List of Publications

Non-Blind Deconvolution

- Sanghvi Yash, A. Gnanasambandam and S. H. Chan, *”Photon Limited Non-Blind Deblurring Using Algorithm Unrolling,”* IEEE TCI 2022

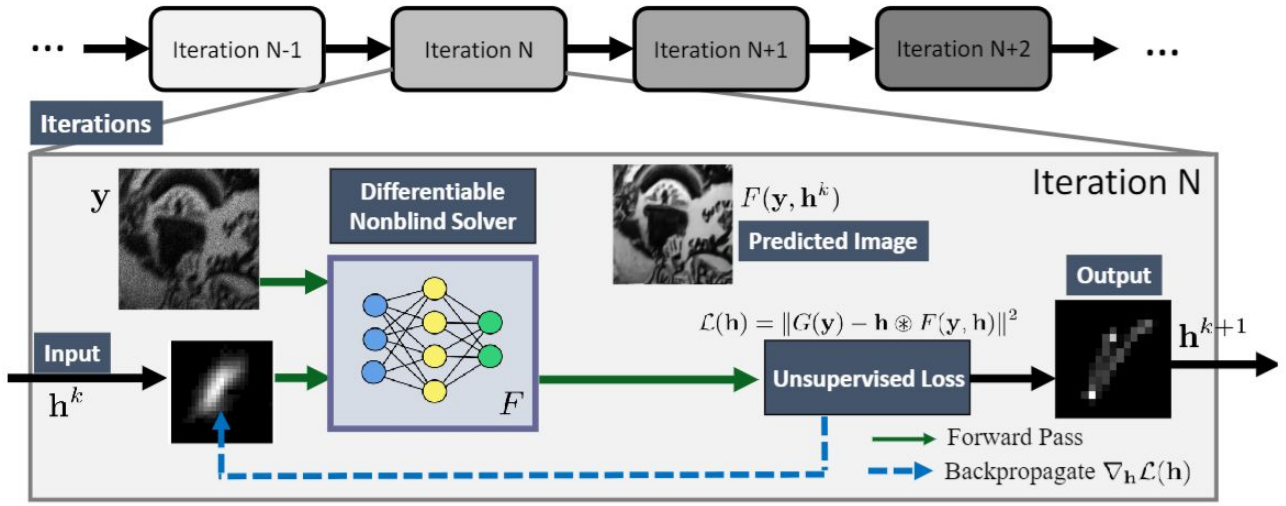
- Gnanasambandam, Abhiram, Y. Sanghvi, and S. H. Chan. *"The Secrets of Non-Blind Poisson Deconvolution."* IEEE TCI 2024.

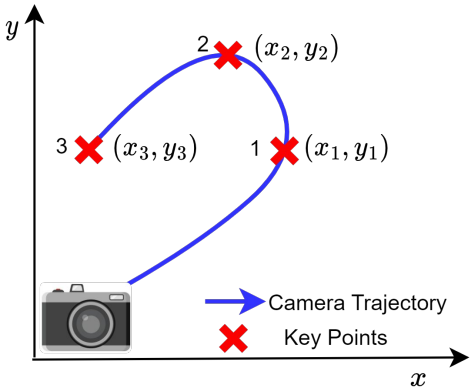
Blind Deconvolution

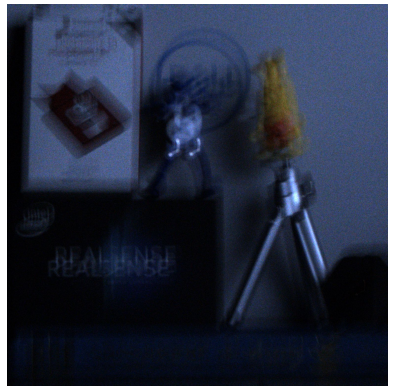
- Sanghvi Yash , A. Gnanasambandam, Z. Mao, and S.H. Chan, *“Photon-Limited Blind Deconvolution Using Unsupervised Iterative Kernel Estimation,”* TCI 2022

