Self-supervised Image Restoration

Gabriele Facciolo

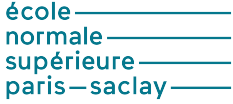
Centre Borelli, ENS Paris-Saclay

This is joint work with

Ngoc Long Nguyen, Valery Dewil, Jérémy Anger,

Axel Davy, Thibaud Ehret, Pablo Arias, Jean-Michel Morel

ILLS Workshop - 26/5/2023



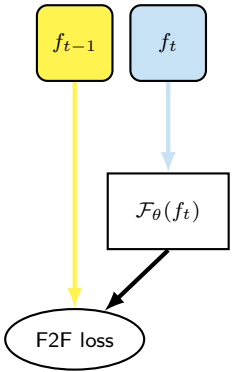
Collaborators and Image Processing Group @Centre Borelli

The image processing group at Centre Borelli

- About 30 researchers ($\sim$ 14 PhD students) in image processing and computer vision
- Main research areas:
  - Image/video processing and analysis (restoration, synthesis, detection)
  - Remote Sensing data exploitation (optical, radar, multi-spectral)
  - Detection theory and applications (low level vision, anomaly detection, forgery detection)

The works in this presentation are motivated by **image/video restoration** and **remote sensing applications**

Overview



Suppose we're given a video with an unknown noise distribution.

We're going to see how to denoise it:

- Starting from a pre-trained denoising network (e.g. AWGN $\sigma = 20$)
- Fine-tuning it using a **single noisy video by penalizing the loss between the predicted frame and the previous one**

In this talk:

- Review of noise-to-noise (N2N)
- extension to self-supervised **video denoising** (frame-to-frame, mosaic-to-mosaic, multi-frame-to-frame)
- extension to self-supervised multi-image **super-resolution**

Initial Motivation: Denoising real noise with CNNs is not easy

	Applied/ Evaluated	BM3D [10]	NLM [4]	KSVD [1]	KSVD-DCT [11]	KSVD-G [11]	LPG-PCA [32]	FoE [27]	MLP [6]	WNNM [16]	GLIDE [29]	TNRD [8]	EPLL [35]	DnCNN [33]
PSNR	Raw/Raw	**45.52**	44.06	43.26	42.70	42.50	42.79	43.13	43.17	44.85	41.87	42.77	40.73	43.30
	Raw/sRGB	**30.95**	29.39	27.41	28.21	28.13	30.01	27.18	27.52	29.54	25.98	26.99	25.19	28.24
	sRGB/sRGB	25.65	26.75	26.88	**27.51**	27.19	24.49	25.58	24.71	25.78	24.71	24.73	27.11	23.66
SSIM	Raw/Raw	**0.980**	0.971	0.969	0.970	0.969	0.974	0.969	0.965	0.975	0.949	0.945	0.935	0.965
	Raw/sRGB	0.863	0.846	0.832	0.784	0.781	0.854	0.812	0.788	**0.888**	0.816	0.744	0.842	0.829
	sRGB/sRGB	0.685	0.699	0.842	0.780	0.771	0.681	0.792	0.641	0.809	0.774	0.643	**0.870**	0.583
Time	Raw	34.3	210.7	2243.9	133.3	153.6	438.1	6097.2	131.2	1975.8	12440.5	**15.2**	653.1	51.7
	sRGB	**27.4**	621.9	9881.0	96.3	92.2	2004.3	12166.8	564.8	8882.2	36091.6	45.1	1996.4	158.9

[Benchmarking Denoising Algorithms with Real Photographs, Plötz'17]

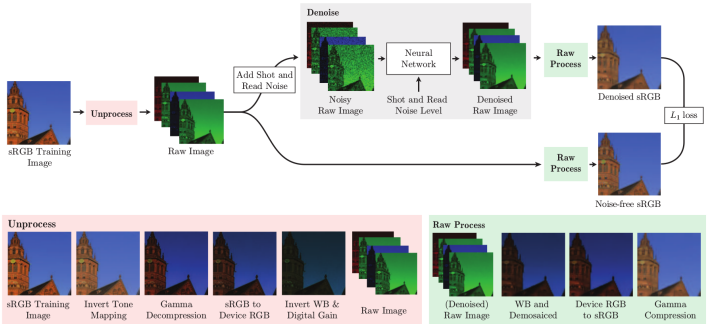
The problem of domain gap

- Simulated noise $\neq$ real noise
- CNNs are less robust than traditional methods to inaccurate noise models!
- How do we train with realistic noise?

