Super-resolution of Sentinel-2 Images with Generative Adversarial Networks

Master Thesis Defense

Run Zhang, RWTH Aachen University, Jülich Supercomputing Center

First thesis examiner: Prof. Dr. Bastian Leibe
Second thesis examiner: Prof. Dr. Morris Riedel
Supervisor: Dr. Garbriele Cavallaro, Dr. Jenia Jitsev

Roadmap

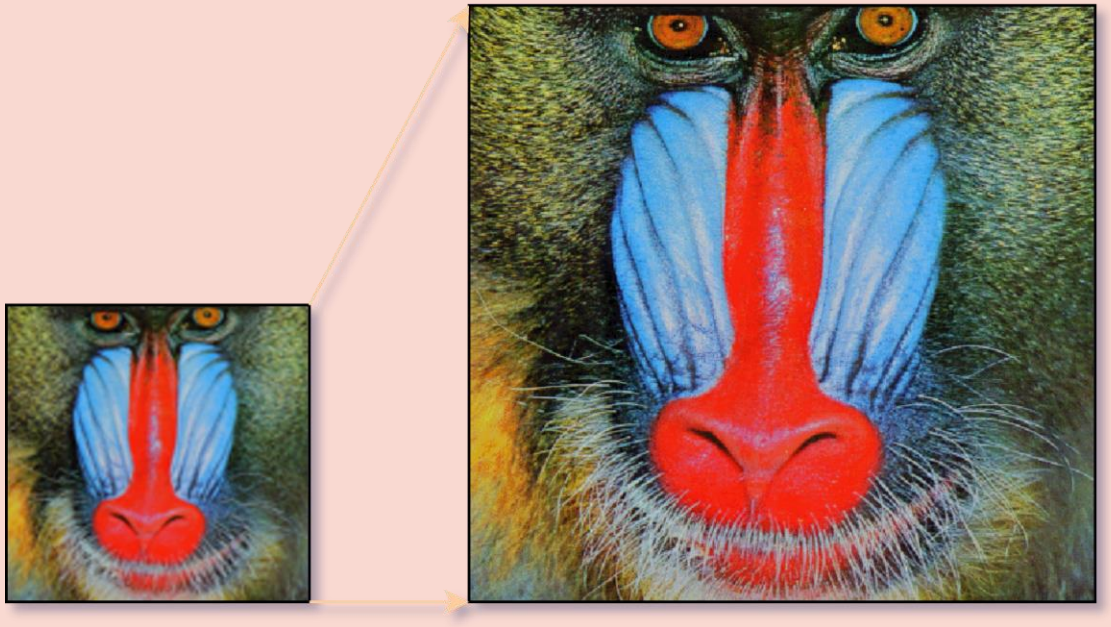
Background, problem defination

Methodology, evaluation results

Conclusion, future directions

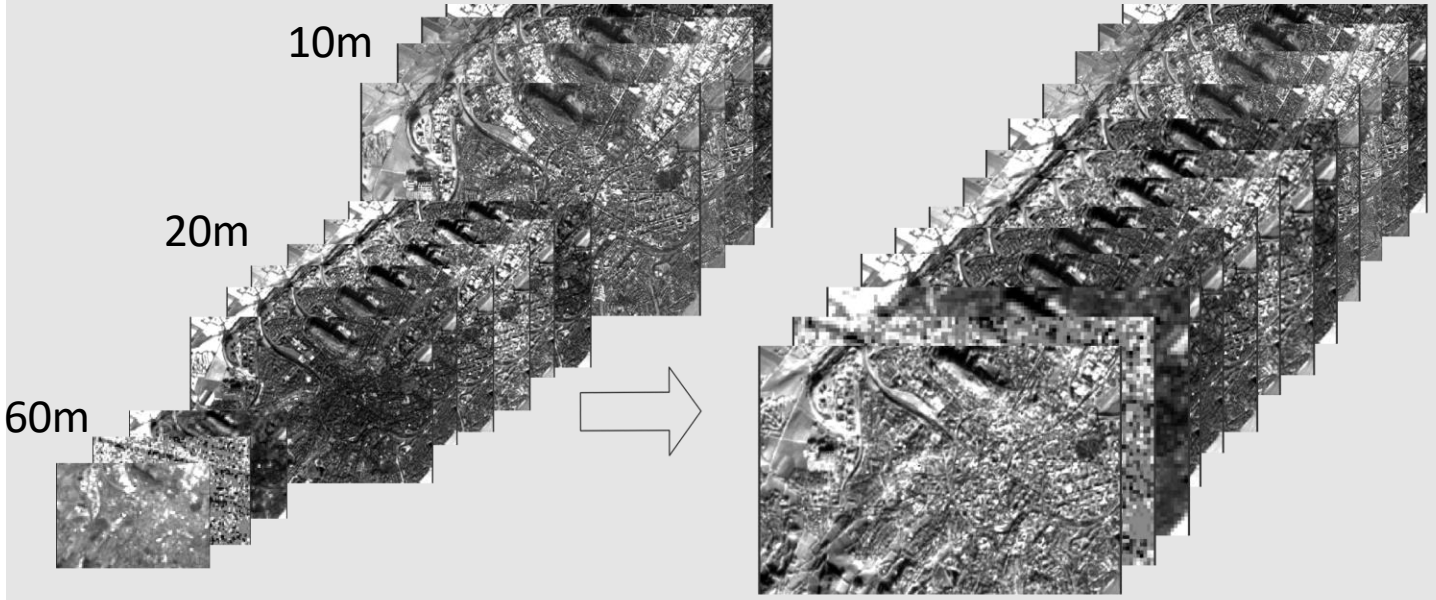
Questions

(Image) super-resolution



- Construct HR images
- Add high-frequency components (details) or remove degradation
- Ill-posed problem!!