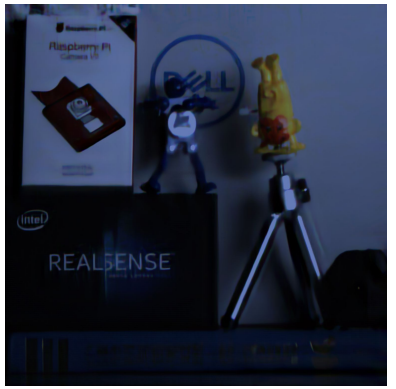
- Sanghvi Yash, Z. Mao, and S. H. Chan. "*Structured Kernel Estimation for Photon-Limited Deconvolution.*" CVPR 2023.

- Sanghvi Yash, Y. Chi, and S. H. Chan. "*Kernel Diffusion: An Alternate Approach to Blind Deconvolution.*" under review at ECCV.

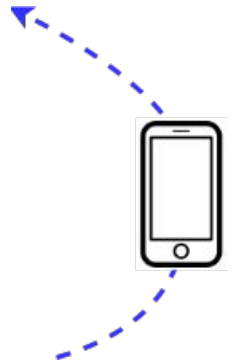








Blind Deconvolution



Camera shake during exposure

+

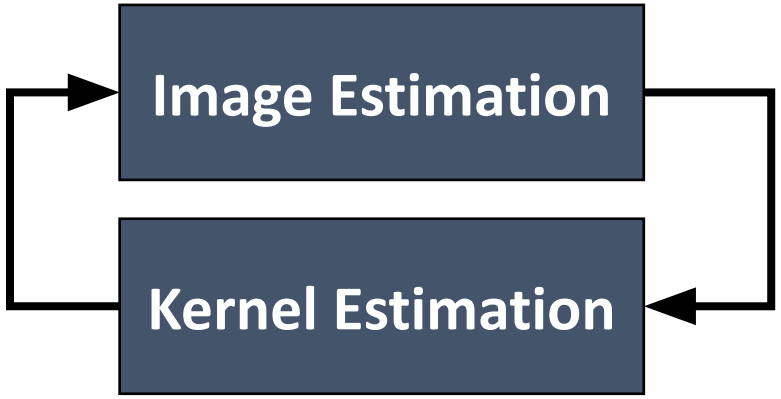
Fronto-Parallel Scene

=

Blurred Image, kernel in inset: $y = k * x$

Traditional Blind Deconvolution [1990s-2015)

- T. F. Chan and C.-K. Wong, *"Total variation blind deconvolution,"* TIP 1998
- S. Cho and S. Lee, "*Fast motion deblurring*," ACM ToG 2009
- Cai, Jian-Feng, et al. *"Blind motion deblurring from a single image using sparse approximation*," CVPR 2009
- L. Xu and J. Jia, *"Two-phase kernel estimation for robust motion deblurring,"* ECCV 2010



