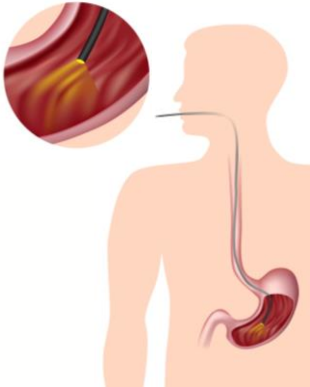
Hyperparameter Optimization in Black-box Image Processing using Differentiable Proxies

Ethan Tseng, Felix Yu, Yuting Yang, Fahim Mannan, Karl St. Arnaud, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide

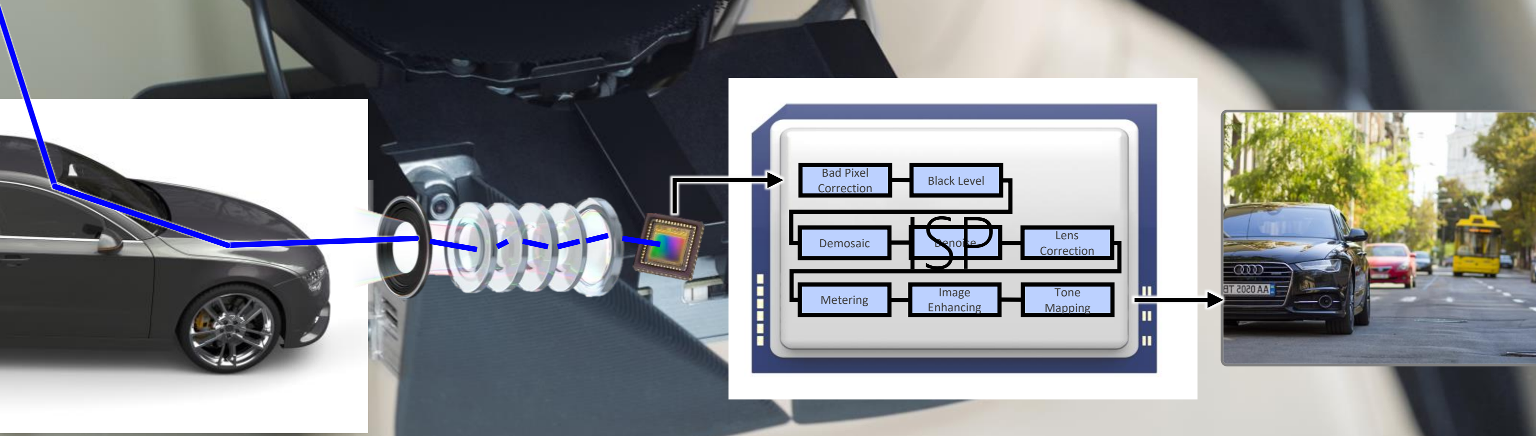
Princeton University, Algolux, McGill University, Université Laval

ACM SIGGRAPH 2019

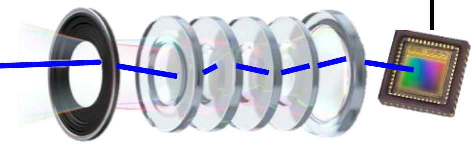
Proprietary Image Processing Pipelines are ubiquitous

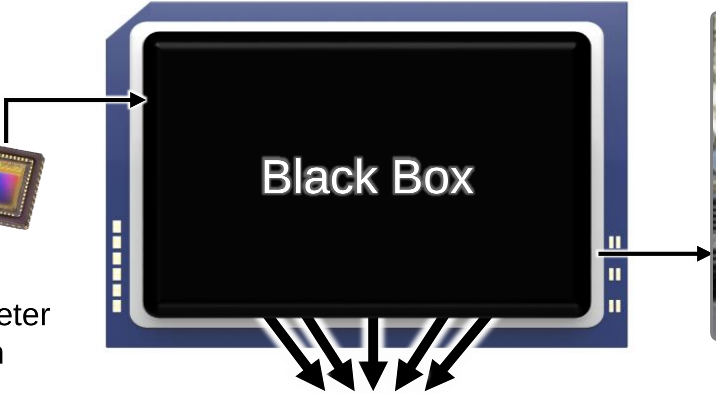


Typical Imaging Stack

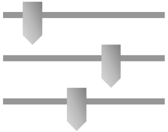


Difficulties and Constraints of ISPs





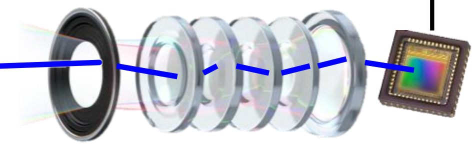
Parameters ?

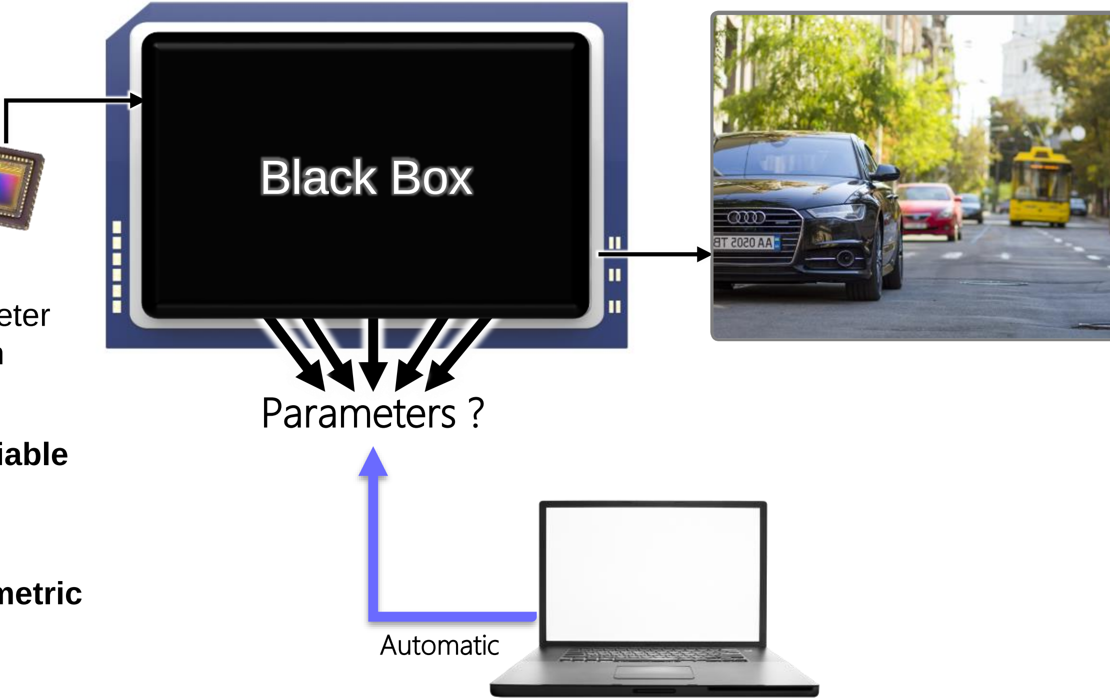


Outputs?

- **High-dimensional** parameter space and **mixed** between continuous and discrete
- ISP is often **non-differentiable** and **black box**
- Want to tune parameters towards almost **any task metric**

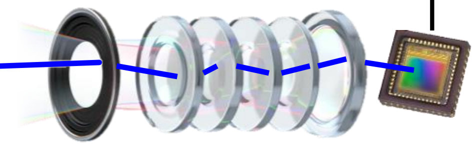
Automating ISP parameter optimization

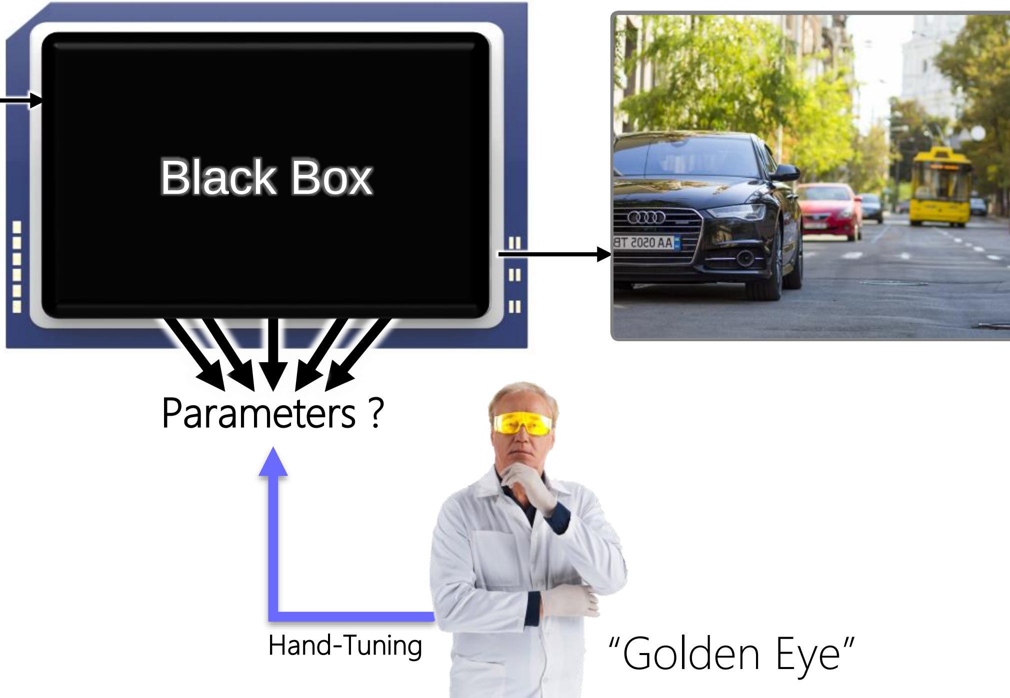




- **High-dimensional** parameter space and **mixed** between continuous and discrete
- ISP is often **non-differentiable** and **black box**
- Want to tune parameters towards almost **any task metric**

