Self-Supervised Super-Resolution for Multi-Exposure Push-Frame Satellites Supplementary material

Ngoc Long Nguyen[1] Jérémy Anger[1,2] Axel Davy[1] Pablo Arias[1] Gabriele Facciolo[1]
[1] Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, France [2] Kayrros SAS
https://centreborelli.github.io/HDR-DSP-SR/

A. Weights for the base fusion

Since the base component only contains low frequencies and cannot be super-resolved, we propose a simple pipeline consisting of *i)* alignment of the LR base components B_i to the reference, *ii)* temporal fusion via weighted average to attenuate noise, *iii)* upscaling using bilinear interpolation. For the temporal fusion the weights in the weighted average are simply the exposure times:

$$B^{LR}(x) = \frac{\sum_i e_i \text{Warp}(B_i^{LR}(x))}{\sum_i e_i} \tag{S1}$$

In this section we will provide a justification for this choice, which is based on two approximations.

Approximate noise model for the base. The base results from the convolution with a Gaussian kernel G. At pixel x we have

$$B_i^{LR}(x) = \sum_h G(h) I_i^{LR}(x+h).$$

Assuming the signal-dependent Gaussian noise model of Eq. (2)[1], we have that $B_i^{LR}(x)$ also follows a Gaussian distribution with the following mean and variance:

$$\mathbb{E}\{B_i^{LR}(x)\} = \sum_h G(h)\mathcal{I}_i^{LR}(x+h)$$
$$\mathbb{V}\{B_i^{LR}(x)\} = \frac{a}{e_i}\sum_h G^2(h)\mathcal{I}_i^{LR}(x+h) + \frac{b}{e_i^2}\sum_h G^2(h).$$

We are going to assume that the clean LR image $\mathcal{I}_i^{LR}$ varies smoothly in the filter support, and thus

$$\mathbb{E}\{B_i^{LR}(x)\} \approx \mathcal{I}_i^{LR}(x), \quad \mathbb{V}\{B_i^{LR}(x)\} \approx \frac{\alpha e_i \mathcal{I}_i^{LR}(x) + \beta}{e_i^2}. \tag{S2}$$

where $\alpha = a\sum_h G^2(h)$ and $\beta = b\sum_h G^2(h)$. This rough approximation allows us to use a signal-dependent Gaussian noise model like (2). The approximation is only valid in regions where the image is smooth (away from edges, textures, etc.). However, these are the regions in which we are mainly interested, since it is where the low frequency noise present in the base becomes more noticeable.

Approximate MLE estimator for the weights. After alignment, for a given pixel x we have different values acquired with varying exposure times, which we are going to denote as $z_i = \text{Warp}(B_i^{LR}(x))$ to simplify notation. We also have the corresponding clean LR base images $\mathcal{B}_i^{LR}$, and we are going to assume that they coincide after alignment, i.e. $y = \text{Warp}(\mathcal{B}_i^{LR})(x)$ for $i = 1, ..., m$. We would like to estimate y from the series of observations

$$z_i \sim \mathcal{N}\left(y, \sigma_i^2(y)\right), \quad \sigma_i^2(y) = \frac{\alpha e_i y + \beta}{e_i^2}.$$

This problem occurs in HDR imaging, when estimating the unknown irradiance given noisy acquisitions with varying exposure times [1, 6]. Each z_i is an unbiased estimator of y. Therefore, if the variances were known, we can minimize the MSE with the following weighted average, where the weights are the inverse of the variances:

$$\hat{y} = \frac{\sum_i w_i z_i}{\sum_i w_i}, \quad w_i = \frac{e_i^2}{\alpha e_i y + \beta}. \tag{S3}$$

The problem is that the weights depend on the unknown y. In [6] Granados et al. solve this problem with an iterative weighted average:

$$w_i^0 = \frac{e_i^2}{\alpha e_i z_i + \beta}.$$
$$w_i^k = \frac{e_i^2}{\alpha e_i \hat{y}^k + \beta}, \quad \hat{y}^{k+1} = \frac{\sum_i w_i^k z_i}{\sum_i w_i^k}, \quad k = 1, 2, ...$$

It can be shown that this converges to the maximum likelihood estimate.