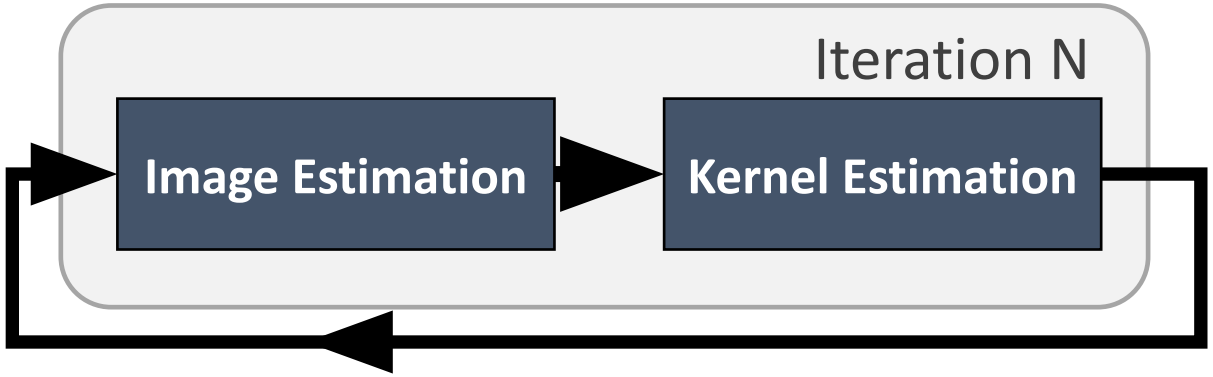
Conventional Alternating Minimization

Main Idea I – Something's not right here

$$\hat{x}, \hat{h} = \text{argmax}\Big[p(x, h|y)\Big]$$

$$\implies \hat{x}, \hat{h} = \text{argmax}\Big[p(y|h, x)p(x)p(h)\Big]$$

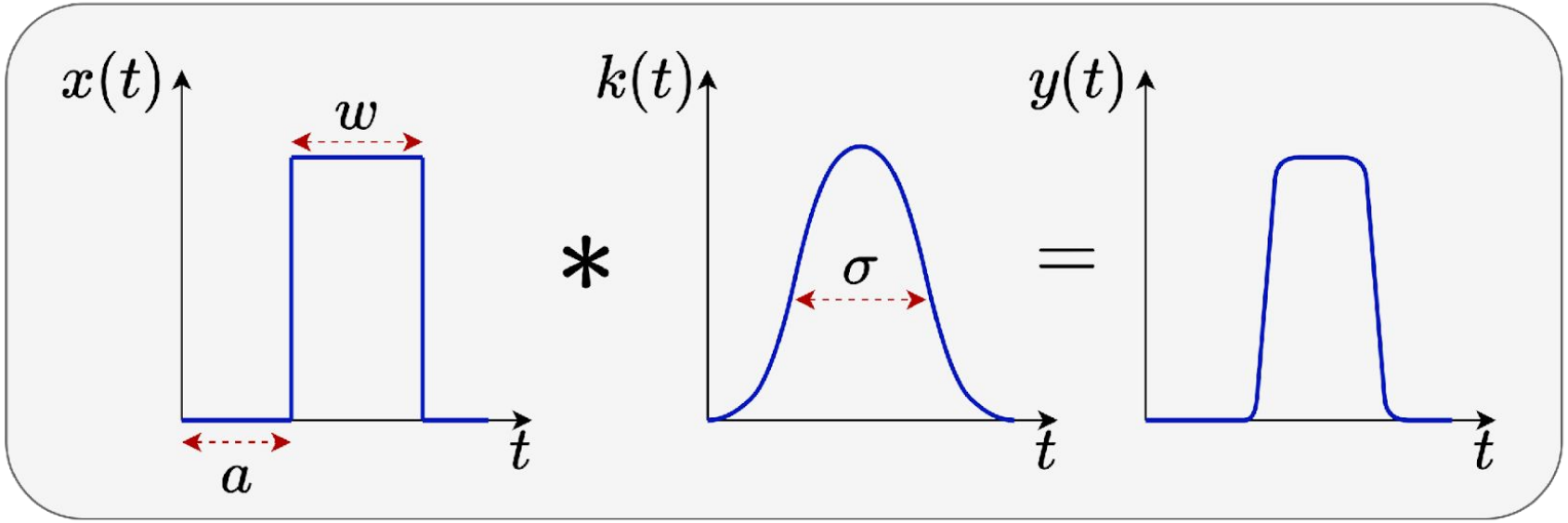
$$\implies \hat{x}, \hat{h} = \text{argmin}\Big[||y - h * x||^2 + \lambda R(x) + S(h)\Big]$$