Sentinel-2 image SR



- Multi-resolution multi-spectral images
- Super-resolution = to obtain a complete HR data cube
- Max ground sampling distance~10m
- Ill-posed problem!

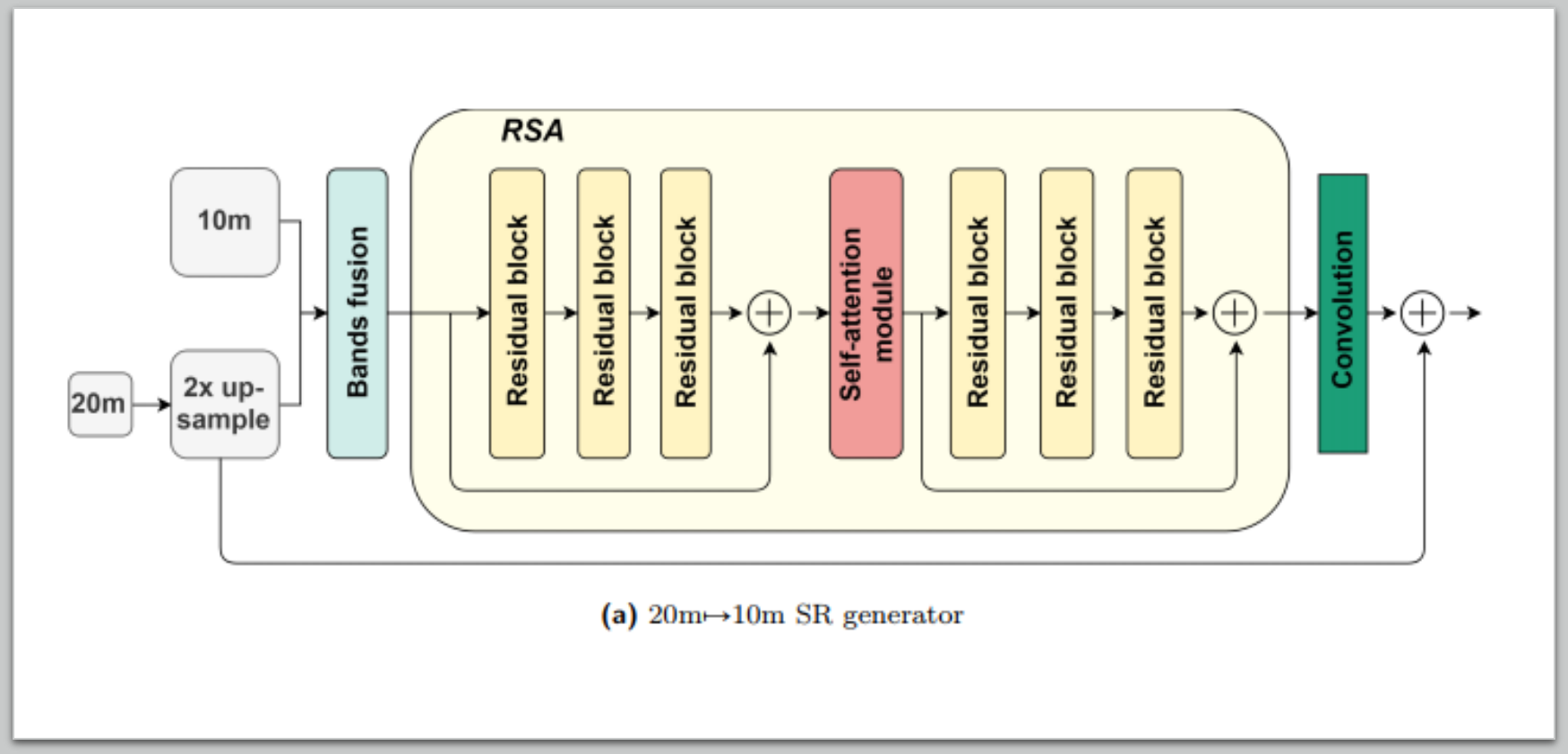
Methodology

NETWORK ARCHITECTURE

COMPREHENSIVE EVALUATION

DISTRIBUTED LEARNING

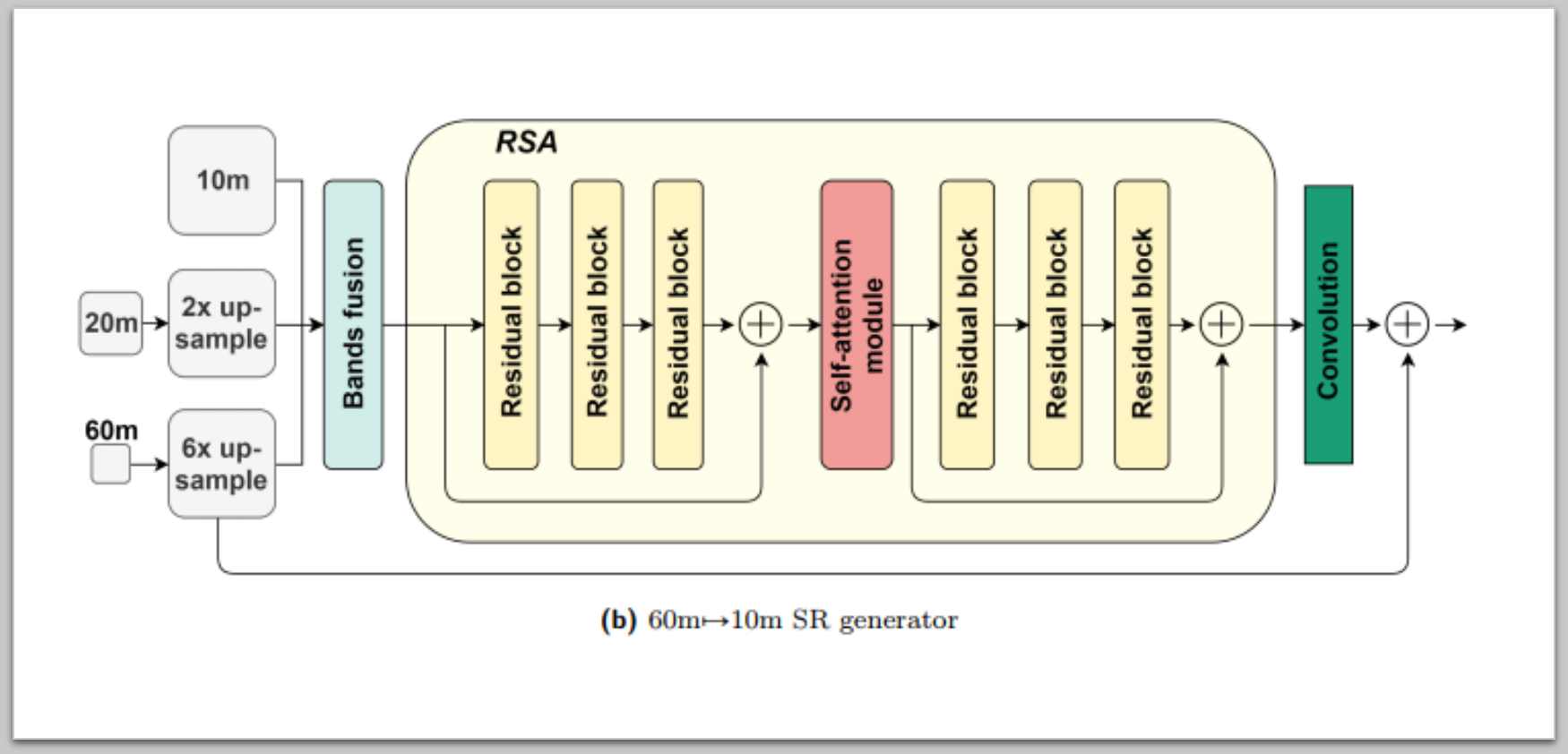