Current Industry Approach



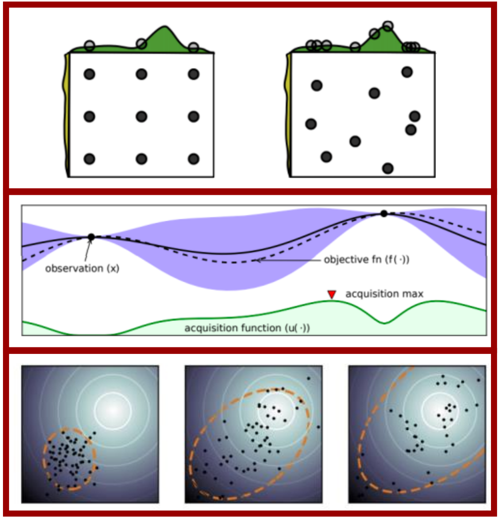


- **High-dimensional** parameter space and **mixed** between continuous and discrete
- ISP is often **non-differentiable** and **black box**
- Want to tune parameters towards almost **any task metric**

Related works: Optimization

- **High-dimensional** parameter space and **mixed** between continuous and discrete
- ISP is often **non-differentiable** and **black box**
- Want to tune parameters towards almost **any task metric**

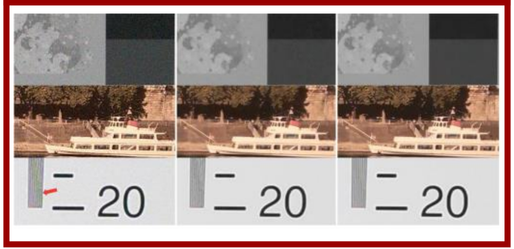
0th order Optimization



[Bergstra12, Ruben14, Bergstra13, Shahriari16, Snoek12, Swersky13, Hansen03]

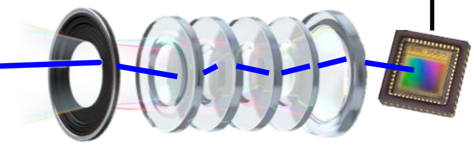
Nonlinear Optimization

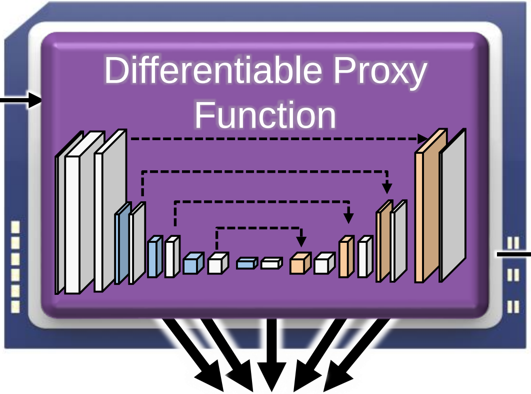
$$B_{k+1}(\mathbf{x}_{k+1} - \mathbf{x}_k) = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k).$$



[Davidson91, Bonnans06, Nelder65, Nishimura18]

Parameter Optimization using Differentiable Proxies





Parameters ?

- **High-dimensional** parameter space and **mixed** between continuous and discrete
- ISP is often **non-differentiable** and **black box**
- Want to tune parameters towards almost **any task metric**

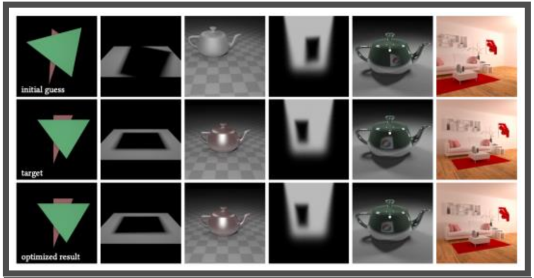
Related works: Neural Networks in ISP Domain

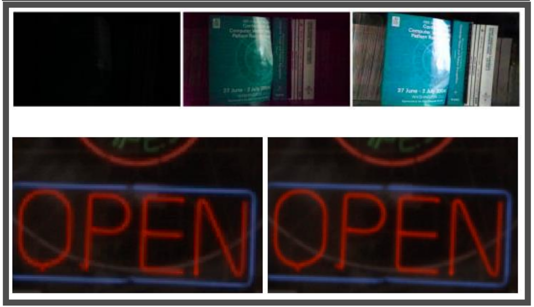
Existing work goals:

- Mimicking ISPs
- Creating new ISPs

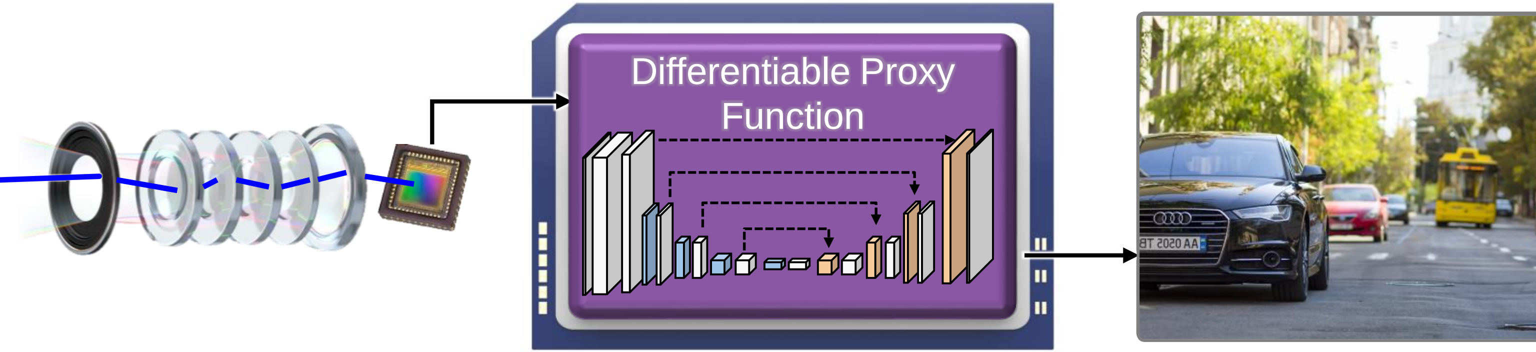
Our work goals:

- Optimize ISP parameters using neural networks





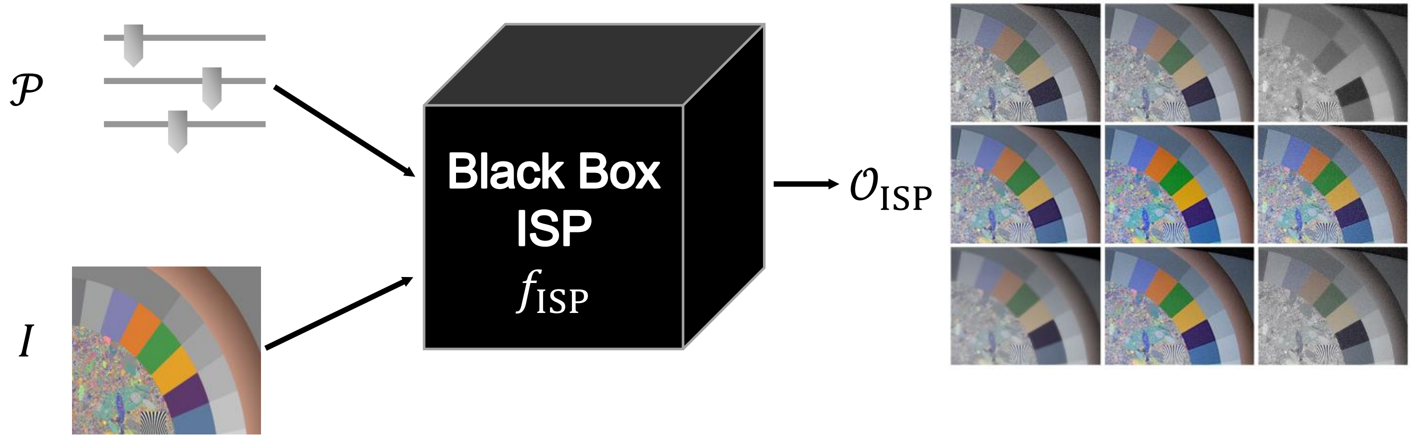
[Li18, Chen18, Gharbi18, Chaitanya17, Liu19]