How to train for real noise?

I) Acquire data with ground truth

II) Simulate realistic data



[Benchmarking Denoising Algorithms with Real Photographs, Plötz'17]

[A High-Quality Denoising Dataset for Smartphone Cameras, Abdelhamed'18]

[Real-world Noisy Image Denoising: A New Benchmark, Xu'18]

[Unprocessing images for learned raw denoising, Brooks'19]

How to train for real noise?

III) Self-supervised training

Train using exclusively noisy images. By using the noisy image as target?

$$R_{\mathsf{self}}(\mathcal{F}) = \sum_{i=1}^{m} \|\mathcal{F}(v_i) - v_i\|^2.$$

Trivial minimizer: identity function $\mathcal{F}(v) = v$.

- No need for acquiring GT data
- No simulation required (useful when noise model is unkown)

Contents

Supervised training: clean labels

When training we **minimize the empirical risk** to find an estimator $\mathcal{F}$:

$$\mathcal{R}^{\text{emp}}(\mathcal{F}) = \frac{1}{m}\sum_{i=1}^{m} \ell(\mathcal{F}(v_i), u_i) \xrightarrow[m\to\infty]{} \mathbb{E}_{v,u}\{\ell(\mathcal{F}(v), u)\}$$

where $p(v, u) = p(v|u)p(u)$ is the joint PDF for

- the data v (noisy image) and
- the label u (clean image).

Optimal estimators: $\implies$

if ℓ is squared L_2	$\Rightarrow \mathcal{F}^*(v) = \mathbb{E}\{u\|v\}$ (MMSE)
if ℓ is L_1	$\Rightarrow \mathcal{F}^*(v) = \text{median}\{u\|v\}$

Noise-to-noise training: noisy labels

$$\underbrace{\text{minimize} \sum_{i=1}^{m} \ell(\mathcal{F}(v_i), z_i)}_{\text{N2N training}} \approx \underbrace{\text{minimize} \sum_{i=1}^{m} \ell(\mathcal{F}(v_i), u_i)}_{\text{supervised training}}$$

- u_i clean ground truth images
- $v_i = u_i + n_i$ noisy images used as input
- $z_i = u_i + n_i'$ noisy images used as label

- Requires **independent noise realizations**
- Some other properties of the noise (zero mean, unbiased with respect to the median, etc.)

[Noise2Noise: Learning Image Restoration without Clean Data, Lehtinen'18]

Noise-to-noise: training with noisy labels

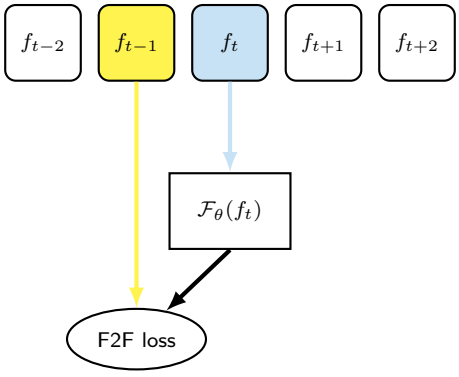
Networks trained with N2N attain almost the same performance as those trained with clean labels

	Gaussian (σ=25)			Poisson (λ=30)			Bernoulli (p=0.5)		
	clean	noisy	BM3D	clean	noisy	ANSC	clean	noisy	DIP
Kodak	32.50	32.48	31.82	31.52	31.50	29.15	33.01	33.17	30.78
BSD300	31.07	31.06	30.34	30.18	30.16	27.56	31.04	31.16	28.97
Set14	31.31	31.28	30.50	30.07	30.06	28.36	31.51	31.72	30.67
Average	**31.63**	**31.61**	**30.89**	**30.59**	**30.57**	**28.36**	**31.85**	**32.02**	**30.14**

[Noise2Noise: Learning Image Restoration without Clean Data, Lehtinen'18]