ADVERSARIAL TRAINING



(a) 20m↦10m SR generator

Network architecture

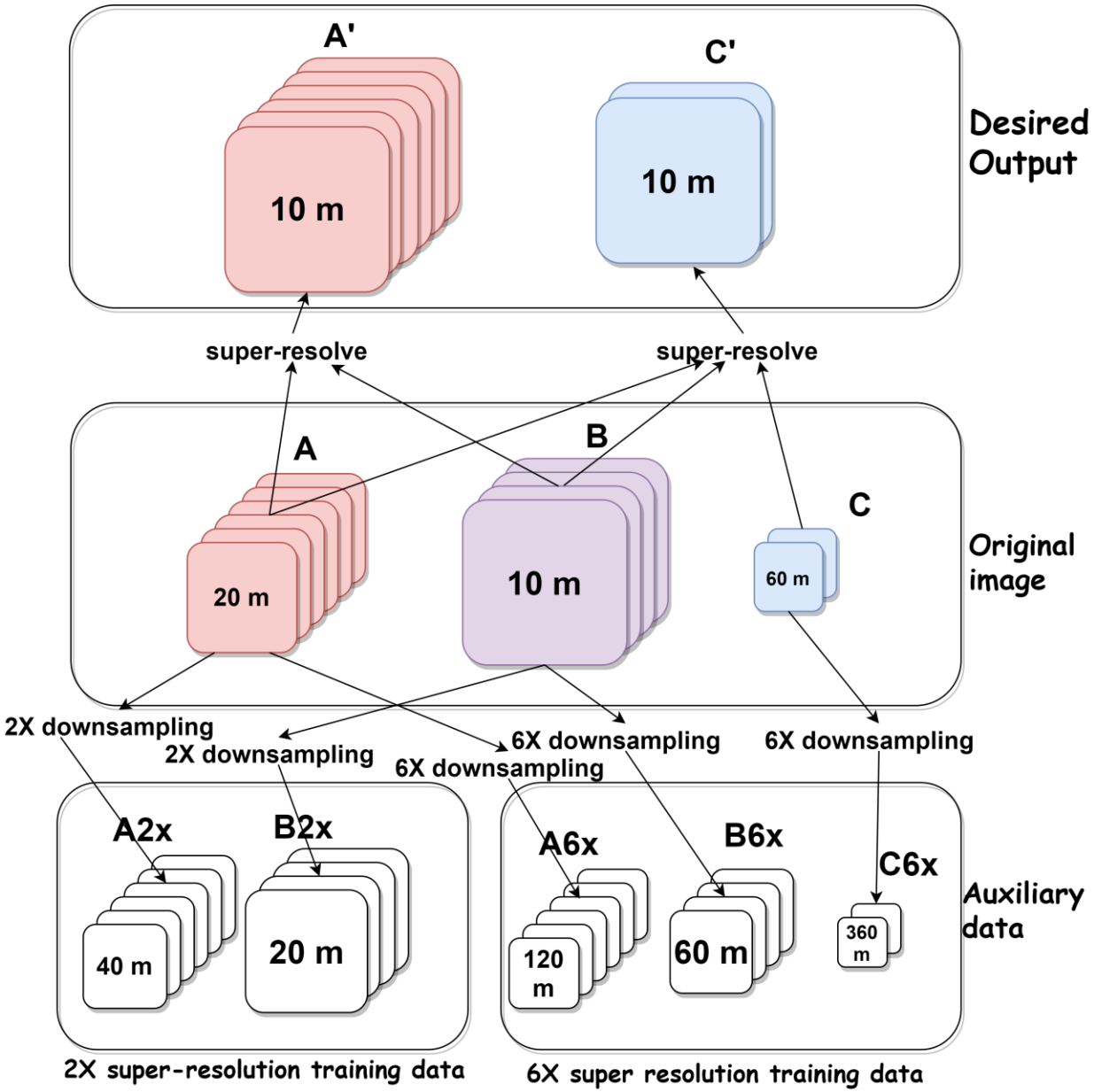
1. Inputs with multiple resolution, blend in information from different spectrum;
2. Six residual blocks show fairness with comparable DSen2 on model complexity;
3. Residual learning enables a sparse feature space and a fast convergence speed;