Stage 1: Learning the Differentiable Proxy Function

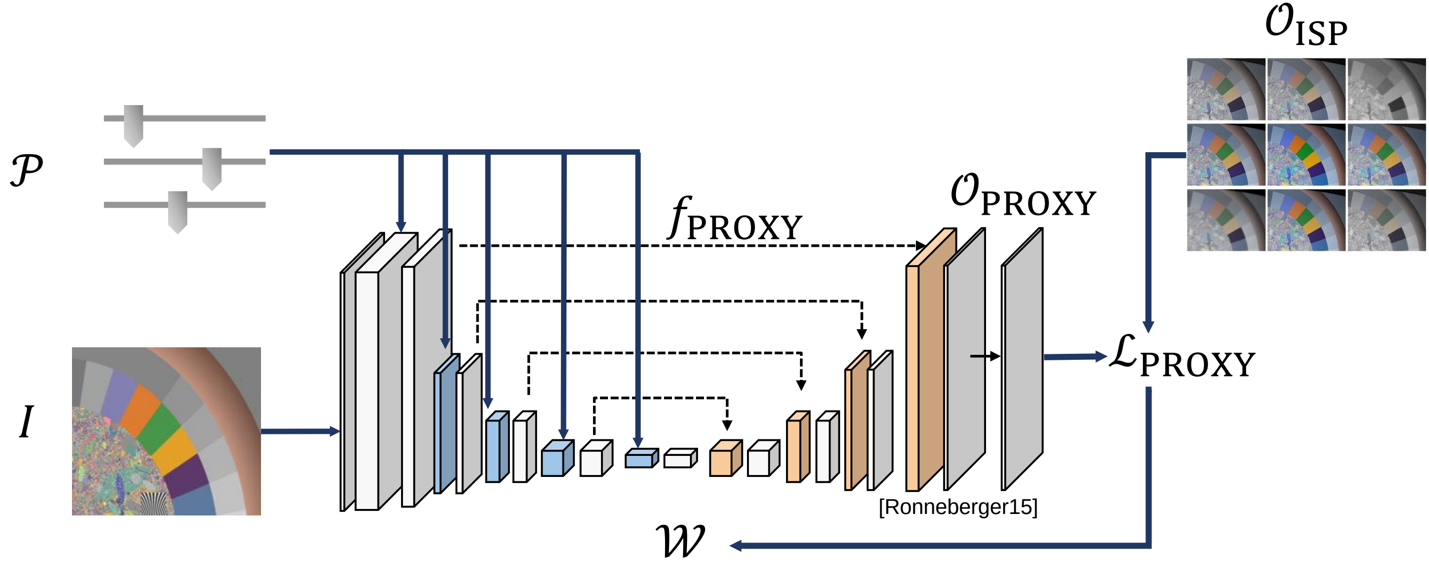
Stage 2: Optimizing Hyperparameters for Task-Specific Outputs

Stage 1: Learning the Differentiable Proxy Function



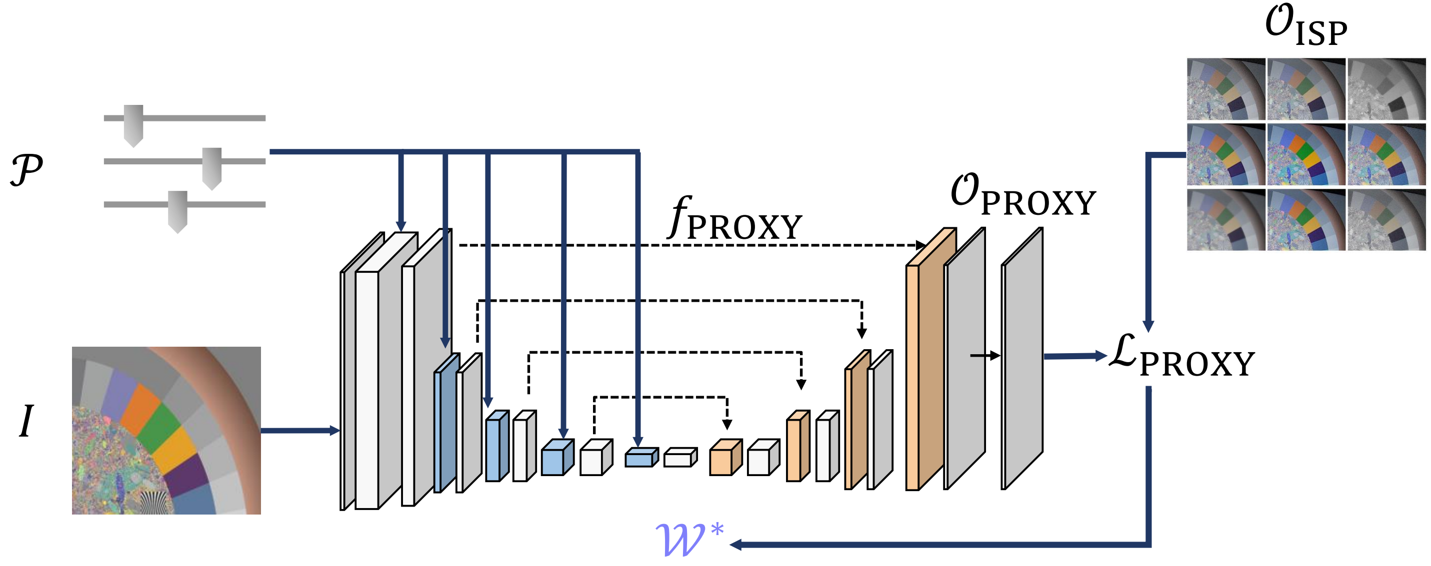
$$\mathcal{O}_{\mathrm{ISP}} = f_{\mathrm{ISP}}(I, \mathcal{P})$$

Stage 1: Learning the Differentiable Proxy Function

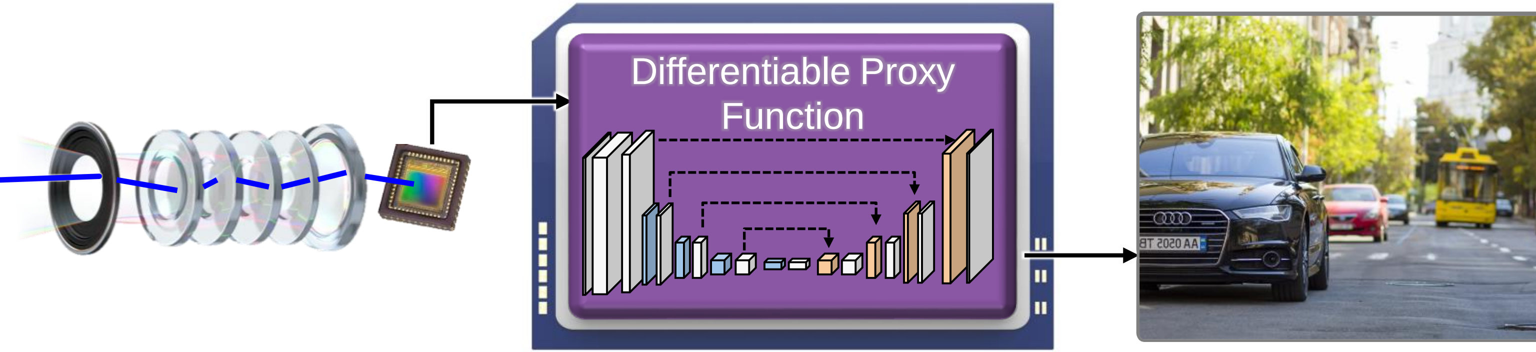


$$\mathcal{O}_{\text{PROXY}} = f_{\text{PROXY}}(I, \mathcal{P}, \mathcal{W})$$

Stage 1: Learning the Differentiable Proxy Function



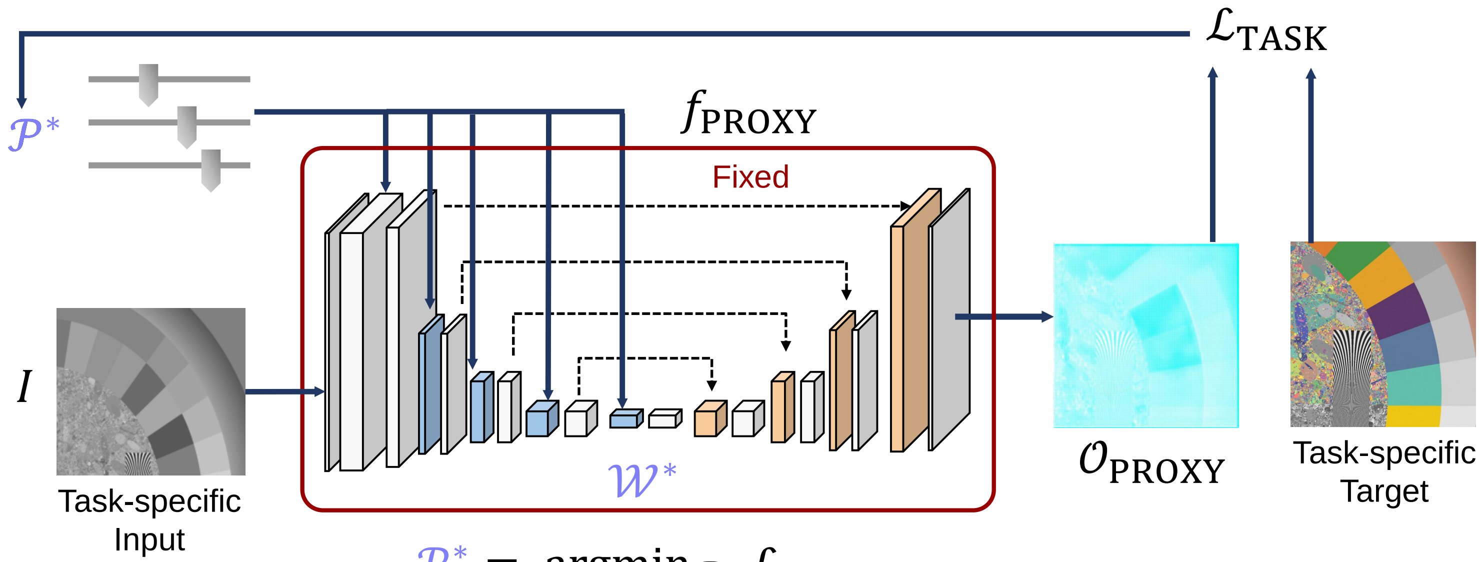
$$f_{\mathrm{PROXY}}(I, \mathcal{P}, \mathcal{W}^*) \approx f_{\mathrm{ISP}}(I, \mathcal{P})$$