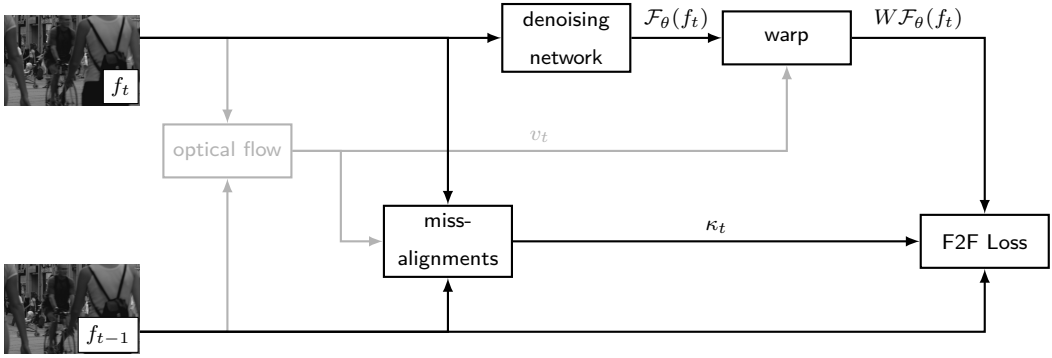
Contents

Exploit the temporal redundancy between video frames to apply N2N



[Model-blind video denoising via frame-to-frame training, Ehret'18]

Frame-to-frame (F2F) loss



- TV-L1 optical flow [Zach,Pock,Bischof'07]
- Occlusion detection based on alignment residual and optical flow colisions
- Warp with differentiable bicubic interpolation
- Optical flow and occlusion masks are computed from the noisy data

Frame-to-frame loss

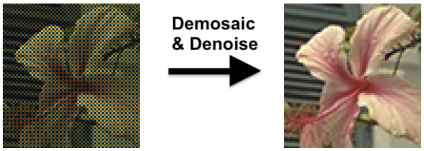
$$\ell_1^{\mathsf{F2F}}(W\mathcal{F}_\theta(f_t), f_{t-1}, \kappa_t) = \sum_x \kappa_t(x)\,|W\mathcal{F}_\theta(f_t)(x) - f_{t-1}(x)|$$

where

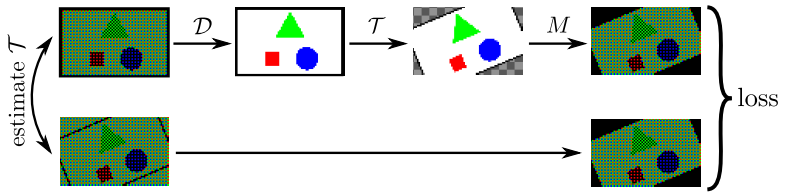
- $v_{t-1,t}$: is the optical flow from frame $t-1$ to t
- $Wu_t(x) = u_t(x + v_{t-1,t}(x))$: warps the frame according to the flow $v_{t-1,t}$
- κ_t: is a mask of mismatched pixels

The main **drawback of F2F** is that it trains a bf single-frame denoiser $\longrightarrow$ lacks **temporal consistency**

Extends to Joint Denoising and Demosaicking

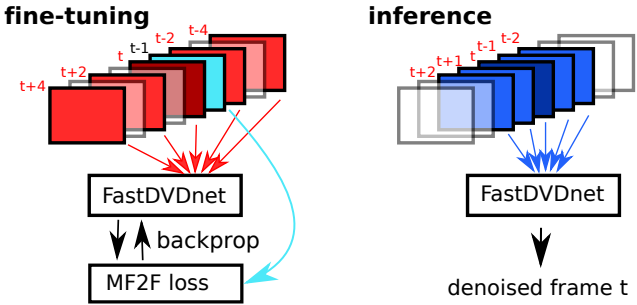


- Train a joint denoising and demosaicking network **without supervisory data**
- **Use bursts** of mosaicked images for training



[Joint demosaicing and denoising by overfitting of bursts of raw images. Ehret'19]

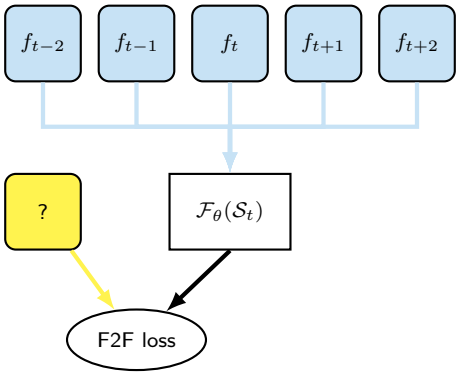
Extension to multi-frame video denoising: Multi-Frame-to-Frame (MF2F)



[Self-supervised training for blind multi-frame video denoising, Dewil'21]

We use it to train/fine-tune a state-of-the-art multi-frame video denoiser as FastDVDnet [Tassano'19] directly on REAL DATA (no need for GT)

Stack options for MF2F fine-tuning



Observation. *Let* (z, y) *distributed according to* $p(z, y)$.