(b) 60m↦10m SR generator

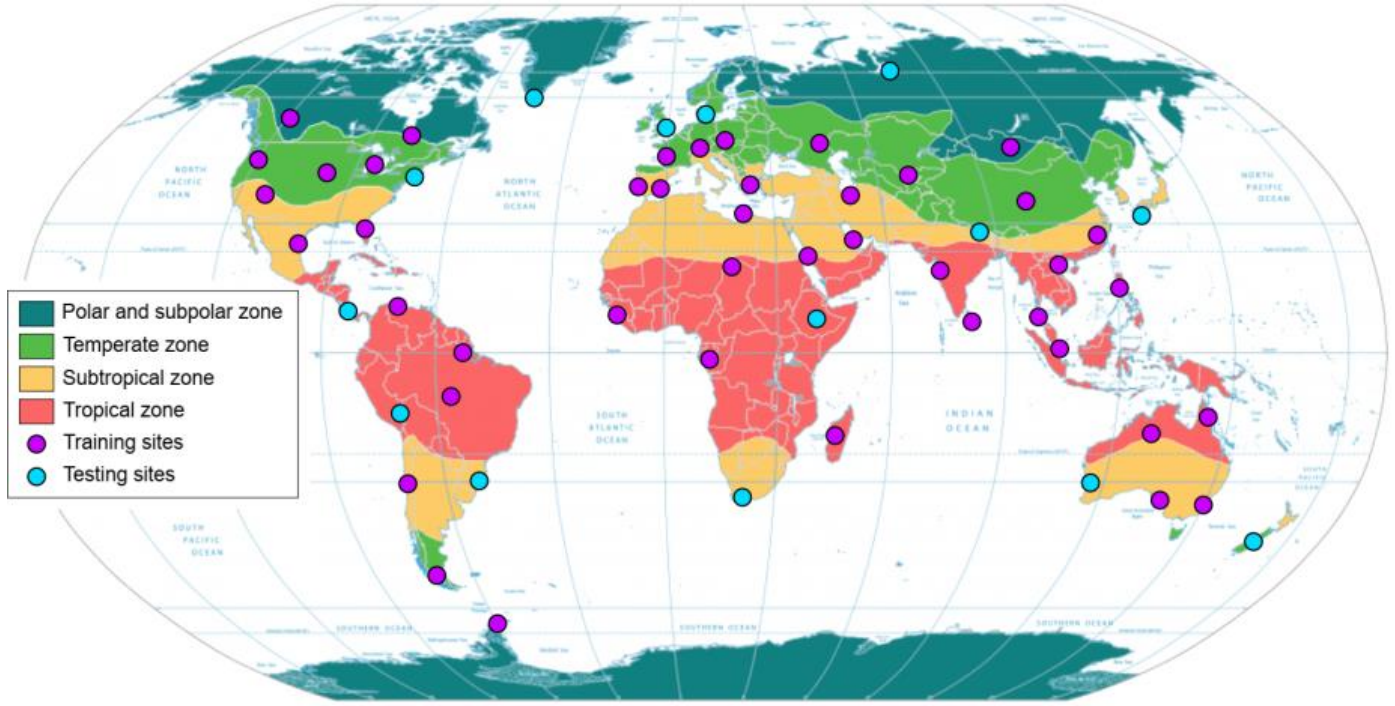
Network architecture

1. one more input branch;
2. one more branch in the band fusion module;



Synthetical training

- Wald's protocol
- 20m to 10m mapping is learned from 40m to 20m, same for the case of 60m to 10m.
- The resolution scale invariance can be ***strengthened*** if the decimation filter is matched to the modulation transform function (MTF) of the image sensor.



DSen2: Lanaras et.al [9]

Name	High-level Description	Production and Distribution	Data Volume
Level-1B	Top-of-atmosphere radiances in sensor geometry	Systematic generation and online distribution	27 MB (each 25*23 km^2)
Level-1C	Top-of-atmosphere reflectances in cartographic geometry	Systematic generation and online distribution	500 MB (each 100*100 km^2)
Level-2A	Bottom-of-atmosphere reflectances in cartographic geometry	Generation on user side	600 MB (each 100*100 km^2)

Data collection

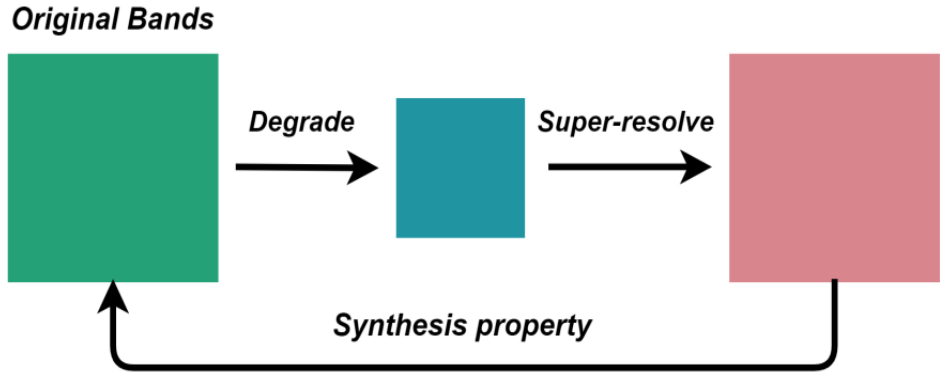
- Worldwide random selection
- Cover all climate zones
- 45 for training and 15 for testing (1.5 million patches)
- Level 1C – Level 2A conversion by Sen2cor [26]

$$\mathcal{L}_{L1}(\hat{HR}, HR) = \frac{1}{HWC}||HR_{i,j,c} - \hat{HR}_{i,j,c}||$$

