

Physics-Aware Deblurring with Coded Exposure

Pedram Yazdinia, Seo Won Yi
Department of Computer Science, University of Toronto

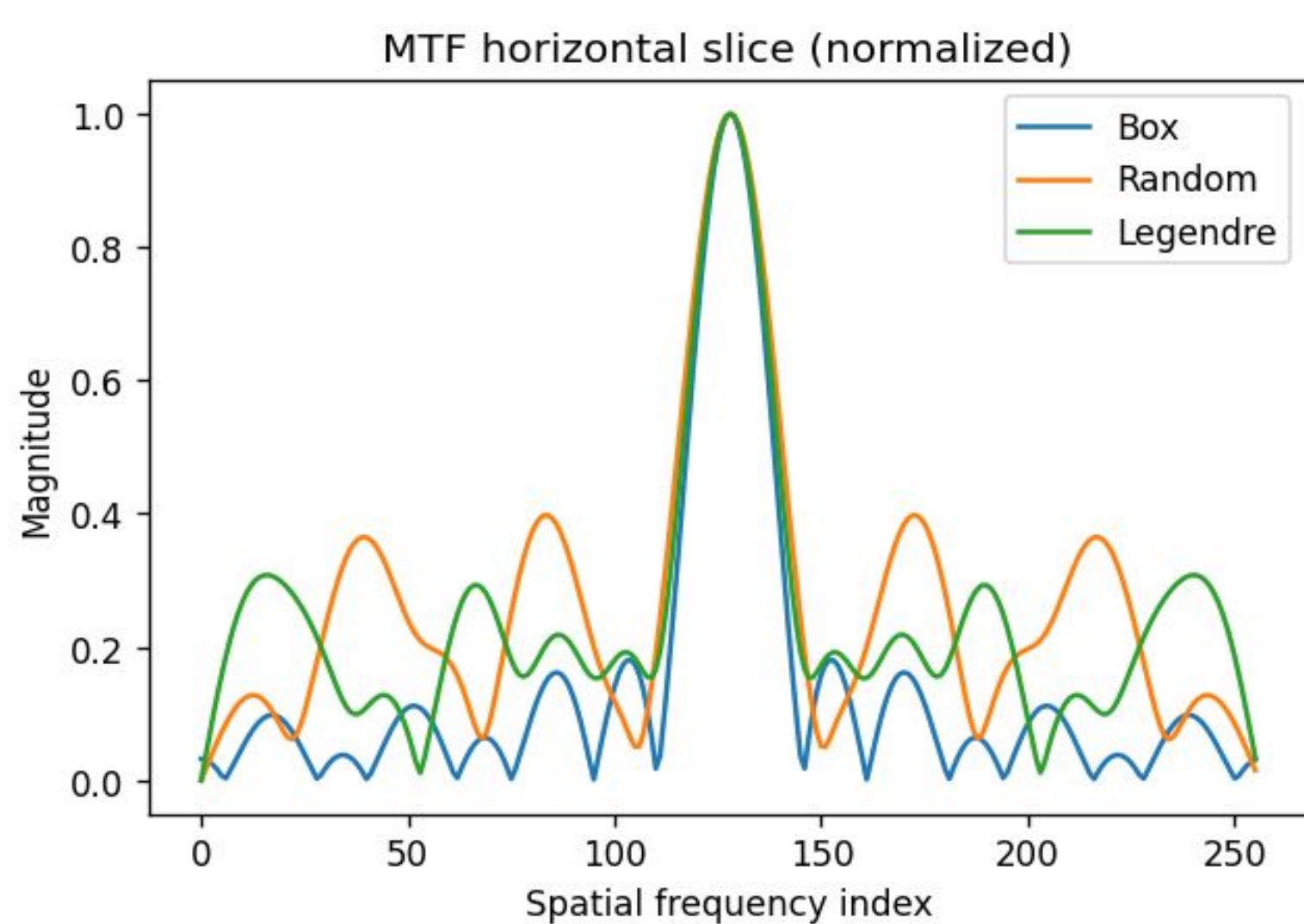
Motivation

Real imaging systems such as smartphones, drones, and robotic cameras often operate under severe motion.

Coded-exposure imaging attempts to preserve recoverable information by modulating the shutter with a temporal code $c(t)$. The observation can be modeled as

$$Y(\omega) = H(\omega)X(\omega) + N(\omega)$$

in the frequency domain. The term $H(\omega)$ is the optical transfer function whose magnitude is the Modulation Transfer Function (MTF). The MTF determines how much contrast survives at each spatial frequency.