Conventional alternating minimization can be numerically unstable

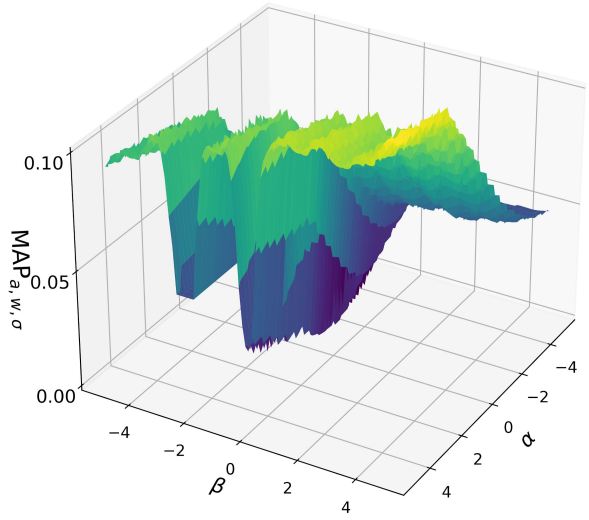
[1] Levin, Anat, et al. "Understanding blind deconvolution algorithms." *IEEE TPAMI 2011*

Main Idea II – A Toy Example



Toy Blind Deconvolution Problem: Estimate a, w, σ from $y(t)$

$$\hat{a}, \hat{w}, \hat{\sigma} = \underset{a,w}{\operatorname{argmin}} \left[||y - x_{a,w} * h_{\sigma}||^2 \right]$$

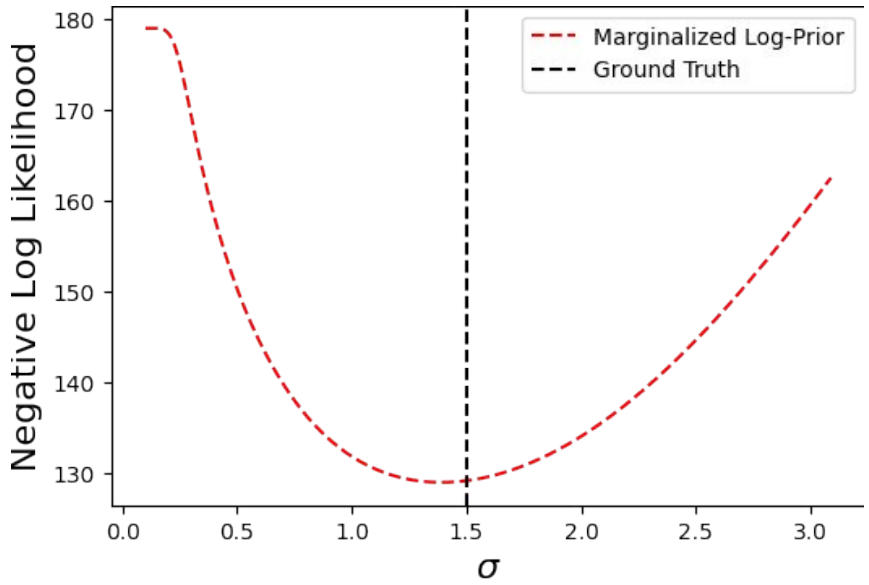


Main Idea III – So what should I do instead?