[1] Tables, figures and equations in the supplementary material are labeled S1, S2, . . . to differentiate them from references to the main paper.

In our case, we are going to simplify expression (S3) by assuming that $\alpha e_i y \gg \beta$, and therefore $w_i \approx \frac{e_i}{\alpha y}$. Under this assumption, we obtain

$$\hat{y} = \frac{\sum_i e_i z_i}{\sum_i e_i}. \tag{S4}$$

This assumption holds for brighter pixels and well exposed images [1].

B. HDR-DSP architecture

Our HDR-DSP architecture has 3 trainable modules: Motion estimator, Encoder, and Decoder. The Feature Shift-and-Pool block does not have any trainable parameters. Our motion estimator follows the work of [17]. Our encoder and our decoder are inspired from the SRResNet architecture [10], and built from the residual blocks (see Table S1). Convolutions of the encoder and decoder are performed using reflection padding. In total, our networks have 2853411 trainable parameters (Table S2).

Table S1. `ResBlock(N)`

Input	Tensor N channels
Layer 1	`Conv2d(in=N, out=N, k=3, s=1, p=1)`
	`ReLU`
Layer 2	`Conv2d(in=N, out=N, k=3, s=1, p=1)`
Output	Layer 2 + Input

Table S2. HDR-DSP Architecture

Modules	**Layers**	**Nb parameters**
Motion Estimator	`FNet` [17]	1744354

Encoder	`Conv2d(in=1, out=64, k=3, s=1, p=1)`
	`ResBlock(64)` ×4
	`ReLU`
	`Conv2d(in=64, out=64, k=3, s=1, p=1)`
		332992

FS&P		0

Decoder	`Conv2d(in=64, out=64, k=3, s=1, p=1)`
	`ResBlock(64)` ×10
	`ReLU`
	`Conv2d(in=64, out=1, k=3, s=1, p=1)`
		776065

		Total: 2853411

C. Training details

We train HDR-DSP in two stages, first pretraining the motion estimator and then the end-to-end system.

Phase 1: Pre-train the Motion Estimator. Training the motion estimator in the case of images obtained with different exposures is a challenging task. We first pretrain it on the simulated dataset to ensure that it produces accurate flows. We monitor the quality of the estimations by comparing with the ground truth flows, until reaching an averaged error of 0.05 pixel.

For training the motion estimator our first choice was to use the L_1 distance between the reference image and the radiometrically corrected warped image. However, the quality of the estimated flows were not acceptable (with errors above 0.1 pixel). Indeed, since motion estimation relies on the photometric consistency between frames, it is very sensitive to the intensity fluctuations between frames (as it is the case for our normalized LR frames I_i^{LR}), which results in imprecise alignments.

To prevent this issue we compute the warping loss on the details rather than on the images, which is common in traditional optical flow [11, 17]. The loss is computed for each flow $F_{i\to r}$ estimated by the **MotionEst** module

$$\begin{aligned}&\ell_{me}(\{F_{i\to r}\}_{i=1}^{m}) = \\ &\sum_i \|\textbf{Detail}(I_i^{LR}) - \textbf{Detail}\left(\textbf{Pullback}(I_r^{LR}, F_{i\to r})\right)\|_1 \\ &\qquad\qquad + \lambda_1 TV(F_{i\to r}),\end{aligned} \tag{S5}$$

where **Pullback** computes a bicubic warping of I_r^{LR} according to a flow, **Detail** applies a high-pass filter to the images, TV is the finite difference discretization classic Total Variation regularizer [16], and $\lambda_1 = 0.003$ is a hyperparameter controlling the regularization strength.

We set the batch size to 32 and use Adam [9] with the default Pytorch parameters and a initialized learning rate of 10^{-4} to optimize the loss. The pre-training converges after 50k iterations and takes about 3 hours on one NVIDIA V100 GPU.