Stage 1: Learning the Differentiable Proxy Function

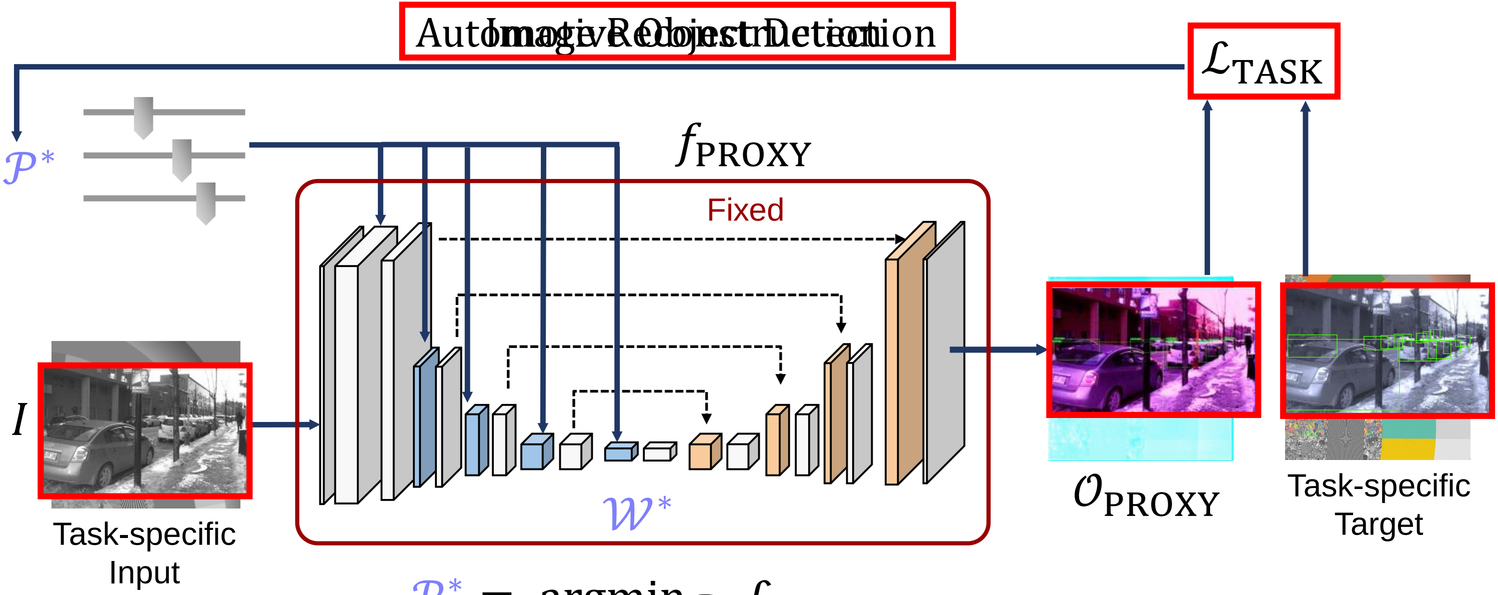
Stage 2: Optimizing Hyperparameters for Task-Specific Outputs

Stage 2: Optimizing Hyperparameters



$$\mathcal{P}^* = \operatorname{argmin}_{\mathcal{P}} \mathcal{L}_{\text{TASK}}$$

Stage 2: Using Task-Specific Datasets

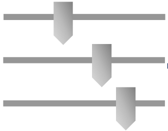


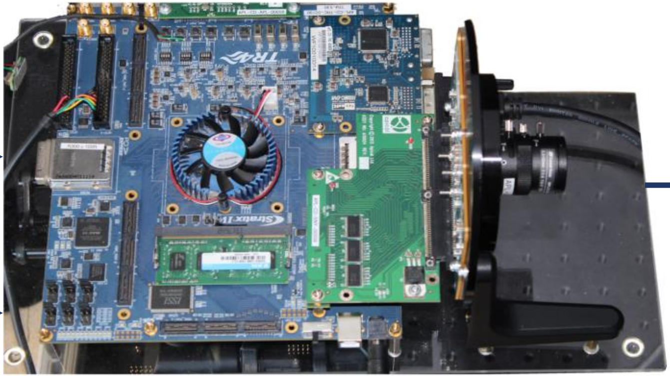
$$\mathcal{P}^* = \operatorname{argmin}_{\mathcal{P}} \mathcal{L}_{\text{TASK}}$$

Deploying Optimized Parameters

f_{ISP}

$\mathcal{P}^*$

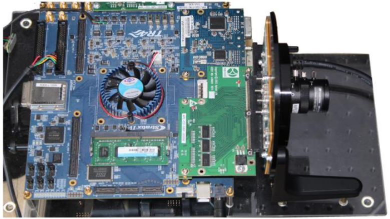




$\mathcal{O}_{\text{ISP}}$

Task-specific Hyperparameter Optimization

Natural Image Optimization



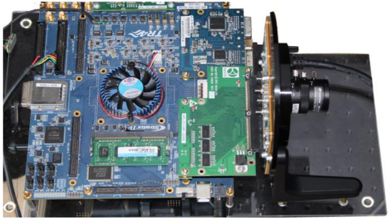
Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

Natural Image Capture with Hardware ISP

Natural Image Optimization



Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

ARM Mali-C71 Hardware ISP

Demoisaicking

Parameter	Value	Max
vh slope	190	2^8
vh thresh	220	2^{12}
va slope	175	2^8
va thresh	210	2^{12}
aa slope	170	2^8
aa thresh	100	2^{12}
uu slope	165	2^8
uu thresh	210	2^{12}
sharp alt ld	45	2^8
sharp alt ldu	45	2^8
sharp alt lu	25	2^8
fc alias slope	85	2^8
fc alias thresh	0	2^8
fc slope	130	2^8
np offset	3	2^8

White Balance

Parameter	Value	Max
gain 00	583	2^{12}
gain 01	271	2^{12}
gain 11	587	2^{12}

Denoising

Parameter	Value	Max
thresh 1h	5	2^8
strength 1	190	2^8
thresh 4h	10	2^8
strength 4	255	2^8
thresh long	48	2^8

Color & Tone Correction

Parameter	Value	Max
lut knee	207	2^8
lut power	129	2^8
lut shadow	31	2^8

Colorspace Conversion

Parameter	Value	Max
coef a 11	4461	5880
coef a 12	4001	5880
coef a 21	4068	5880
coef a 22	4398	5880
coef a 31	4135	5880
coef a 32	3842	5880

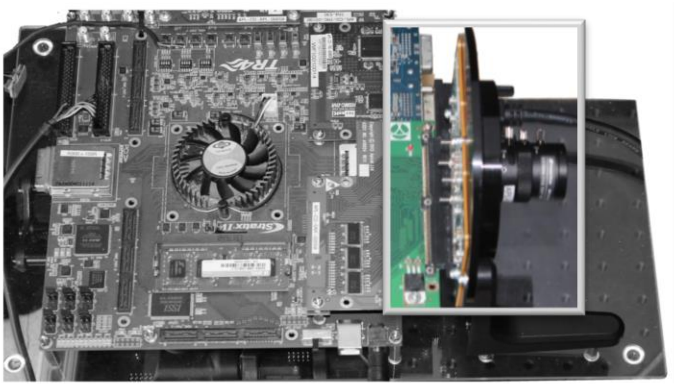
No problem for differentiable proxies!