$$\hat{h} = \operatorname{argmax}\Big[p(h|y)\Big]$$

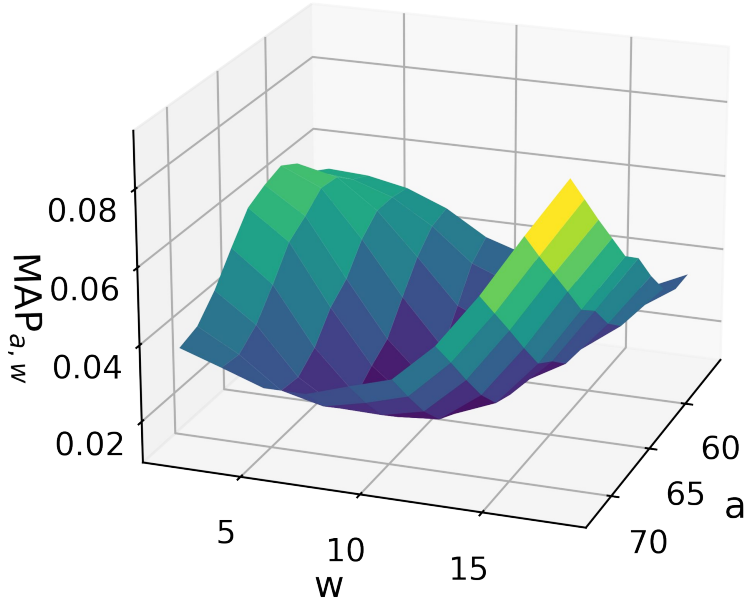
Fewer unknowns

$$\hat{\sigma} = \operatorname{argmax}\Big[\sum_{a,w} \exp\Big[-\frac{1}{2\beta}||y - x_{a,w} * h_{\sigma}||^2\Big]\Big]$$

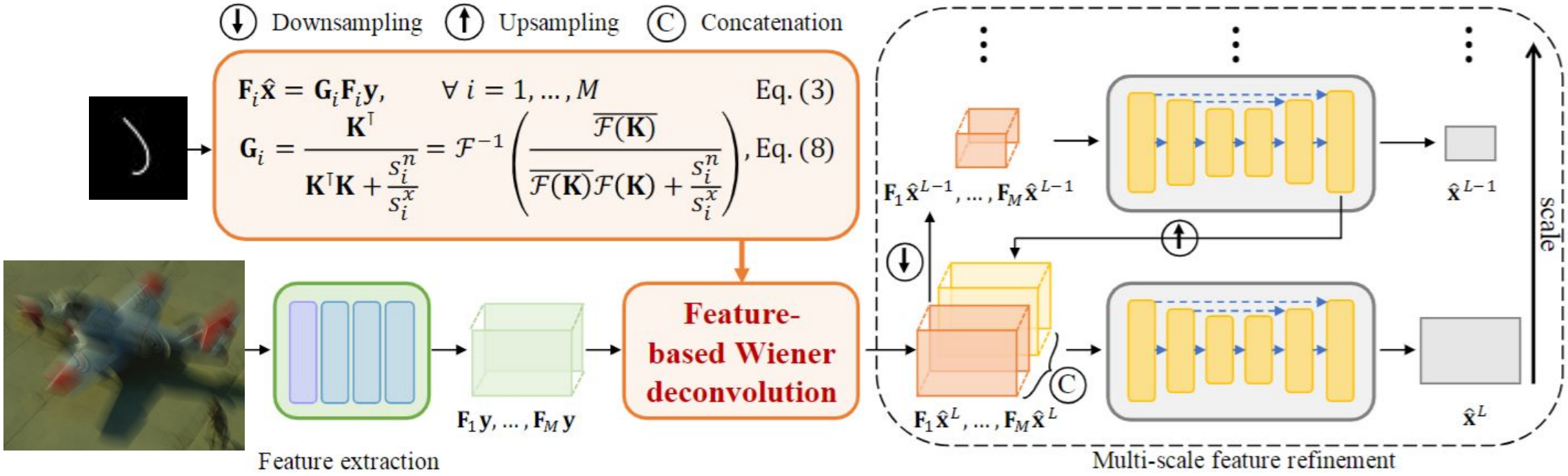


$$\hat{x} = \operatorname{argmax}\Big[p(x|y,\hat{h})\Big]$$

$$\hat{a},\hat{w} = \operatorname{argmax}_{a,w}\Big[||y - x_{a,w} * h_{\hat{\sigma}}||^2\Big]$$