Phase 2: Train the whole system end-to-end. We then use the pretrained motion estimator and train the entire system end-to-end using the total loss:

$$\text{loss} = \ell_{self} + \lambda_2 \ell_{me}. \tag{S6}$$

We set $\lambda_2 = 3$ in our experiments. Furthermore, to avoid boundary issues, the loss does not consider values at a distance below 2 pixels from the border of the frames.

We train our model on LR crops of size 64 × 64 pixels and validate on LR images of size 256 × 256 pixels. During training, our network is fed with a random number of LR input images (from 4 to 14) in each sequence. We set the batch size to 16 and optimize the loss using the Adam optimizer with default parameters. The learning rates are initialized to 10^{-4}, then scaled by 0.3 each 400 epochs. The training takes 20h (1200 epochs) on one NVIDIA V100 GPU.

D. Trainable feature pooling alternative

The feature pooling block FSP described in Section 4.1 of the main article does not have any trainable parameters. In this section we investigate the use of a trainable layer, named PoolNet, for performing this task.

We considered a simple trainable network PoolNet that performs feature pooling (Table S3) instead of statistical feature poolings (Avg-Max-Std) as presented in the paper. To this aim, PoolNet takes as input the concatenation of N features J_i^{HR} and N weights W_i^{HR} (computed by the SPMC [19] module from N LR images) and produces the fused HR features. Then, the Decoder network reconstructs the HR detail image from the fused features.

A drawback of PoolNet is that it can only be applied on a fixed number of frames. Table S4 compares the performance of the PoolNet (which replaces the Avg-Max-Std feature pooling) trained on 4 and 14 frames with our original method. We can see that in the case of small number of frames, PoolNet attains a performance comparable to our HDR-DSP method using the Avg-Max-Std feature pooling. However, in the case of 14 frames, there is a big gap of 0.3 dB between our method and PoolNet. It seems that it is more difficult for PoolNet to capture the necessary statistics from many features.

E. Alternative exposure weighting strategies

As discussed in the main paper, the LRs with longer exposure time should contribute more to the reconstruction because of their high signal-to-noise ratio. In our proposed method, we use the un-normalized LR images as additional input to the Encoder so as that the Encoder perceives the noise level in each LR image. Subsequently, the Encoder can decide which features are more important.

We also evaluated an alternative strategy to weight the features (WF) based on the exposure times. This simply consists in weighting the features J_i^{LR} by the corresponding exposure time in the SPMC module. Actually, this was inspired from the ME S&A method.

This strategy leads to slightly worse yet adequate feature encodings (-0.08dB) as shown in Table S5. Moreover, using both feature weighting and LRs encoding (third column) leads to the same performance as only using LRs encoding. This implies that the Encoder already encodes the necessary information about the signal-dependent noise on the features.

F. Adaptation of existing methods to multi-exposure sequences

We detail here the adaptations to the algorithms we used in the comparisons.

ME S&A. *Multi-exposure Shift-and-add* is a weighted version of the classic shift-and-add method [5, 7, 8, 12] designed for multi-exposure sequences. Usually, S&A produces the HR image by registering the LR images onto the HR grid using the corresponding optical flows. After the registration step, the intensities of the LR images are splatted to the neighborhood integer-coordinate pixels using some kernel interpolation. Finally, pixel-wised aggregation is done to obtain the HR output image. Therefore a naive method consists of using the classic S&A method on the normalized LR images. However this ignores the different signal-to-noise ratios in the normalized images and fails to greatly reduce the noise. Using the same arguments as in the Sec. A, we propose the weighted S&A for multi-exposure sequence

$$\widehat{I}^{HR} = \frac{\sum_{i=1}^{m} \textbf{Register}(\bar{I}_i^{LR})}{\sum_{i=1}^{m} e_i} \tag{S7}$$

where **Register** maps and splats the un-normalized images $\bar{I}_i^{LR}$ onto the HR grid.

Base-detail ACT (BD ACT). ACT [2] is a traditional multi-image super-resolution method developed for Planet SkySat single-exposure sequences. It formulates the reconstruction as an inverse problem and solves it by an iterative optimization method. BD ACT extends ACT to support multi-exposure images by adopting the same base-detail strategy as proposed in HDR-DSP: the details of the images are fused by ACT, and the base is reconstructed by the upsampled average of the bases of the input images.