* Legendre-coded exposure (green) preserves higher and more uniform MTF than box (blue) or random (orange)

Related Work

We follow the Plug-and-Play ADMM paradigm, coupling a physics-driven solver with a deep prior:

- **The Engine (ADMM):** Iteratively enforces agreement with the coded-exposure forward model and measurement noise statistics.
- **The Deep Prior (PnP):** A pretrained CNN denoiser replaces handcrafted regularizers and restores texture during the iterative updates.

Building on this foundation, prior work has explored key directions relevant to ours:

- Jeon, Lee, and Han introduced a modified Legendre coded-exposure pattern [1] that creates long, low-autocorrelation binary shutter codes for motion deblurring.
- Song *et al.* introduced a quality-aware deblurring model [2] that conditions its restoration on image-quality metrics, conceptually similar to our physics-aware scheduling strategy.

References

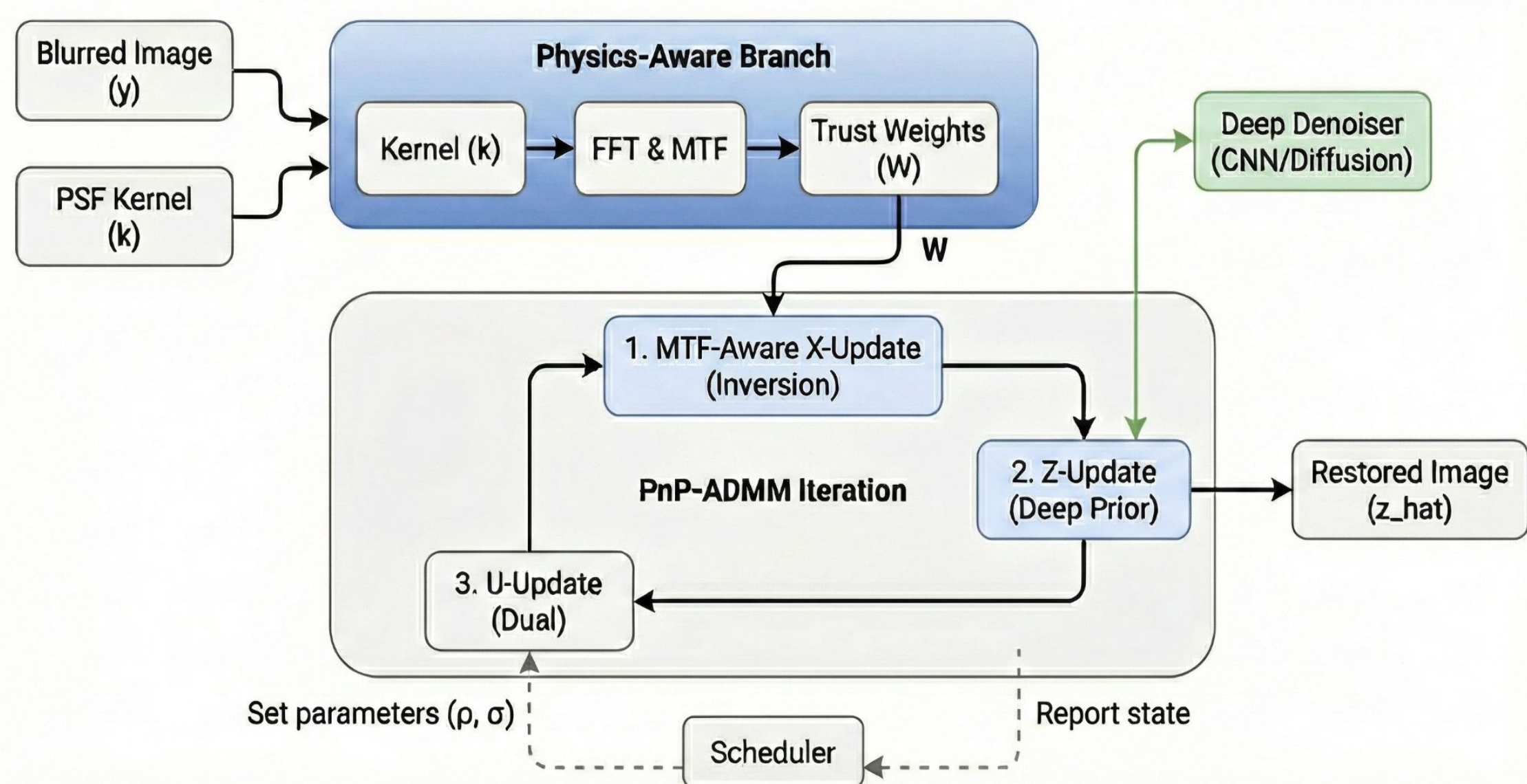
- [1] Jeon, H.-G., Lee, J.-Y., Han, Y., Kim, S.-J., & Kweon, I. S. (2013). *Fluttering Pattern Generation Using Modified Legendre Sequence for Coded Exposure Imaging*. In *Proceedings of the IEEE International Conference on Computer Vision*
- [2] Song, T., Li, L., Wu, J., Dong, W., & Cheng, D. (2024). *Quality-aware blind image motion deblurring*. *Pattern Recognition*, 153, 110568.