Let $\hat{y}(z) = \mathcal{F}^*(z)$ *given by the minimizer of a loss with a label that is a function* t *of the observation* z*:*

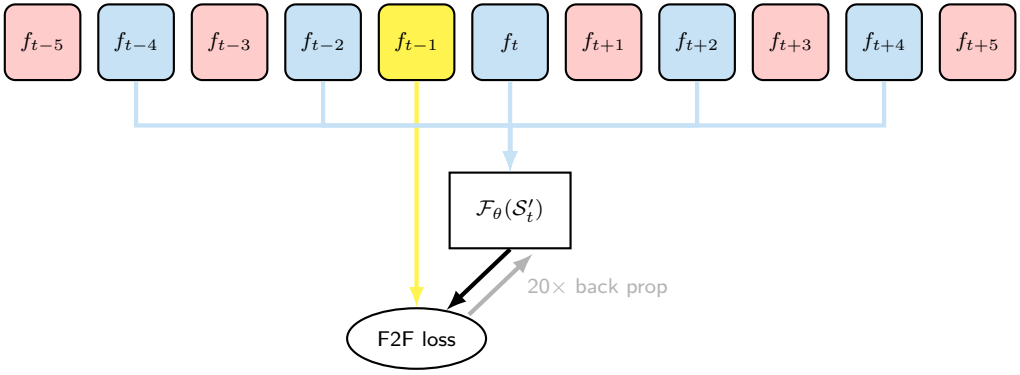
$$\mathcal{F}^* = \arg\min_{\mathcal{F}} \mathbb{E}_z\{\ell(\mathcal{F}(z), t(z))\}.$$

Then $\mathcal{F}^*(z)$ *doesn't depend on the data distribution* $p(z, y)$.

In other words: if the label is a function of the network's input you won't have data-driven learning.

Fine-tuning stack

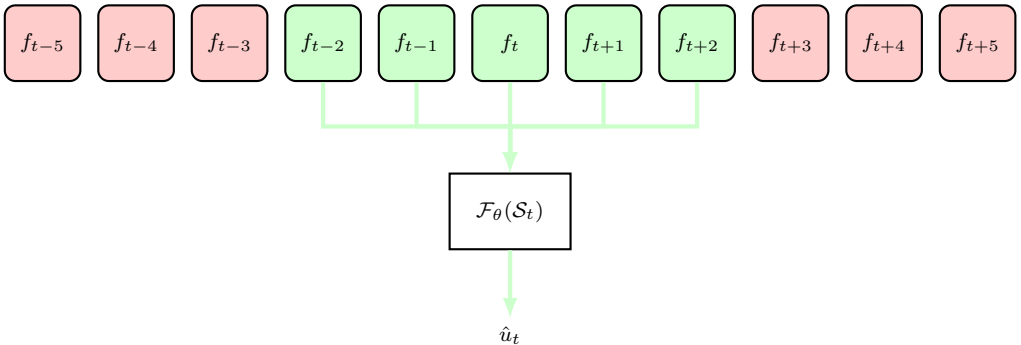
Fine-tune must be done by dilating or shifting the frames in the stack.



- Fine-tuning can be done **online** (while processing the video frames) ⇒ fast adaptation to noise changes
- or **offline** by training on a set of videos ⇒ better overall performance

Inference stack

Use the normal stack at inference time.