HighRes-net (HR-net) and RAMS. HighRes-net [4] and RAMS [18] are two super-resolution methods for multi-temporal PROBA-V satellite images. However in the PROBA-V dataset, the identity of the LR reference image is unavailable. This hinders the true potential of the methods trained on this dataset. As a result we use the reference-aware super-resolution [14] of HighRes-net and RAMS. In HighRes-net, the reference image is used as a shared representation for all LR images. Each LR image is embedded jointly with this reference before being recursively fused. In RAMS, each LR image is aligned to the reference image before being input to the residual attention block. The registration step of RAMS is done with inverse compositional algorithm [3], which is robust to noise and brightness change. As HighRes-net and RAMS are supervised methods, we also use a radiometric correction on the output before computing the loss [4].

DSA. *Deep shift-and-add* [15] DSA is a self-supervised method for super-resolution of push-frame single-exposure satellite images. We adapt DSA to multi-exposure case by

Table S3. PoolNet(N) architecture for trainable feature pooling.

Input	N Features (64 channels) + N Weights (1 channel)
	Conv2d(in=N(64+1), out=256, k=1, s=1, p=0)
	ReLU
	Conv2d(in=256, out=128, k=1, s=1, p=0)
	ReLU
	Conv2d(in=128, out=64, k=1, s=1, p=0)
	ReLU
Output	fused HR feature

Table S4. Trainable feature pooling evaluation. Since the trainable feature pooling networks PoolNet only accepts a fixed number of frames (in this case 4 and 14) we compare it with HDR-DSP also trained with fixed number of frames.

Methods	HDR-DSP	HDR-DSP 4	HDR-DSP 14	PoolNet 4	PoolNet 14
PSNR(dB) 4 frames	**52.81**	52.69	51.31	52.76	N/A
PSNR(dB) 14 frames	**55.85**	54.26	55.53	N/A	55.55
PSNR(dB) variable n frames	**54.70**	53.85	54.07	N/A	N/A

Table S5. Handling of the signal-dependent noise

Methods	HDR-DSP	DSP (+WF - LR)	DSP (+WF)
PSNR(dB) ME	**54.70**	54.62	54.70

using the normalized LR images as input. We also use the loss on the details to train the motion estimator in DSA.

G. Execution time

Table S6 reports the execution time of the methods studied on the synthetic multi-exposure dataset. Due to its convolutional architecture, HighRes-net is the fastest. HDR-DSP is slightly more costly than DSA since it performs feature pooling instead of a simple average and requires fusing the bases together. ME S&A and BD-ACT are both executed on CPU, the later being quite costly due to the linear spline system inversion.

H. Additional comparisons using real SkySat sequences

Figure S1 presents results obtained on real multi-exposure SkySat images using 9 frames. This is a challenging sequence as it contains moving vehicles. Note how the road markings are better seen in the HDR-DSP result. However, since HDR-DSP does not account for moving objects (the motion estimator only predicts smooth motion within a range of 5 pixels) the cars are blurry.

Figure S2 shows another example of reconstruction on a real sequence of 7 SkySat images. Even though there are only 7 images in this sequence and most of them are very noisy, HDR-DSP is able to produce a clean image. The fine details are well restored.