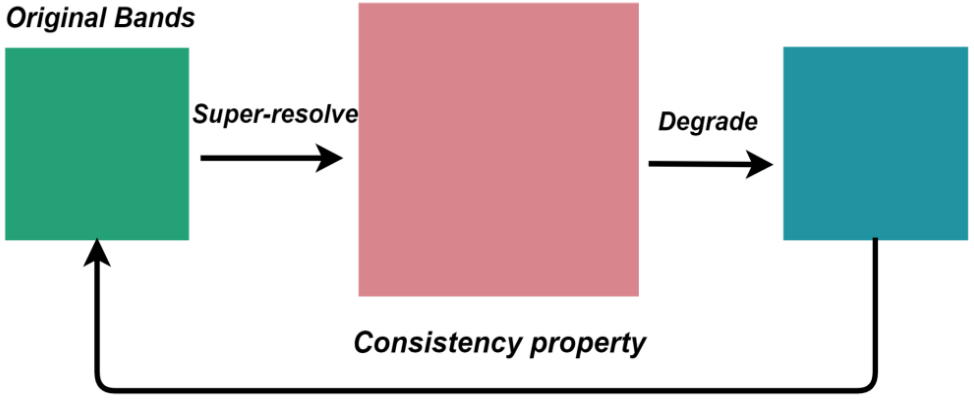
$$\mathcal{L}_{Charb}(\hat{HR}, HR) = \frac{1}{HWC}\sqrt{(HR_{i,j,c} - \hat{HR}_{i,j,c})^2 + \epsilon^2}$$

Loss functions

- Pixel-wise loss (L1 loss)
- L2 loss is not used due to the blurry output.
- Variant: Charbonnier loss



(a) Synthesis property



(b) Consistency property

Evaluation metrics

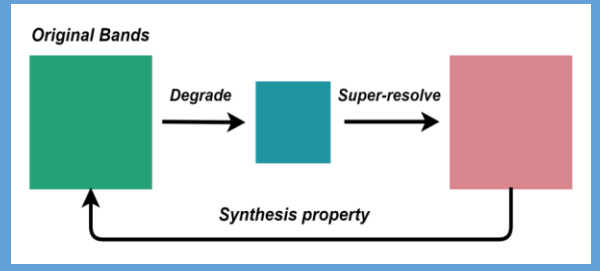
- Synthesis property
- Consistency property
- Quantitative metrics：

1. Rooted means square error (RMSE)
2. Erreur Relative Globale Adimensionnelle de Synthese（ERGARS）
3. Spectral angle mapper (SAM)
4. Signal reconstruction error (SRE)
5. Structural similarity index (SSIM)
6. Peak signal to noise ratio (PSNR)
7. **Brightness equalized PSNR (bPSNR)**

Super-resolution of L1C 20m bands (synthesis)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	125.69	25.64	1.22	3.48	0.82	44.9998	45.0003
ATPRK	116.2	25.7	1.68	-	-	-	-
SupReME	69.7	29.7	1.26	-	-	-	-
Superres	66.2	30.4	1.02	-	-	-	-
DSen2	35.85	35.94	0.78	1.07	0.9322	55.5416	55.9317
$\mathcal{S}_{2\times}^{L1C}$	**34.99**	**36.19**	**0.75**	**1.03**	**0.9336**	**55.7756**	**56.3358**

Table 6.5: Average performance of super-resolving 6 $20m$ bands in B by pre-trained generator $\mathcal{S}_{2\times}^{L1C}$ in sense of synthesis property. Our method achieved the best result consistently over all evaluation metrics.

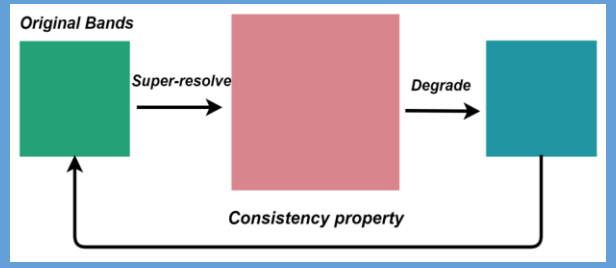


- Compared with Bicubic, reduce the RMSE by ***72%***
- Better performance consistently over all metrics
- A **0.1** improvement of PSNR and SRE can be considered as an effective improvement in the problem of image super-resolution

Super-resolution of L1C 20m bands (consistency)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	34.96	36.70	0.40	0.95	0.9849	56.1555	55.6129
DSen2	5.91	52.10	0.08	0.18	0.9899	71.9820	72.3170
$\mathcal{S}_{2\times}^{L1C}$	**4.28**	**55.11**	**0.07**	**0.1251**	**0.9901**	**74.9196**	**75.4208**

Table 6.10: Average performance of super-resolving each $20m$ bands in B by pre-trained generator $\mathcal{S}_{2\times}^{L1C}$ in sense of consistency property.

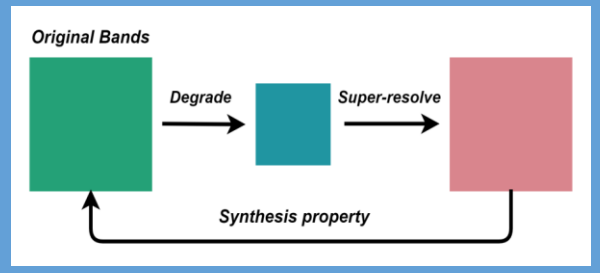


- Compared with Bicubic, reduce RMSE by ***87%***.
- Compared with DSen2, reduce RMSE by ***27%***.
- Better performance consistently over all metrics.