Results on synthetic data

		Non-blind	Model blind	
		FastDVDnet	MF2F	
		supervised	online	offline
Derf	Gaussian 20	36.97	37.32	**37.48**
	Gaussian 40	34.00	34.24	**34.27**
	Poisson 1	40.45	40.39	**40.51**
	Poisson 8	35.30	35.57	**35.68**
	Box 40 3	35.42	35.50	**35.60**
	Box 65 5	34.78	34.29	**34.35**
	Demosaicked 4	34.85	34.75	**34.81**
NTIRE	Gaussian 20	37.49	37.32	**37.55**
	Gaussian 40	34.27	34.17	**34.26**
	Poisson 1	40.63	40.01	**40.16**
	Poisson 8	35.72	34.99	**35.00**
	Box 40 3	37.28	36.65	**36.76**
	Box 65 5	36.81	35.65	**35.79**
	Demosaicked 4	34.50	33.95	**33.98**

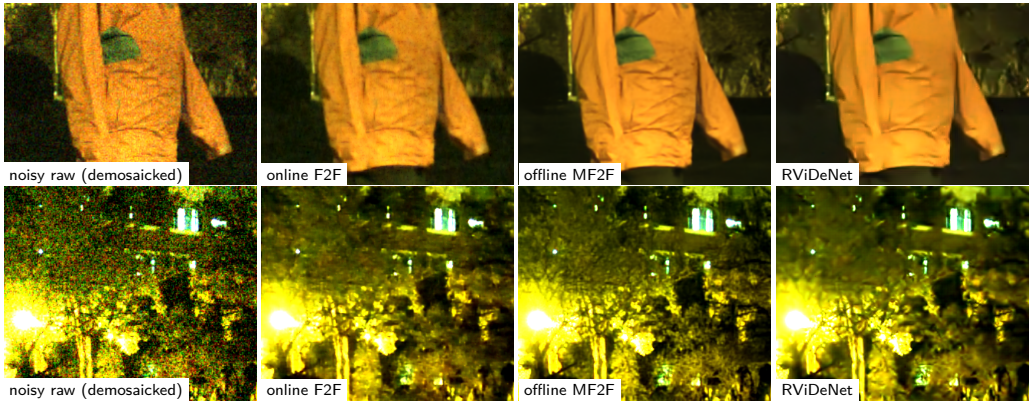
MF2F fine-tuning applied to FastDVDnet pre-trained for AWGN $\sigma = 25$

F2F fine-tuning applied to the weights of a single frame DnCNN for AWGN $\sigma = 25$.

Best PSNR

Best PSNR among blind methods.

Results on real noise



Details from frame of a denoised raw video (ISO 12800) processed by F2F, offline MF2F, and RViDeNet. All images are demosaicked and gamma corrected.

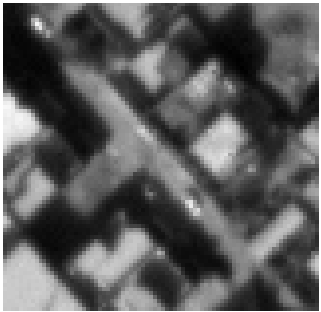
[Supervised raw video denoising with a benchmark dataset on dynamic scenes (RViDeNet), Yue'20]

Contents

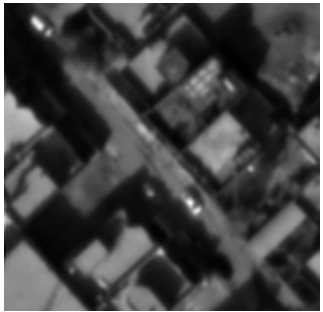
Multi-image super resolution of push-frame satellite images

SkySAT satellites operate in push-frame mode:

- 40 frames per second.
- Each point is observed in $\geq$ 15 consecutive frames.



DSA-Self



Real Skysat L1A Frames

Proposed method $\times 2$

[Self-supervised multi-image super-resolution for push-frame satellite images, Nguyen'21] ← Best student paper

[Self-Supervised Super-Resolution for Multi-Exposure Push-Frame Satellites, Nguyen'22]

Self-supervised Multi-image super-resolution

Input: LR seq. minus the ref.

DSA-Self

Output: SR image

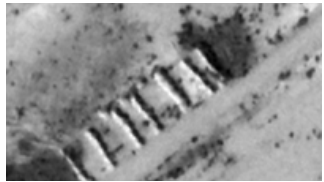
Train using the noisy LR reference frame as the target!

Self-supervised Multi-image super-resolution

Input: LR seq. minus the ref.