I. Exposure error analysis

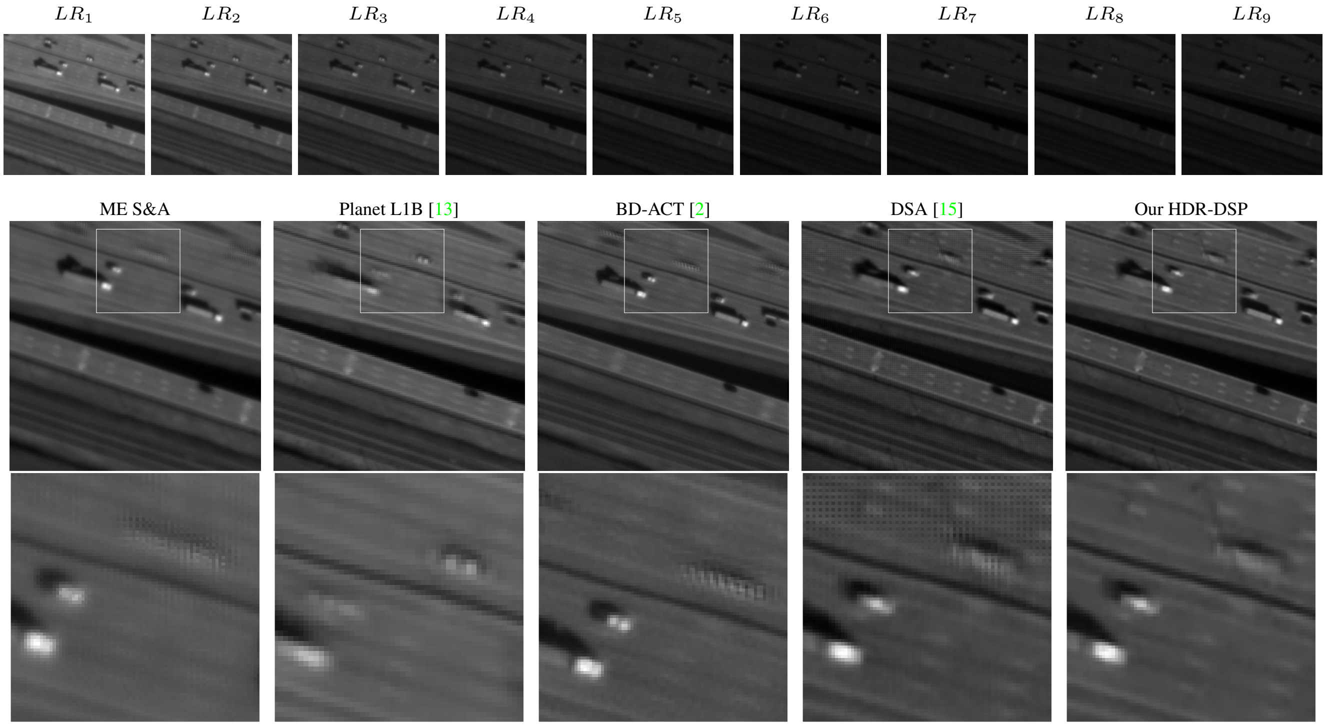
We observed a discrepancy between the reported exposure time by Planet and correct normalization ratios. This can be explained by measurement imprecision since the quantities are in sub-milliseconds range, or by local illumination effects such as vignetting. To estimate the correct exposure ratio for a given pair of images, we registered the images using phase correlation, masked saturated pixels and computed the spatial median of the ratio between the two frames. We then validated visually that such exposure ratios were more precise that the reported exposure time (less flicker was observed).

Figure S3 shows the relation between the reported ratio, and the estimated one. We find that errors are usually in the order of a few percent, but also observe larger errors. The nominal exposure times range from 0.4ms to 4.5ms. Note that the absolute error in exposure the time measurement is probably constant regardless of the exposure time. However, when computing the ratio of two exposures with errors this might result in a large divergence of the ratio, especially if the exposure in the denominator is a short one.

Note that for the proposed super-resolution method, we used the imprecise, reported exposure times and not the estimated one, as the estimation method itself can fail.

Table S6. Execution time (s) on 200 sequences of size 15 × 256 × 256 pixels.

Methods	RAMS	ME S&A	HR-net	BD-ACT	DSA	HDR-DSP
Time (s)	93	276	22	555	82	97



ME S&A Planet L1B [13] BD-ACT [2] DSA [15] Our HDR-DSP

Figure S1. Super-resolution from a real multi-exposure sequence of 9 SkySat images. Top row: Original low resolution images with different exposures. Middle row: Reconstructions from five methods, including ours trained with self-supervision (right). Bottom row: Zoom on a detail of the results.