New Technique

We inject "optical intelligence" into the standard PnP framework.

- **Physics-Aware Weighting (The Core Novelty)** Pre-computes frequency-domain "Trust Weights" (W) based on the kernel's MTF.
- **MTF-Selective Data Consistency** Modifies the standard ADMM inversion step by injecting Trust Weights.
- **Adaptive Physics Scheduler** Eliminates manual tuning by dynamically adjusting parameters (ρ, σ) at every iteration.

MTF-Aware PnP-ADMM Algorithm



Experimental Results

- We compare our method against classical (**Richardson-Lucy**) and learned PnP baselines (**DnCNN**, **DRUNet**) under identical coded-exposure settings.
- Quantitative evaluation uses **PSNR**, **SSIM**, and **LPIPS** across multiple shutter codes (**Box**, **Random**, **Legendre**).

Comparison of Deblurring Methods (Photon Budget 1000, Blur Length 15px, Read Noise 10 Photons RMS) Across Coded Exposure Types

Method	PSNR (Higher is Better ↑)			SSIM (Higher is Better ↑)			LPIPS (Lower is Better ↓)		
	Box	Random	Legendre	Box	Random	Legendre	Box	Random	Legendre
Richardson-Lucy	19.01	19.00	19.02	0.388	0.387	0.388	0.636	0.619	0.623
Adam w/ DnCNN	22.16	22.78	22.69	0.540	0.544	0.538	0.397	0.372	0.367
ADMM w/ DRUNet	26.84	28.23	28.11	0.735	0.797	0.785	0.256	0.223	0.237
Physics-Aware (Ours)	27.10	28.50	29.55	0.743	0.804	0.853	0.235	0.217	0.211

Comparison of Deblurring Methods (Photon Budget 1000, Blur Length 30px, Read Noise 50 photons RMS) Across Coded Exposure Types

Method	PSNR (Higher is Better ↑)			SSIM (Higher is Better ↑)			LPIPS (Lower is Better ↓)		
	Box	Random	Legendre	Box	Random	Legendre	Box	Random	Legendre
Richardson-Lucy	18.35	18.63	18.63	0.319	0.324	0.325	0.686	0.661	0.665
Adam w/ DnCNN	12.45	17.05	17.08	0.143	0.281	0.282	0.700	0.574	0.572
ADMM w/ DRUNet	10.43	13.48	13.57	0.107	0.147	0.153	0.705	0.585	0.602
Physics-Aware (Ours)	20.79	23.59	24.04	0.541	0.675	0.693	0.539	0.422	0.409

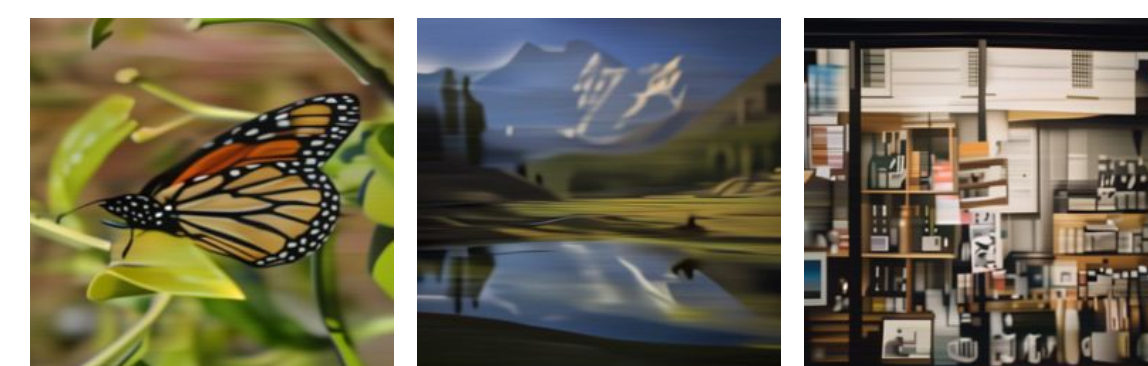
Blur +
Noise



Recon



Left: Physics-Aware Deblurring on mildly blurred and noisy inputs.



Right: Physics-Aware Deblurring on severely blurred and noisy inputs.