- Parameter space is **high-dimensional** and of **mixed** types (continuous and discrete)
- ISP is **non-differentiable** and processing units are **black box**
- Tuning towards **arbitrary task metrics** is difficult for traditional hyperparameter optimization methods

ARM Mali-C71 Parameter Optimization: Stage 1

- Train proxy for ARM Mali-C71 ISP
- Acquire proxy training data using hardware-in-the-loop

ARM Mali-C71 ISP

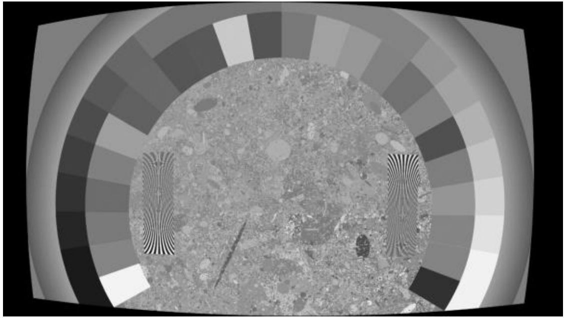


Hardware-in-the-loop

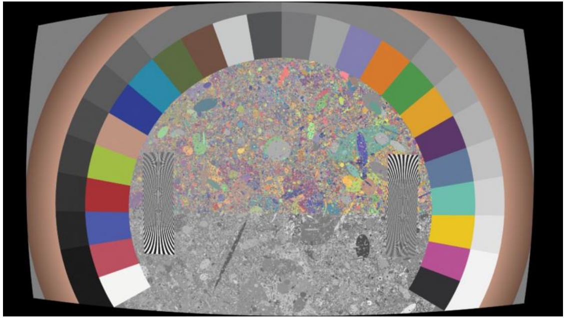
ARM Mali-C71 Parameter Optimization: Stage 1

- Train proxy for ARM Mali-C71 ISP
- Acquire proxy training data using hardware-in-the-loop

Captured RAW Image

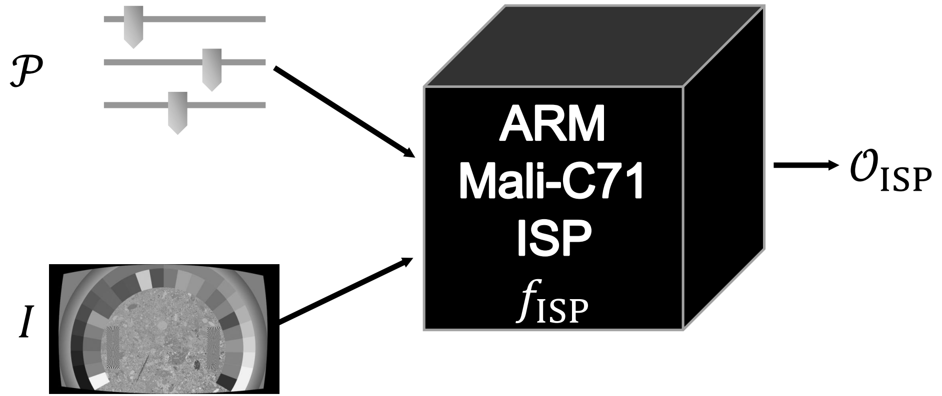


Calibration Image on Screen



ARM Mali-C71 Parameter Optimization: Stage 1

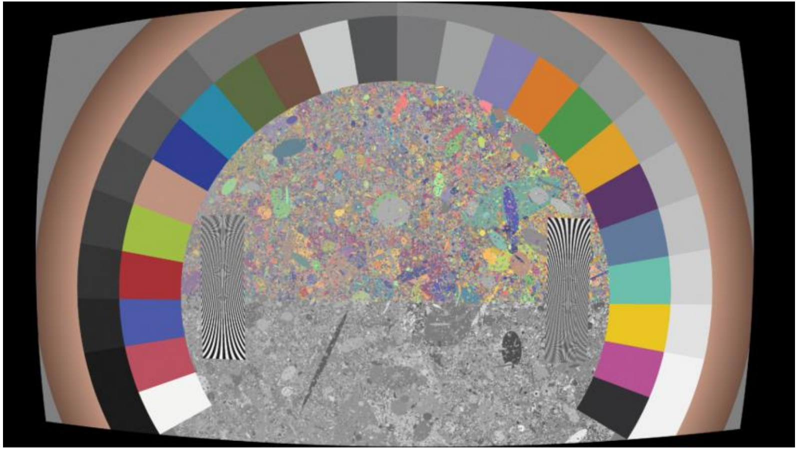
- Train proxy for ARM Mali-C71 ISP
- Acquire proxy training data using hardware-in-the-loop



ARM Mali-C71 Parameter Optimization: Stage 2

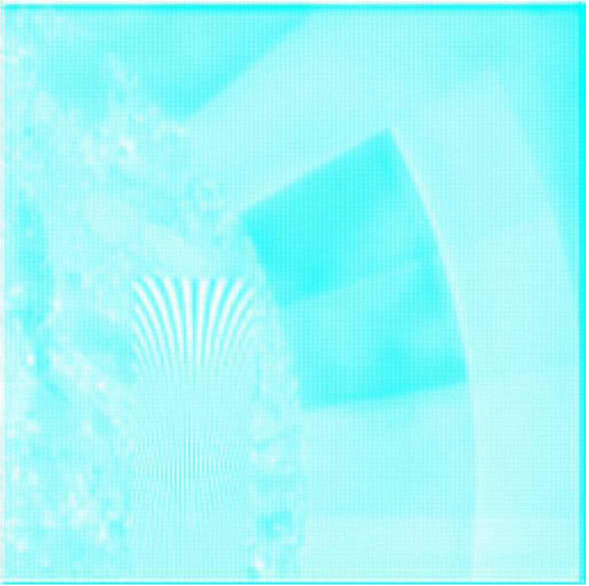
- Set rainbow calibration chart as target image
- Task loss is $\mathcal{L}_{\mathrm{TASK}} = \mathcal{L}_{\mathrm{PERCEPTUAL}} + 2\mathcal{L}_1$
- Parameters found enable ARM Mali-C71 to capture natural images

Target

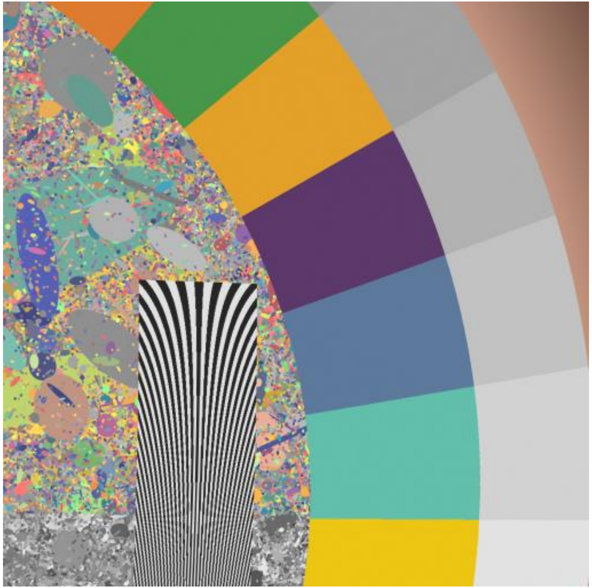


ARM Mali-C71 Parameter Optimization: Stage 2

Hyperparameter Optimization



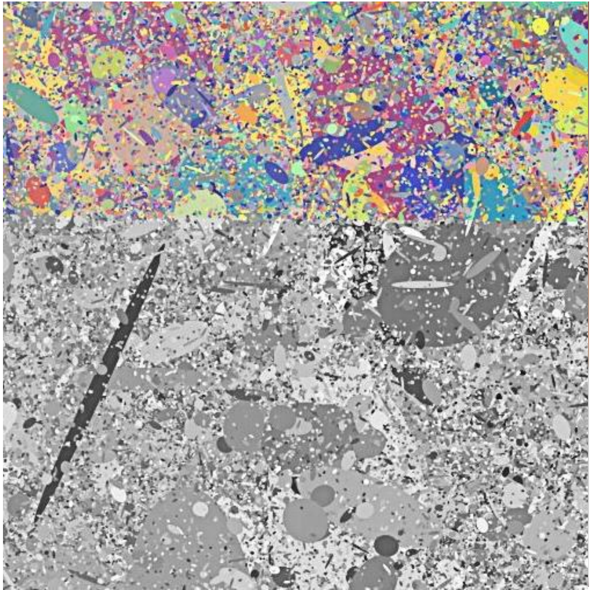
Target



ARM Mali-C71 Parameter Optimization: Stage 2

Hyperparameter Optimization

Target



Natural Image Capture Comparison

Manual Expert Tuning

Automatic Tuning Using Proxy

Natural Image Capture Comparison

Manual Expert Tuning

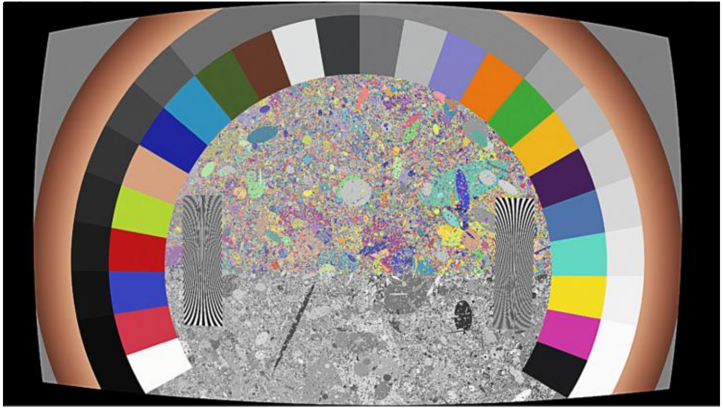
Automatic Tuning Using Proxy

	ARM MALI ISP (1× gain)		ARM MALI ISP (16× gain)	
	Manual Tuning	Proxy Optimized	Manual Tuning	Proxy Optimized
Perceptual Loss	0.244	**0.217**	0.523	**0.403**
Detail Accuracy	14.94	**13.03**	19.56	**18.79**
Color Accuracy	10.49	**10.23**	12.50	**12.36**
Zipper	0.111	**0.096**	0.227	**0.200**
Color Moire	2676	**727**	2836	**888**