References

[1] Cecilia Aguerrebere, Julie Delon, Yann Gousseau, and Pablo Musé. Best algorithms for hdr image generation. a study of performance bounds. *SIAM Journal on Imaging Sciences*, 7(1):1–34, 2014. 1, 2

[2] Jérémy Anger, Thibaud Ehret, Carlo de Franchis, and Gabriele Facciolo. Fast and accurate multi-frame super-resolution of satellite images. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 5(1), 2020. 3, 5, 6

[3] Thibaud Briand, Gabriele Facciolo, and Javier Sánchez. Improvements of the Inverse Compositional Algorithm for Parametric Motion Estimation. *IPOL*, 8:435–464, 2018. 3

[4] Michel Deudon, Alfredo Kalaitzis, Israel Goytom, Md Rifat Arefin, Zhichao Lin, Kris Sankaran, Vincent Michalski, Samira E Kahou, Julien Cornebise, and Yoshua Bengio. Highres-net: Recursive fusion for multi-frame super-resolution of satellite imagery. arxiv 2020. *arXiv preprint arXiv:2002.06460*, 2020. 3

[5] Andrew Fruchter and Richard Hook. Drizzle: A method for the linear reconstruction of undersampled images. *Publications of the Astronomical Society of the Pacific*, 114(792):144, 2002. 3

[6] Miguel Granados, Boris Ajdin, Michael Wand, Christian Theobalt, Hans-Peter Seidel, and Hendrik PA Lensch. Optimal hdr reconstruction with linear digital cameras. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 215–222. IEEE, 2010. 1

[7] Thomas J. Grycewicz, Stephen A. Cota, Terrence S. Lomheim, and Linda S. Kalman. Focal plane resolution and overlapped array TDI imaging. In *Remote Sensing System Engineering*, volume 7087, page 708704. International Society for Optics and Photonics, 2008. 3