Super-resolution of L2A 20m bands (synthesis)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	149.75	24.51	2.10	3.93	0.8181	41.1114	41.1114
DSen2	47.60	34.27	1.83	1.32	0.91	50.85	52.73
DSen2-L2A	43.00	35.17	1.54	1.19	0.9265	51.7563	**54.1827**
$\mathcal{S}_{2\times}^{L2A}$	**41.45**	**35.55**	**1.46**	**1.14**	**0.9275**	**52.1107**	54.1475

Table 6.13: Average performance of super-resolving 6 $20m$ bands in B by pre-trained generator $\mathcal{S}_{2\times}^{L2A}$ in sense of synthesis property.

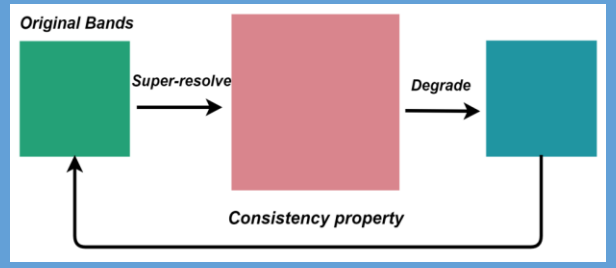


- Compared with Bicubic, reduce the RMSE by ***72%***.
- Improvement of DSen2-L2A compared with DSen2 (Dataset)
- Compared with DSen2-L2A, reduce the RMSE by ***4%***

Super-resolution of L2A 20m bands (consistency)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	41.81	35.83	0.72	1.08	52.25	0.9844	52.3023
DSen2-L2A	8.95	50.00	**0.28**	0.23	0.9819	66.1935	66.6093
$\mathcal{S}_{2\times}^{L2A}$	**5.65**	**53.87**	**0.28**	**0.14**	**0.9821**	**70.2786**	**70.8070**

Table 6.15: Average performance of super-resolving 6 $20m$ bands in B by pre-trained generator $\mathcal{S}_{2\times}^{L2A}$ in sense of consistency property.

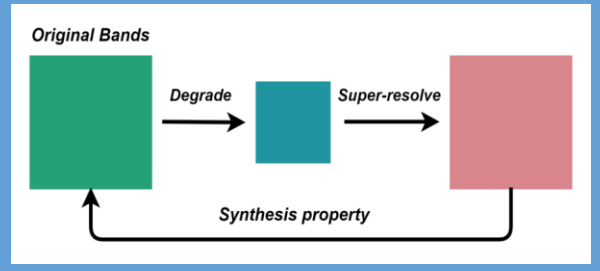


- Compared to naïve Bicubic, reduce the RMSE by ***85%***.
- Compared with DSen2-L2A, reduce the RMSE by ***37%***.
- Better performance over all metrics.

Super-resolution of L1C 60m bands (synthesis)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	161.85	19.79	1.78	7.30	0.36	37.6785	37.6785
ATPRK	145.1	20.4	1.62	-	-	-	-
SupReME	85.7	24.8	0.98	-	-	-	-
Superres	100.2	22.8	1.42	-	-	-	-
DSen2	28.11	34.47	0.36	1.38	0.8953	52.4984	52.1305
$\mathcal{S}_{6\times}^{L1C}$	**26.80**	**34.98**	**0.34**	**1.29**	**0.8991**	**52.9451**	**52.2735**

Table 6.7: Average performance of super-resolving 2 $60m$ bands in C by pre-trained generator $\mathcal{S}_{6\times}^{L1C}$ in sense of synthesis property. Our method achieved the best result consistently over all evaluation metrics.

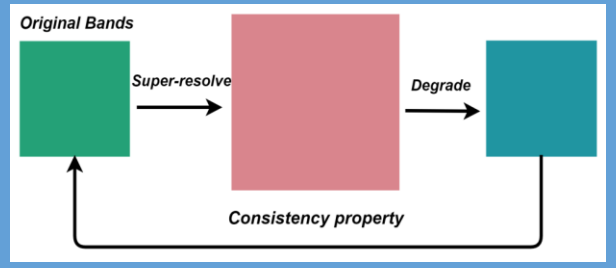


- Compared with bicubic interpolation, reduces the RMSE by ***83%***.
- Compared with DSen2, reduces the RMSE by ***4.7%***.
- Better performance consistently over all metrics.

Super-resolution of L1C 60m bands (consistency)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
Bicubic	**14.02**	**41.04**	**0.18**	**0.63**	**0.9919**	**58.9763**	**58.5900**
DSen2	24.69	36.31	0.20	1.03	0.9710	54.1462	54.6915
$\mathcal{S}_{6\times}^{L1C}$	22.35	37.41	0.17	0.92	0.9798	55.3258	55.5437

Table 6.11: Average performance of super-resolving 2 $60m$ bands in C by pre-trained generator $\mathcal{S}_{6\times}^{L1C}$ in sense of consistency property.



- Learning-based methods perform worse when having larger scale of super-resolution.
- Within the two learning-based methods, our model has better consistency.

