution:** Wiener initialization, and PnPMass on the residual (non-Gaussian component).

- For training, extracting the non-Gaussian part from the training set may be necessary → no unique solution!
- Investigation in progress.

Conclusion

Contributions:

- Application of PnP-FBS to the mass mapping problem;
- Careful choice of “noise-whitening” data fidelity, allowing flexibility and faster convergence;
- Fast UQ method based on moment networks, adapted to the PnP framework;
- Implementation using PyTorch and DeepInverse

Results:

- Near state-of-the-art results, with more flexibility than related work;
- Caveat: increasing the noise level interval degrades performance.

Work in progress / Future work:

- PnPMass with Wiener initialization;
- Robustness to cosmological parameters using CosmoSLICS simulations;
- Conditional conformal prediction.