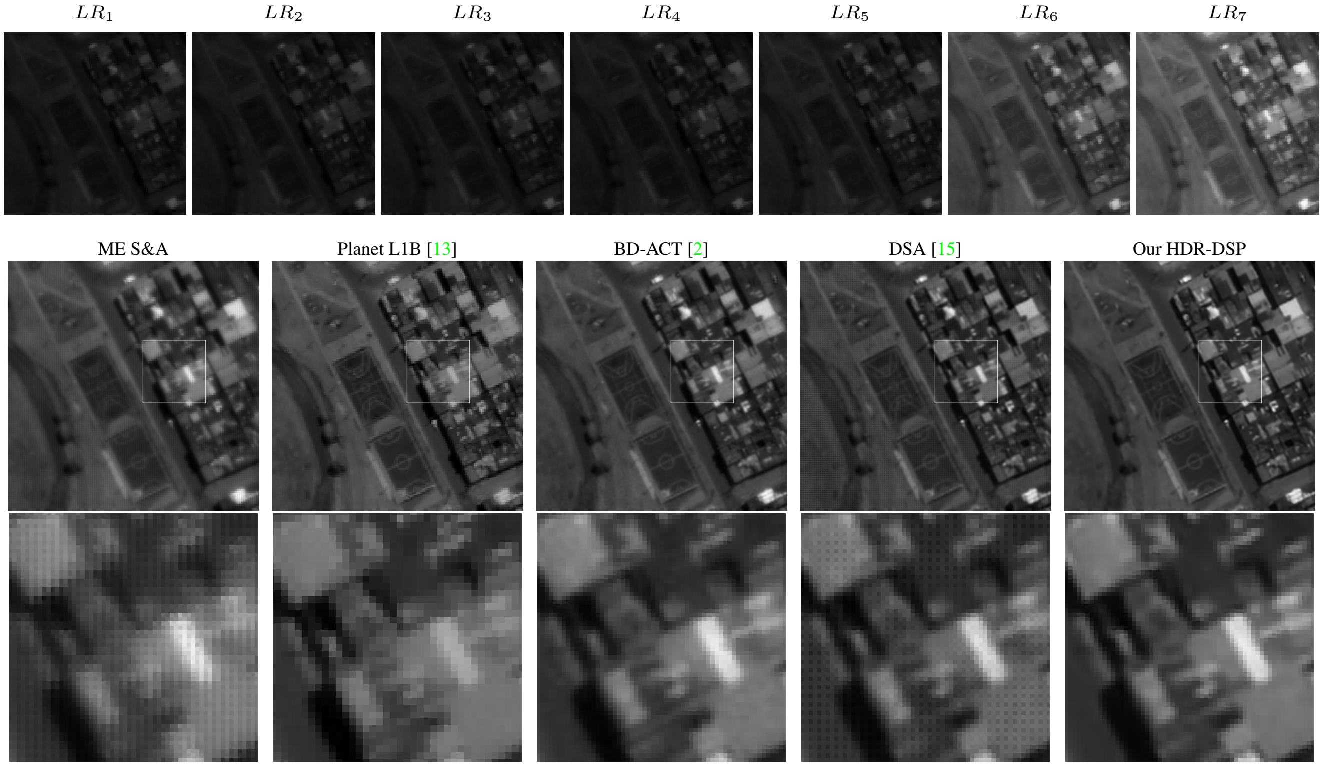


Figure S2. Super-resolution from a real multi-exposure sequence of 7 SkySat images. Top row: Original low resolution images with different exposures. Middle row: Reconstructions from five methods, including ours trained with self-supervision (right). Bottom row: Zoom on a detail of the results.

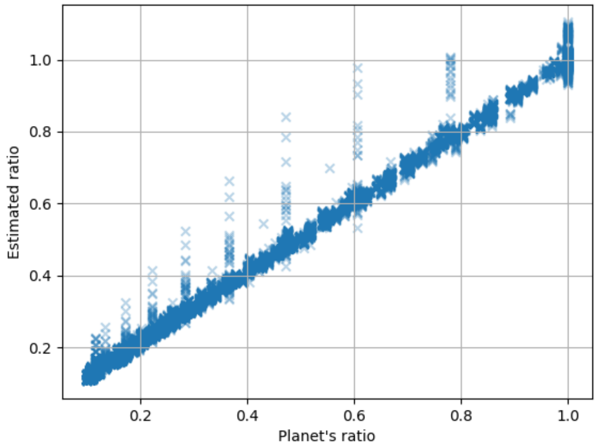


Figure S3. Normalized estimated exposure ratio with respect to provided exposure time.

[8] Yunwei Jia. Method and apparatus for super-resolution of images, Nov. 6 2012. US Patent 8,306,121. 3

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[10] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2

[11] Pengpeng Liu, Michael Lyu, Irwin King, and Jia Xu. Selflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[12] Maria Teresa Merino and Jorge Nunez. Super-resolution of remotely sensed images with variable-pixel linear reconstruction. *IEEE TGRS*, 45(5):1446–1457, 2007. 3

[13] Kiran Murthy, Michael Shearn, Byron D. Smiley, Alexandra H. Chau, Josh Levine, and Dirk Robinson. SkySat-1: very high-resolution imagery from a small satellite. In *Sensors, Systems, and Next-Generation*

Satellites XVIII, volume 9241, page 92411E. International Society for Optics and Photonics, 2014. 5, 6

[14] Ngoc Long Nguyen, Jérémy Anger, Axel Davy, Pablo Arias, and Gabriele Facciolo. PROBA-V-REF: Repurposing the PROBA-V Challenge for Reference-Aware Super Resolution. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 3881–3884. IEEE, jul 2021. 3

[15] Ngoc Long Nguyen, Jérémy Anger, Axel Davy, Pablo Arias, and Gabriele Facciolo. Self-supervised multi-image super-resolution for push-frame satellite images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1121–1131, June 2021. 3, 5, 6

[16] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992. 2

[17] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, 2018. 2

[18] Francesco Salvetti, Vittorio Mazzia, Aleem Khaliq, and Marcello Chiaberge. Multi-image super resolution of remotely sensed images using residual attention deep neural networks. *Remote Sensing*, 12(14):2207, 2020. 3

[19] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4472–4480, 2017. 3