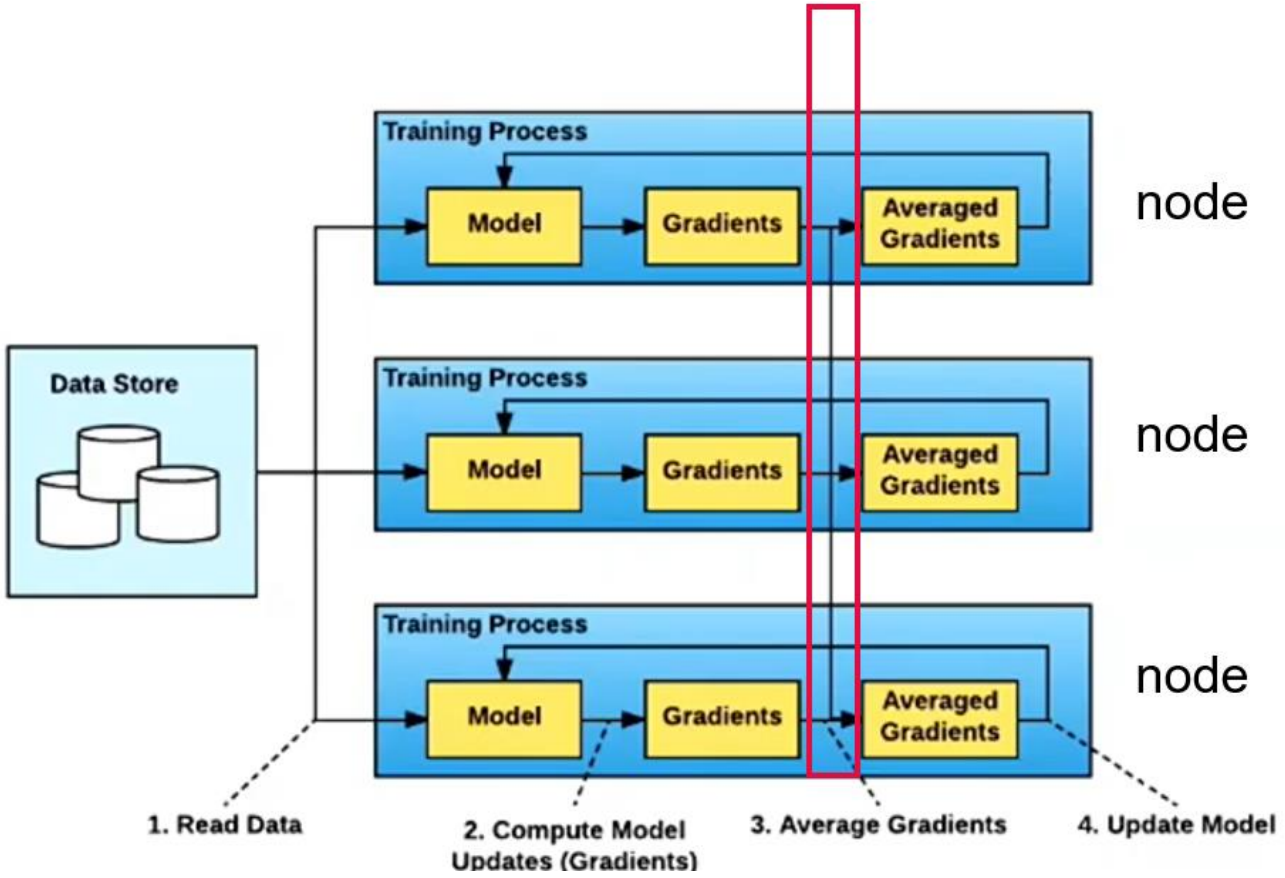
Andreas Herten, Introduction to GPU programming [lecture notes] retrived from https://www.fz-juelich.de/SharedDocs/Downloads/IAS/JSC/EN/slides/openacc/

Next: Booster!

JUWELS – Jülich's New Large System

- 2500 nodes with Intel Xeon CPUs (2 × 24 cores)
- 48 nodes with 4 NVIDIA Tesla V100 cards
- 10.4 (CPU) + 1.6 (GPU) PFLOP/s peak performance (Top500: #26)

Ring reduction mechanism



Data parallelism [27]

Synchronous data parallelism

- Multi-nodes data partitions
- × Central parameter servers
- √ Ring-reduction mechanism

Learning rate scaling

- **Definition 1**

Linear scaling rule [20]: when the mini-batch size is multiplied by k, multiply the learning rate by k

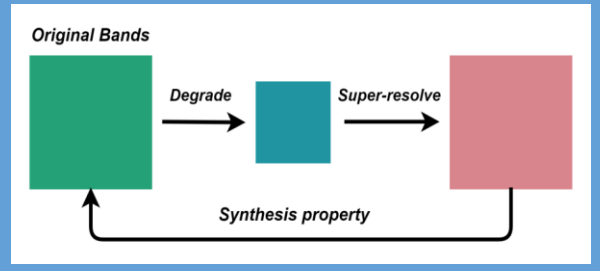
- **Definition 2**

Modified linear scaling rule: when the mini-batch size is multiplied by *k* (the number of nodes), multiply the initial learning rate by *k* and decay it to half every *n/k* SGD iterations, where *n* is a coefficient to control the decay rate.

Distributed Super-resolution of L1C 20m bands (synthesis)

Method	No. GPU	Batch size	Training time	RMSE	SRE	SAM	ERGAS	SSIM	PSNR
Bicubic	-	-	-	125.69	25.64	1.22	3.48	0.82	44.9998
DSen2	1	128	96h	35.85	35.94	0.78	1.07	0.9322	55.5416
$\mathcal{S}_{2\times}^{L1C}$	1	128	24h	36.42	35.83	0.78	1.08	0.9320	55.4393
$\mathcal{S}_{2\times}^{L1C}$	2	256	24h	35.67	36.03	0.77	1.06	0.9329	55.6199
$\mathcal{S}_{2\times}^{L1C}$	4	512	24h	**34.99**	**36.19**	**0.75**	**1.03**	**0.9336**	**55.7756**
$\mathcal{S}_{2\times}^{L1C}$	8	1028	12h	35.61	36.05	0.76	1.05	0.9329	55.6393
$\mathcal{S}_{2\times}^{L1C}$	16	2056	**4h**	38.58	35.27	0.81	1.16	0.9291	54.9243