DSA-Self

Train using the noisy LR reference frame as the target!



Output: SR image

Downsample

Downsampled SR

Self-supervised Multi-image super-resolution

Input: LR seq. minus the ref.

DSA-Self

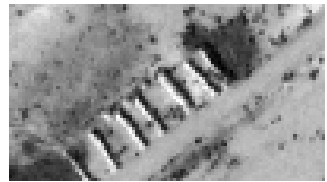
Output: SR image

Train using the noisy LR reference frame as the target!

Downsample

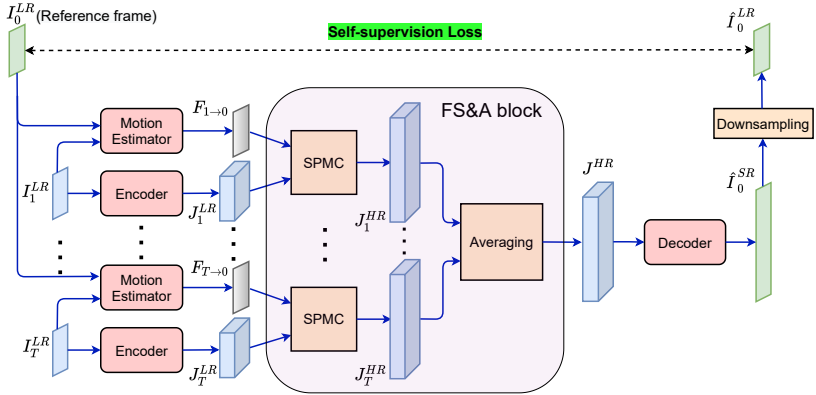
Self-SR Loss

$$\|\mathrm{DS}(I^{out}) - I^{target}\|$$



Downsampled SR

Deep Shift-and-Add (DSA) framework (during training)



Overview of our proposed self-supervised MISR framework at training time. The depicted loss represents the self-supervision term ℓ_{self}, for simplicity the losses concerning the motion estimation module are not illustrated. Note that at inference time the frame I_0^{LR} is also encoded and fed to the **FS&A** block.

Self-supervised training

- Self-supervision loss

$$\ell_{self}(\hat{I}_0^{SR}, I_0^{LR}) = \|\textbf{Downsample}\,(\hat{I}_0^{SR} * k) - I_0^{LR}\|_1$$

This also performs image **sharpening** by convolving with the k, the blur kernel of the system.

- Motion estimation loss

$$\ell_{me}(\{F_{t\to 0}\}_{t=1}^{T}) = \sum_t \|I_t^{LR} - W(I_0^{LR}, F_{t\to 0})\|_1 + \lambda_1 TV(F_{t\to 0}),$$

where W denotes the warp according to the flow $F_{t\to 0}$

The full loss is: loss $= \ell_{self} + \lambda_2 \ell_{me}$, where we set $\lambda_1 = 0.01, \lambda_2 = 10$.

SR on synthetic data: Quantitative analysis

Average PSNR (dB) over the validation dataset for different methods with different number of input images (T) per sequence.

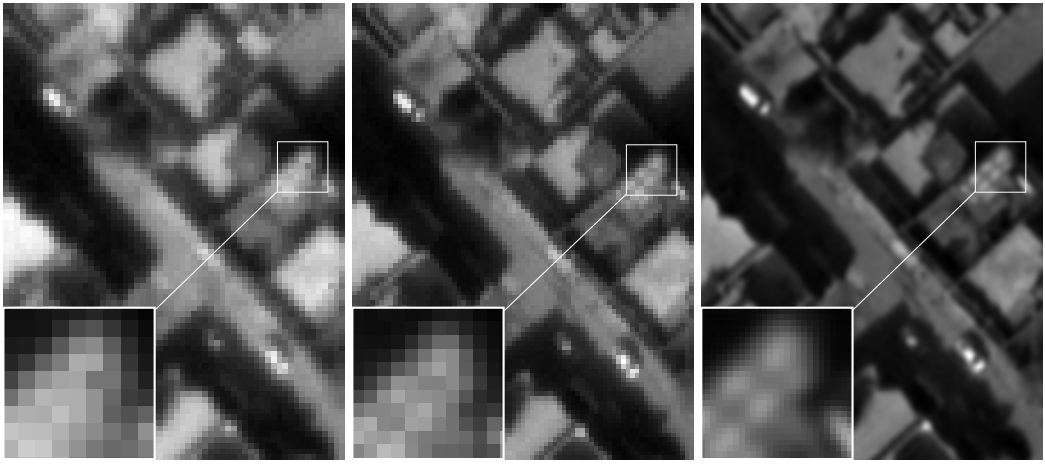
Method	Shift-and-Add	HighRes-net[1]	ACT-Spline[2]	**DSA-Self**	**DSA**
T = 5	42.99	45.63	45.54	**45.75**	**45.82**
T = 16	47.72	48.17	48.38	**49.27**	**49.33**
T = 30	49.95	49.05	50.15	**50.45**	**50.50**