Proxy Tuning – 3 hours

ARM Mali-C71 Parameter Optimization: Modified Target

Edge Enhanced, High Color Contrast
Target



Edge Enhanced, High Color Contrast Comparison

Original, Natural Image

Edge Enhanced, High Color Contrast Image

Edge Enhanced, High Color Contrast Comparison

Original, Natural Image

Edge Enhanced, High Color Contrast Image

Low Light Denoising with Software ISP

Natural Image Optimization

Low-light Denoising

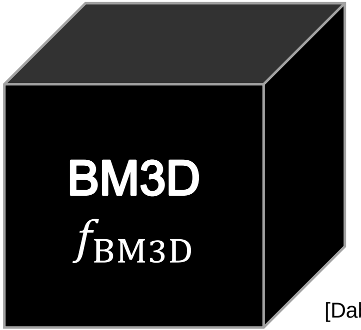
Proxy for Global Image Operations

Object Detection for Autonomous Cars

Low Light Denoising Details

BM3D Denoising Algorithm

- Parameter space is **mixed** between continuous and discrete
- ISP is **non-differentiable** and **black box**



[Dabov07]

Smartphone Image Denoising Dataset

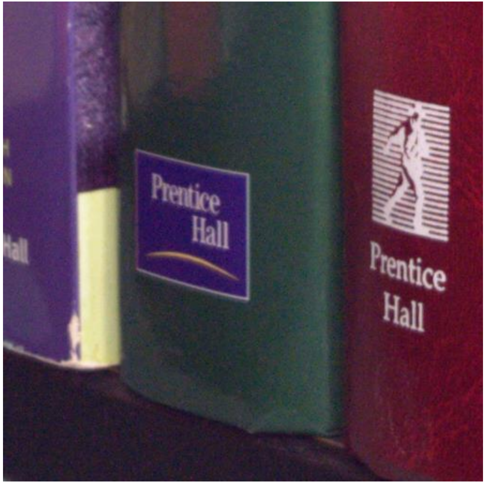
Noisy Images

Noiseless Images

[Abdelhamed18]

- Stage 1: Noisy images are passed through BM3D to form proxy training data
- Stage 2: Noiseless images are set as targets

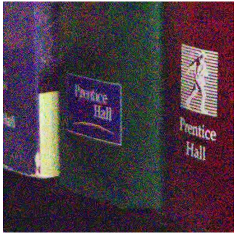
Denoising Qualitative Result



Long Exposure Image

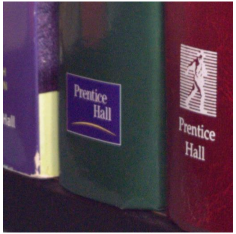
Noisy Image

BM3D w/ Default Hyperparameter Output



BM3D w/ Optimized Hyperparameter Output

Long Exposure Image



Denoising Quantitative Result

Method		PSNR	SSIM
Proxy-opt. BM3D	(Ours)	**34.34**	**0.911**
CBDNet	[Guo et al. 2018]	33.28	0.868
KSVD-DCT	[Elad and Aharon 2006]	27.51	0.780
KSVD-G	[Elad and Aharon 2006]	27.19	0.771
EPLL	[Zoran and Weiss 2011]	27.11	0.870
KSVD	[Aharon et al. 2006]	26.88	0.842
NLM	[Buades et al. 2005]	26.75	0.699
WNNM	[Gu et al. 2014]	25.78	0.809
BM3D	[Dabov et al. 2007]	25.65	0.685
FoE	[Roth and Black 2005]	25.58	0.792
TNRD	[Chen and Pock 2017]	24.73	0.643
MLP	[Burger et al. 2012]	24.71	0.641
GLIDE	[Talebi and Milanfar 2014]	24.71	0.774
LPG-PCA	[Zhang et al. 2010]	24.49	0.681
DnCNN	[Zhang et al. 2017]	23.66	0.583
DemosaicNet	[Gharbi et al. 2016]	22.38	0.369

Notable Results

Method	PSNR	SSIM
Proxy-opt. BM3D	**34.34**	**0.911**
CBDNet	33.28	0.868
BM3D	25.65	0.685

Denoising Quantitative Result

Method		PSNR	SSIM
Proxy-opt. BM3D	(Ours)	**34.34**	**0.911**
CBDNet	[Guo et al. 2018]	33.28	0.868
KSVD-DCT	[Elad and Aharon 2006]	27.51	0.780
KSVD-G	[Elad and Aharon 2006]	27.19	0.771
EPLL	[Zoran and Weiss 2011]	27.11	0.870
KSVD	[Aharon et al. 2006]	26.88	0.842
NLM	[Buades et al. 2005]	26.75	0.699
WNNM	[Gu et al. 2014]	25.78	0.809
BM3D	[Dabov et al. 2007]	25.65	0.685
FoE	[Roth and Black 2005]	25.58	0.792
TNRD	[Chen and Pock 2017]	24.73	0.643
MLP	[Burger et al. 2012]	24.71	0.641
GLIDE	[Talebi and Milanfar 2014]	24.71	0.774
LPG-PCA	[Zhang et al. 2010]	24.49	0.681
DnCNN	[Zhang et al. 2017]	23.66	0.583
DemosaicNet	[Gharbi et al. 2016]	22.38	0.369

Notable Results

Method	PSNR	SSIM
Proxy-opt. BM3D	**34.34**	**0.911**
CBDNet	33.28	0.868
BM3D	25.65	0.685

Denoising Quantitative Result

Method		PSNR	SSIM
Proxy-opt. BM3D	(Ours)	**34.34**	**0.911**
CBDNet	[Guo et al. 2018]	33.28	0.868
KSVD-DCT	[Elad and Aharon 2006]	27.51	0.780
KSVD-G	[Elad and Aharon 2006]	27.19	0.771
EPLL	[Zoran and Weiss 2011]	27.11	0.870
KSVD	[Aharon et al. 2006]	26.88	0.842
NLM	[Buades et al. 2005]	26.75	0.699
WNNM	[Gu et al. 2014]	25.78	0.809
BM3D	[Dabov et al. 2007]	25.65	0.685
FoE	[Roth and Black 2005]	25.58	0.792
TNRD	[Chen and Pock 2017]	24.73	0.643
MLP	[Burger et al. 2012]	24.71	0.641
GLIDE	[Talebi and Milanfar 2014]	24.71	0.774
LPG-PCA	[Zhang et al. 2010]	24.49	0.681
DnCNN	[Zhang et al. 2017]	23.66	0.583
DemosaicNet	[Gharbi et al. 2016]	22.38	0.369

Notable Results

Method	PSNR	SSIM
Proxy-opt. BM3D	**34.34**	**0.911**
CBDNet	33.28	0.868
BM3D	25.65	0.685

Tone Mapping with Software ISP

Natural Image Optimization

Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

See paper for details!

Object Detection with Hardware ISP for Autonomous Cars

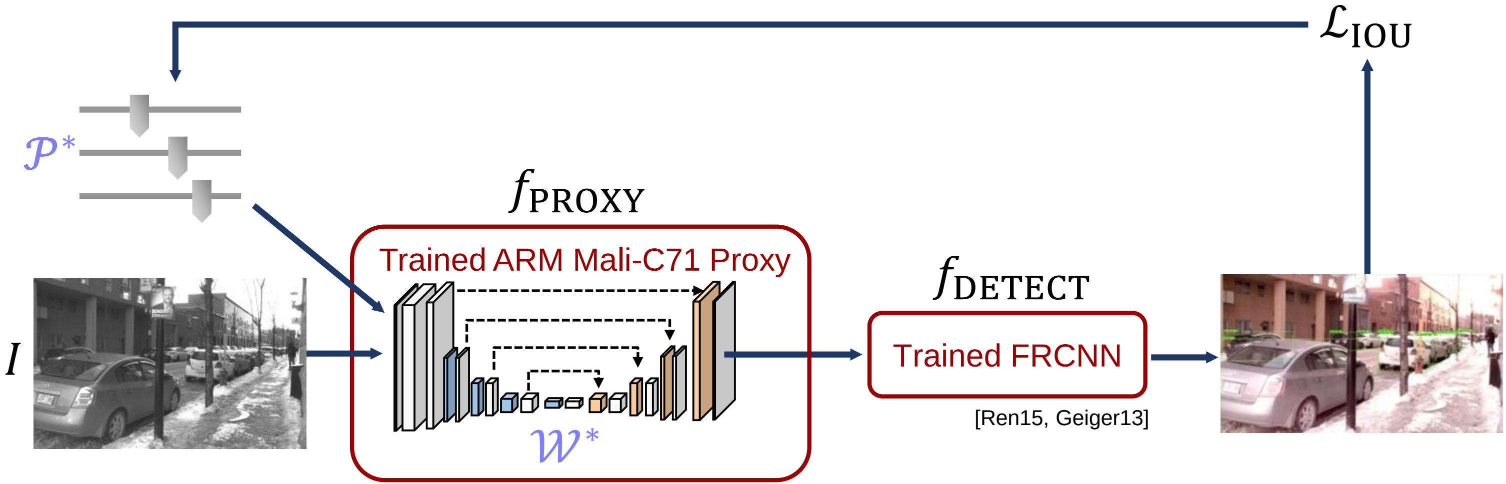
Natural Image Optimization

Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

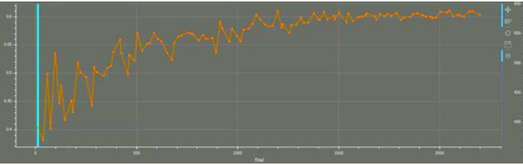
Object Detection with Hardware ISP for Autonomous Cars