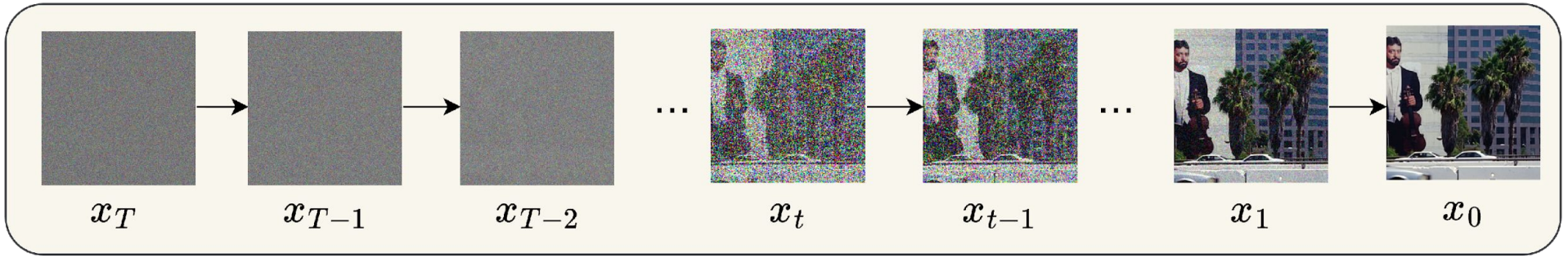
Core Engine of our Methods

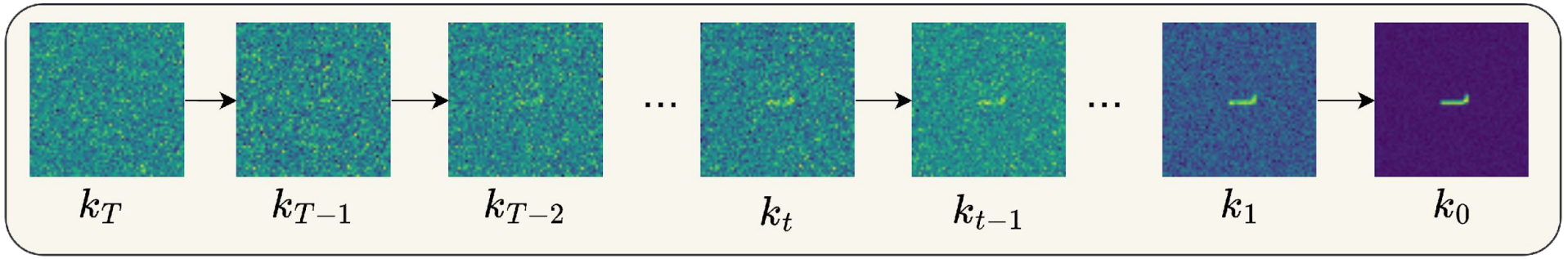


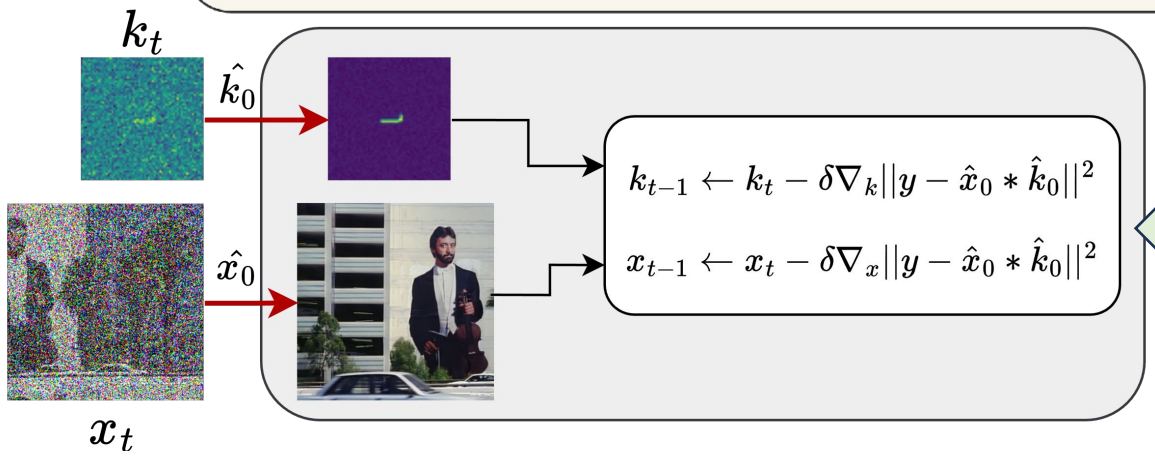
Non-Blind Solver: *differentiable* with respect to inputs