- Our methods rank first (gain ≈1dB).
- The gap between self-supervised and supervised methods is small (<0.1dB).

1 [Deudon et al. "Highres-net: Recursive fusion for multi-frame super-resolution of satellite imagery." arXiv 2020.]

2 [Anger et al. "Fast and accurate multi-frame super-resolution of satellite images." ISPRS (2020).]

SR $\times 3$ of real SkySAT data



(a) L1A frame (b) Planet L1B ($\times 1.25$) (c) DSA-Self ($\times 3$)

Super-resolution from a sequence of 15 SkySat L1A frames. The result of our method is more resolved and contains less noise than the Planet L1B products.

Extension to Multi-Exposure Push-Frame Satellites

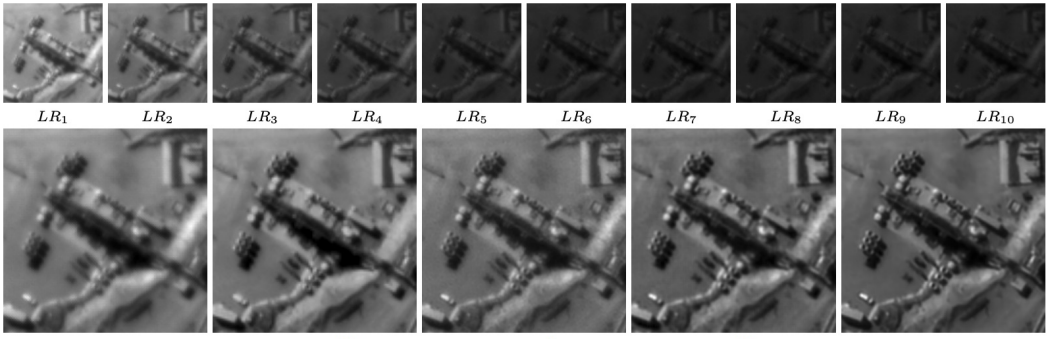


Figure 1. Super-resolution from a real multi-exposure sequence of 10 SkySat images. Top row: Original low resolution images with different exposures. Bottom row: Reconstructions from five methods, including ours trained with self-supervision (right).

[Self-Supervised Super-Resolution for Multi-Exposure Push-Frame Satellites, Nguyen'22]

L1BSR: Super Resolution of Sentinel-2 L1B images

Figure: The transition from Level-1B (L1B) data to Level-1C (L1C) data

- Single image super-resolution (x2) can be done on Sentinel-2 by exploiting inter-band shift and alias
- This can be learned in a self-supervised manner by exploiting the L1B product

[On The Role of Alias and Band-Shift for Sentinel-2 Super-Resolution, Nguyen '23]

[L1BSR: Exploiting Detector Overlap for Self-Supervised SISR of Sentinel-2 L1B Imagery, Nguyen '23] ← Best student paper @EarthVision'23