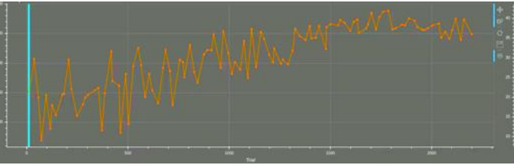
$$\mathcal{P}^* = \operatorname{argmin}_{\mathcal{P}} \mathcal{L}_{\mathrm{IOU}}(f_{\mathrm{DETECT}}(f_{\mathrm{PROXY}}(I, \mathcal{P}, \mathcal{W}^*)))$$

Object Detection Optimization Time-lapse

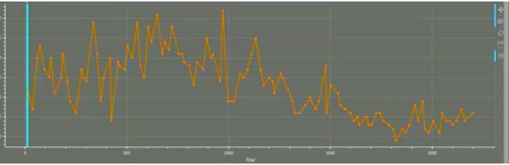
CV Precision KPI



True positives



False negatives

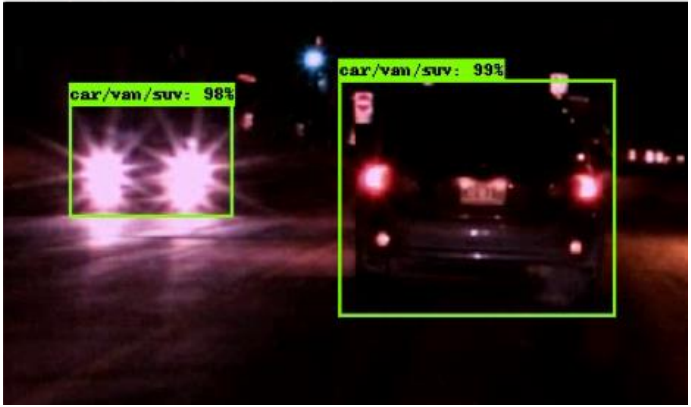


Proxy-optimization produces images that are

- different from images intended for human viewing
- favored by FRCNN due to strong denoising through slight blurring

Object Detection Result: Increased True Positives

Proxy-Optimized



Object Detection Result: Decreased False Positives

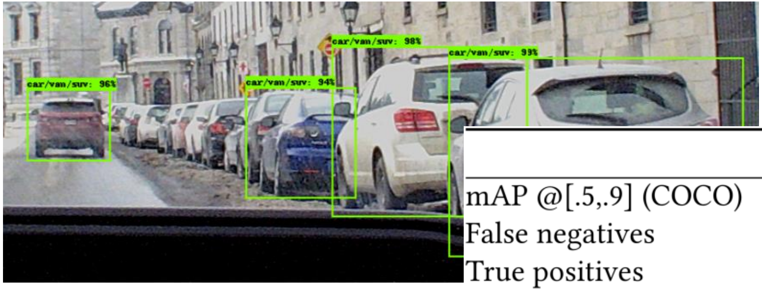
Proxy-Optimized

Object Detection Result: Tighter Bounding Boxes

Proxy-Optimized

Object Detection Result: Quantitative Results

Expert-Tuned



Proxy-Optimiz

	Expert-tuned	Optimized
mAP @[.5,.9] (COCO)	0.31	**0.37**
False negatives	778	**587**
True positives	4860	**5764**
person	662	**713**
bus/truck/tram	97	**134**
car/van/suv	4101	**4917**

Object Detection Comparison

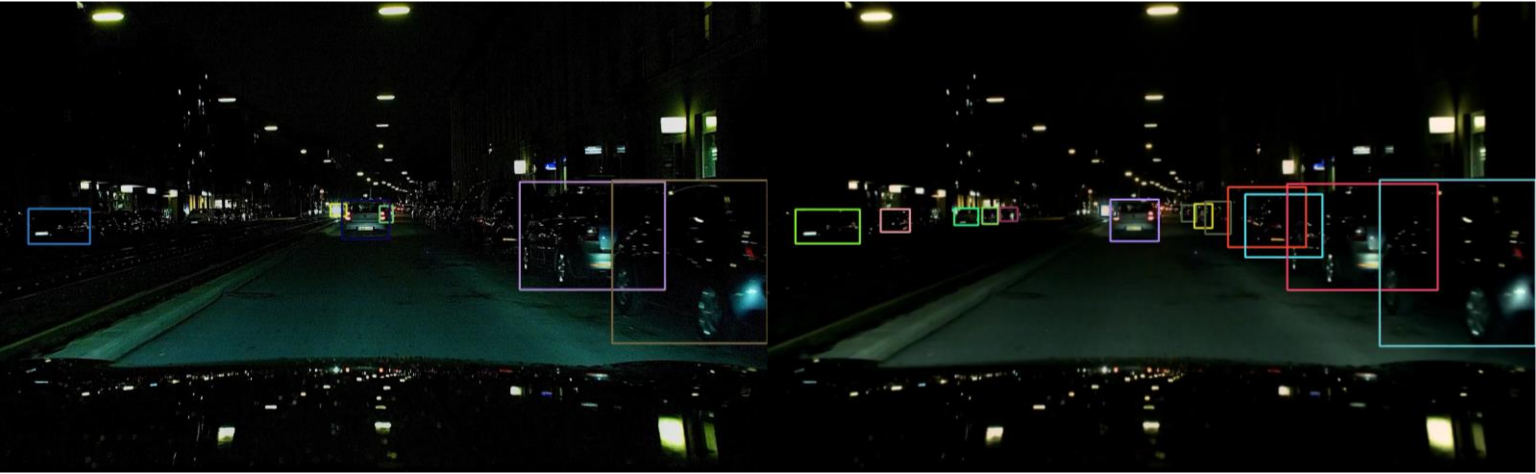
Expert-Tuned

Proxy-Optimized

Object Tracking Comparison

Expert-Tuned

Proxy-Optimized



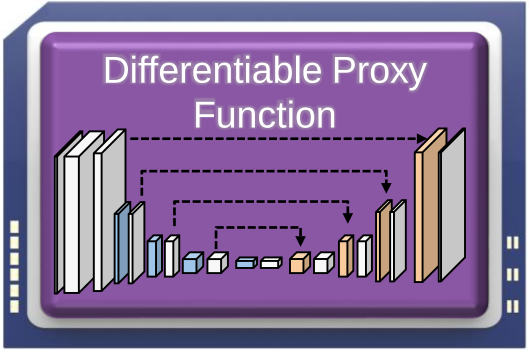
Conclusion

Automated hyperparameter tuning of ISPs

- **High-dimensionality**
- **Continuous / discrete parameter space**
- **Non-differentiability and black box**

Tuned towards **arbitrary task metrics**

- Outperformed state of the art



Natural Image Optimization

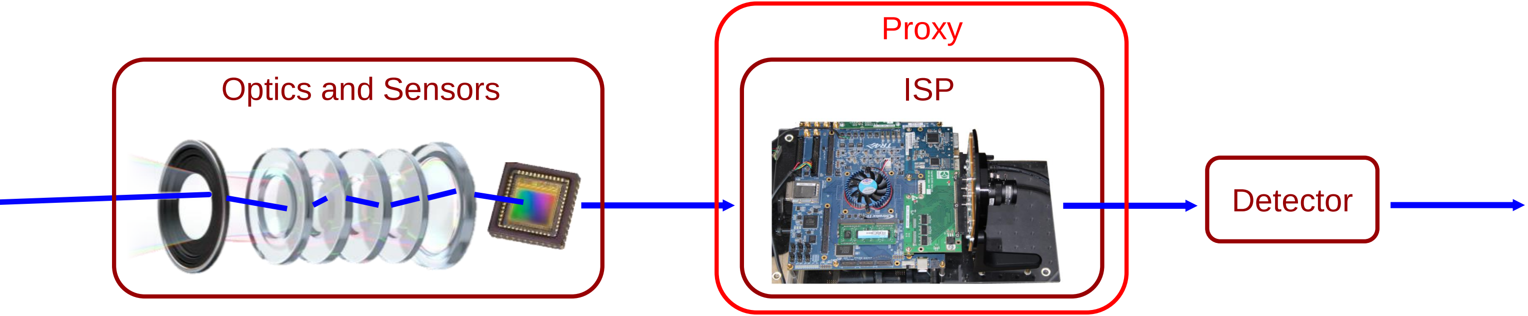
Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