Dong, Jiangxin, et. al. "Deep wiener deconvolution: Wiener meets deep learning for image deblurring." *NeurIPS* 2020.

Blind Deconvolution with Diffusion Models







Sampling from $p(x, h|y)$ using alternating minimization

After each reverse diffusion step

Chung, Hyungjin, et al. "Parallel diffusion models of operator and image for blind inverse problems." *CVPR 2023*

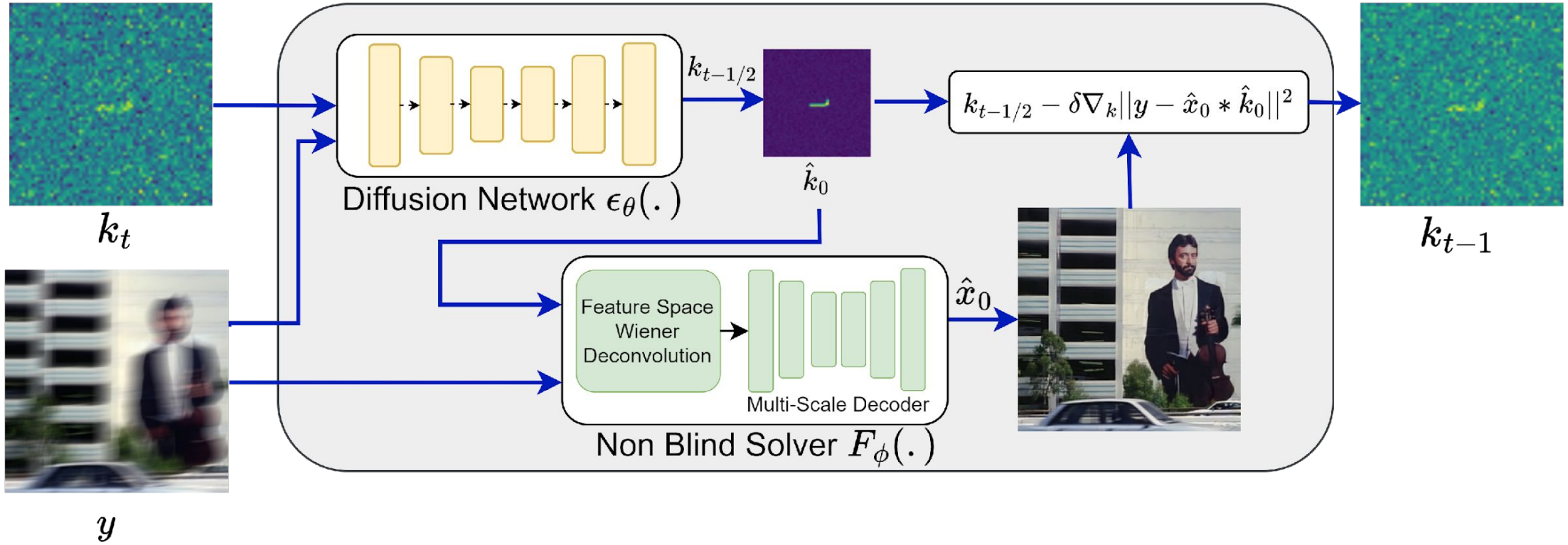
Kernel Estimation Approach

Laplace approximation - $\hat{x}_h$ represents the peak of the conditional distribution $p(x, h|y)$