L1BSR Context: Overlapping areas in L1B images

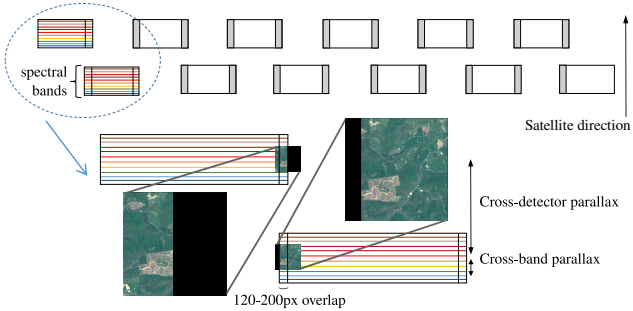
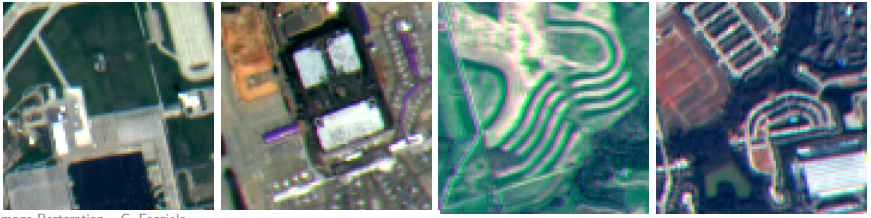


Figure: Overlap between CMOS detectors FOV → overlapping L1B crops



L1BSR Context: Overlapping areas in L1B images

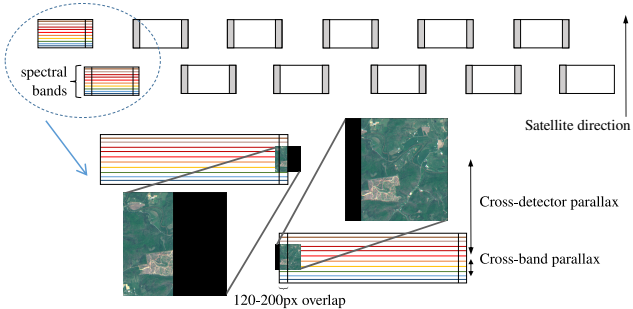
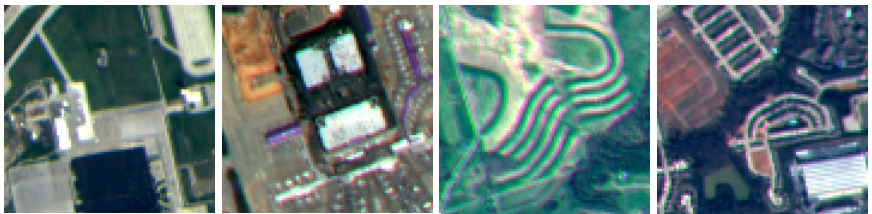


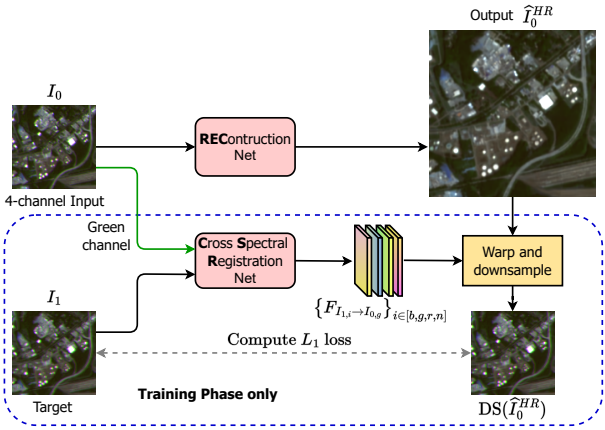
Figure: Overlap between CMOS detectors FOV → overlapping L1B crops



L1BSR Architecture

L1BSR uses the second overlapping image as target $\Longrightarrow$ the network exploits alias patterns in the LR images to recover the high-frequency details in the HR.

- LR Input I with band-misalignment.
- HR Output $\widehat{I}^{HR} = \textbf{REC}(I)$ with all 4 bands aligned with the green band of I.
- **Cross Spectral Registration** must be pretrained but it can also be done with self-supervision



Conclusions

We've seen self-supervised training and fine-tuning for

- Multi-frame video denoising networks [Ehret'18], [Dewil'21]
- Joint denoising and demosaicking of image bursts [Ehret'19]
- Multi-frame super-resolution [Nguyen'21] and HDR fusion [Nguyen'22]
- Single-image super-resolution for Sentinel-2 imagery [Nguyen'23]

Benefits of self-supervised fine-tuning

- Model-blind image restoration
- Train on testing data: **no dataset bias**

Challenges of self-supervised training

- Dependent on good alignment and miss-alignment and masks
- Temporally correlated noise