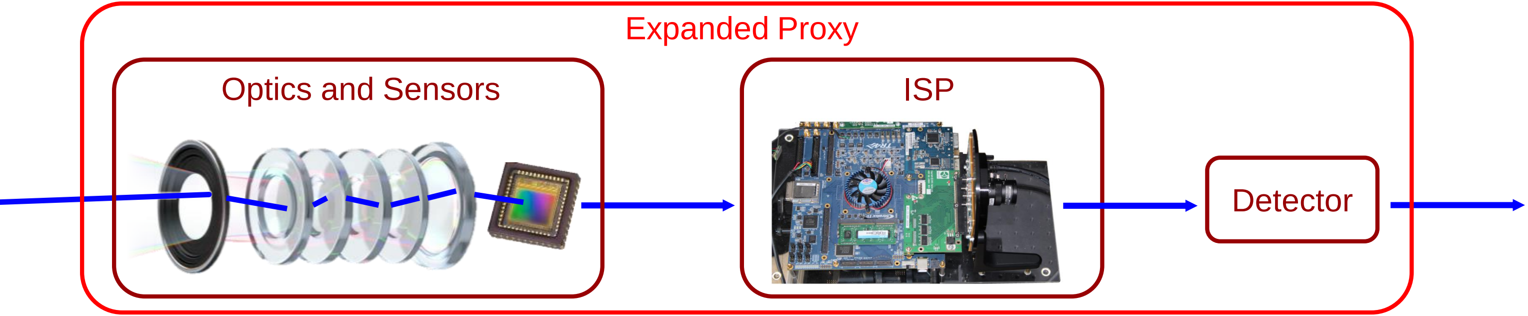
Future work

- Full end-to-end optimization of imaging systems



Future work

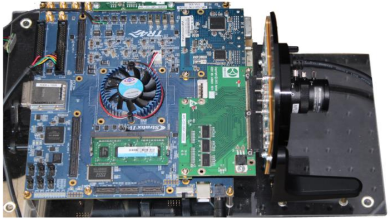
- Full end-to-end optimization of imaging systems



- Dynamic control hyperparameters

Hyperparameter Optimization in Black-box Image Processing using Differentiable Proxies

Natural Image Optimization



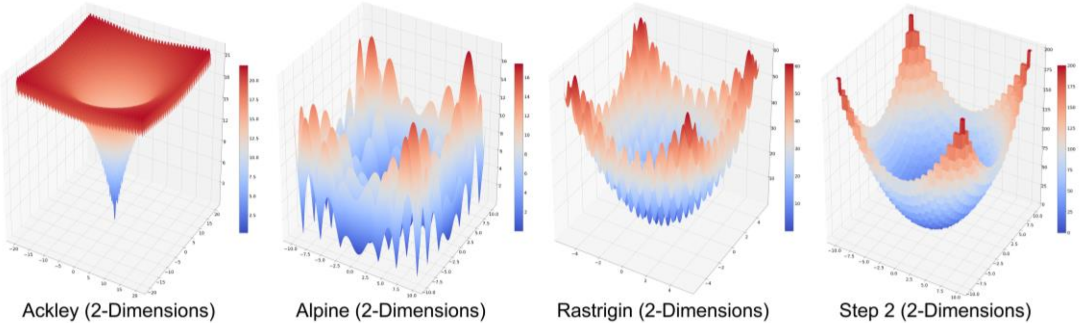
Low-light Denoising

Proxy for Global Image Operations

Object Detection for Autonomous Cars

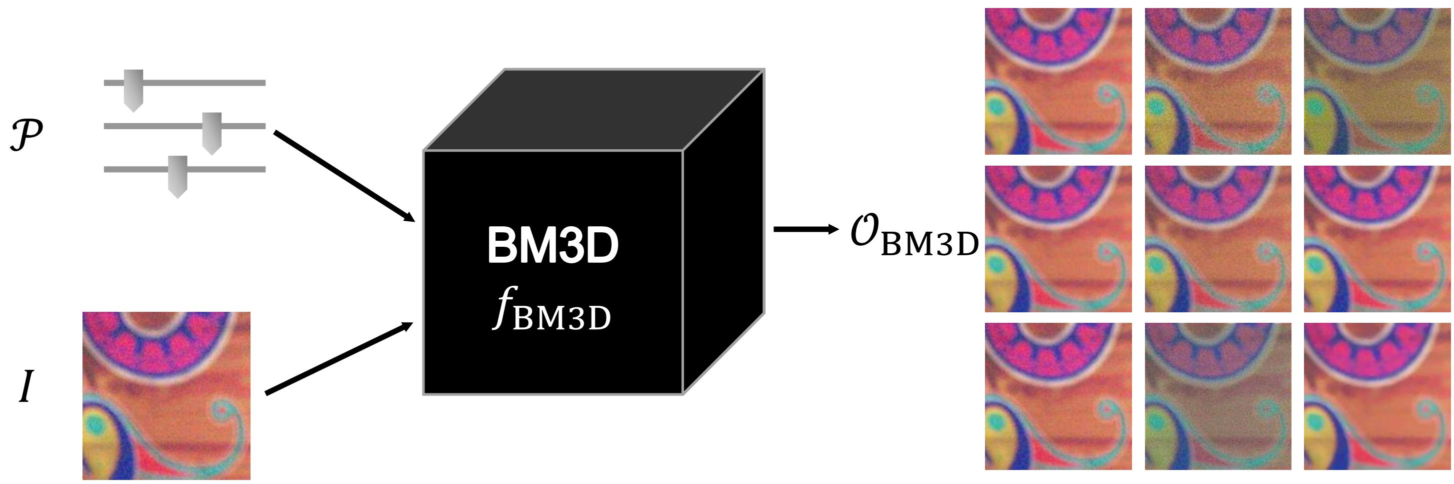
Benchmark functions

Tested using 20 dimensional versions (2D shown here)



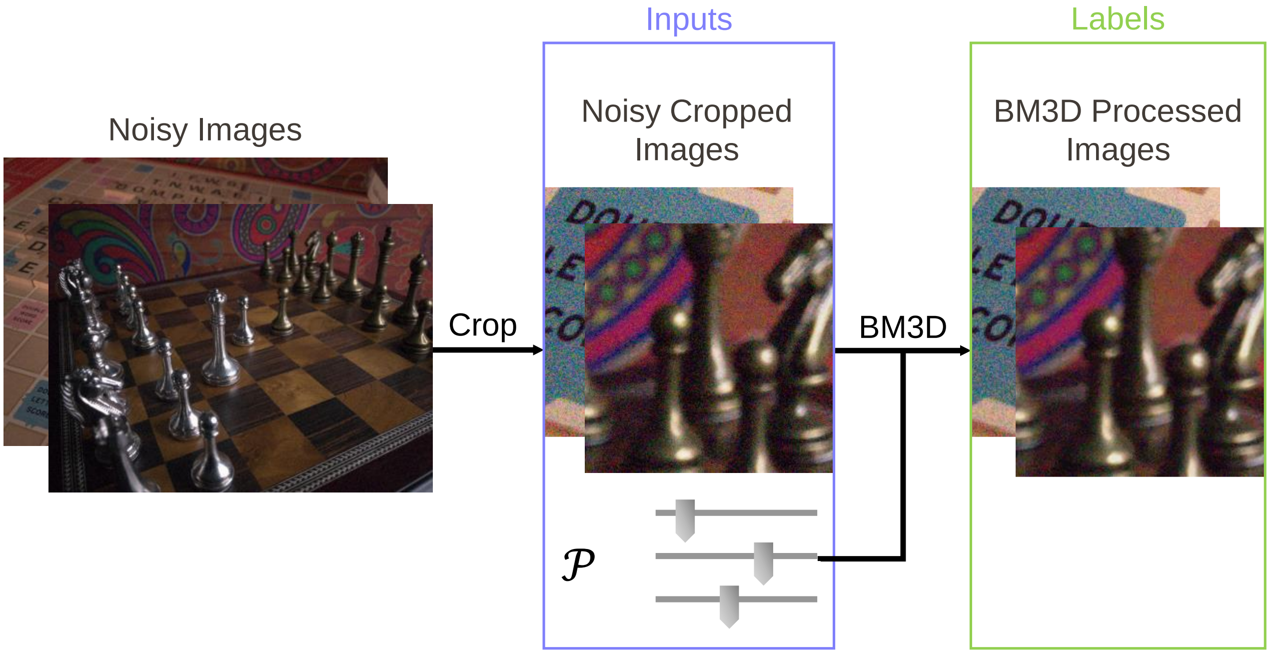
Test Functions (20D variants)	Differentiable Proxies	BayesOpt [2014]	HyperOpt [2013]	Powell [1965]	Nelder-Mead [1965]
Ackley	**16.84**	21.65	20.27	20.95	21.18
Rastrigin	**94.35**	124.91	137.79	209.42	258.51
Step-2	**6359.0**	14018.0	22444.0	49120.0	51669.0
Alpine	**24.48**	24.96	29.02	50.85	24.48

Optimizing a Classical Denoiser



$$\mathcal{O}_{\mathrm{BM3D}} = f_{\mathrm{BM3D}}(I, \mathcal{P})$$

Optimizing a Classical Denoiser



Optimizing a Classical Denoiser

Noisy Cropped Images

Ground Truth Images

Crop

Labels

Cropped Ground Truth Images

$$\mathcal{L}_{\text{TASK}} = \mathcal{L}_{\text{MSE}}(\, f_{\text{PROXY}}(I_{\text{NOISY}}, \mathcal{P}\ ;\ \mathcal{W}^{*}), I_{\text{GT}}))$$

Optimizing a Classical Denoiser

Noisy Image

BM3D w/ Default Hyperparameter Output

BM3D w/ Optimized Hyperparameter Output

Ground Truth Image