$$p(h|y) = \int p(x, h|y)dx$$

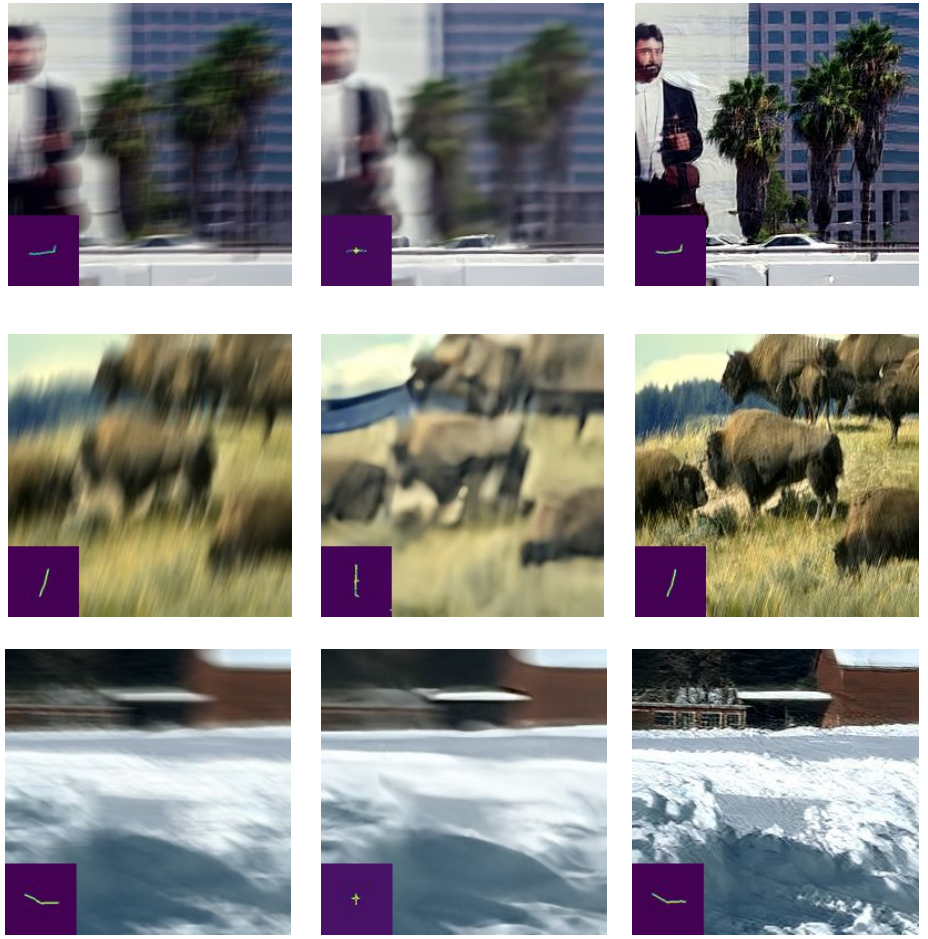
$$\approx p(\hat{x}_h, h|y) \int \exp\left[-\frac{1}{2}(x - \hat{x}_h)S^{-1}(x - \hat{x}_h) \right] dx$$

$$\approx \frac{1}{Z} p(\hat{x}_h, h, |y)$$



Comparison with Blind-DPS

Metric	Blind-DPS (Alternating Estimation)	Kernel-Diff (Ours) (Kernel-First Estimation)
PSNR □	17.56	19.07
SSIM □	0.387	0.500
LPIPS □	0.583	0.355
FID □	280.53	172.33
Total Params	110M	43M
Time (in s)	337	305



Blurred image | Blind-DPS | Kernel-Diff