Table 6.19: The synthetic performance of model $\mathcal{S}_{2\times}^{L1C}$ with scaled batch size and scaled learning rate. The learning rate of the experiment in each row is initialized with $0.0001\times$ No. GPUs

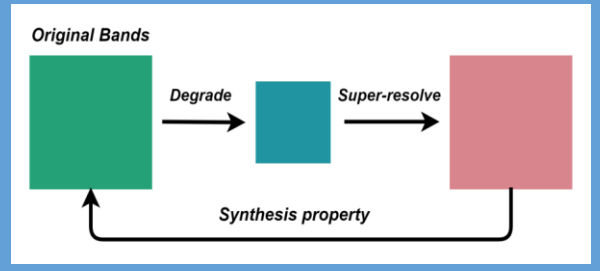


- Best performance when the batch size is scaled up to 512.
- Training time is reduced **from 4d to 4h** without severe performance loss

Distributed Super-resolution of L1C 60m bands (synthesis)

Method	No. GPU	Batch size	Training time	RMSE	SRE	SAM	ERGAS	SSIM	PSNR
Bicubic	-	-	-	161.85	19.79	1.78	7.30	0.36	37.6785
DSen2	1	128	96h	28.11	34.47	0.36	1.38	0.8953	52.4984
$\mathcal{S}_{6\times}^{L1C}$	1	32	24h	29.20	34.00	0.37	1.43	0.8917	51.9991
$\mathcal{S}_{6\times}^{L1C}$	2	64	24h	27.23	34.69	0.35	1.32	0.8959	52.7027
$\mathcal{S}_{6\times}^{L1C}$	4	128	24h	**26.80**	**34.98**	**0.34**	**1.29**	**0.8991**	**52.9451**
$\mathcal{S}_{6\times}^{L1C}$	8	256	12h	27.74	34.54	0.36	1.36	0.8959	52.5506
$\mathcal{S}_{6\times}^{L1C}$	16	512	**4h**	32.28	32.97	0.42	1.62	0.8828	50.9784

Table 6.20: The synthetic performance of model $\mathcal{S}_{6\times}^{L1C}$ with scaled batch size and scaled learning rate. The learning rate of the experiments in each row is initialized with $0.0001\times$ No. GPUs



- Best performance when the batch size is scaled up to 128.
- Train time can be reduced **from 4d to 4h** without severe performance loss

Data throughput when Super-resolving L1C 20 m bands (synthesis)

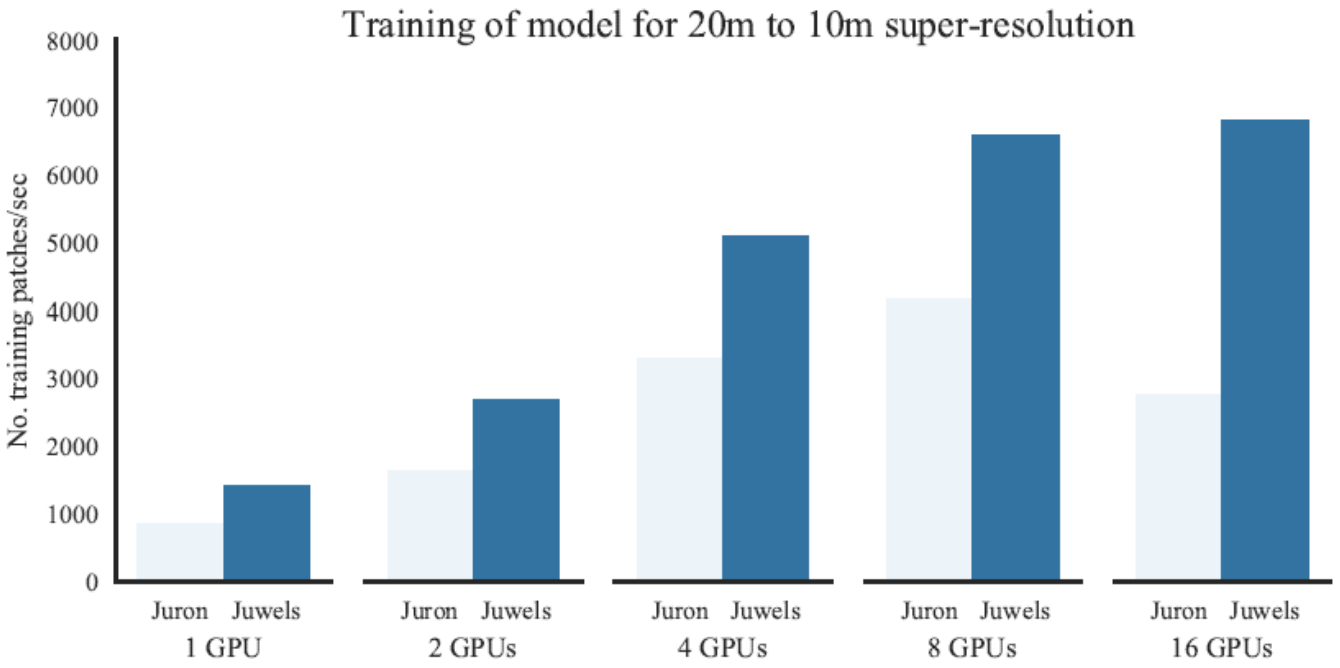


Figure 6.16: Data throughput when training $\mathcal{S}_{2\times}^{L1C}$ on Juron and Juwels.

- Cost to synchronize the gradients.
- GPU topology
- Infrastructure configuration

Data throughput when Super-resolving L1C 60m bands (synthesis)

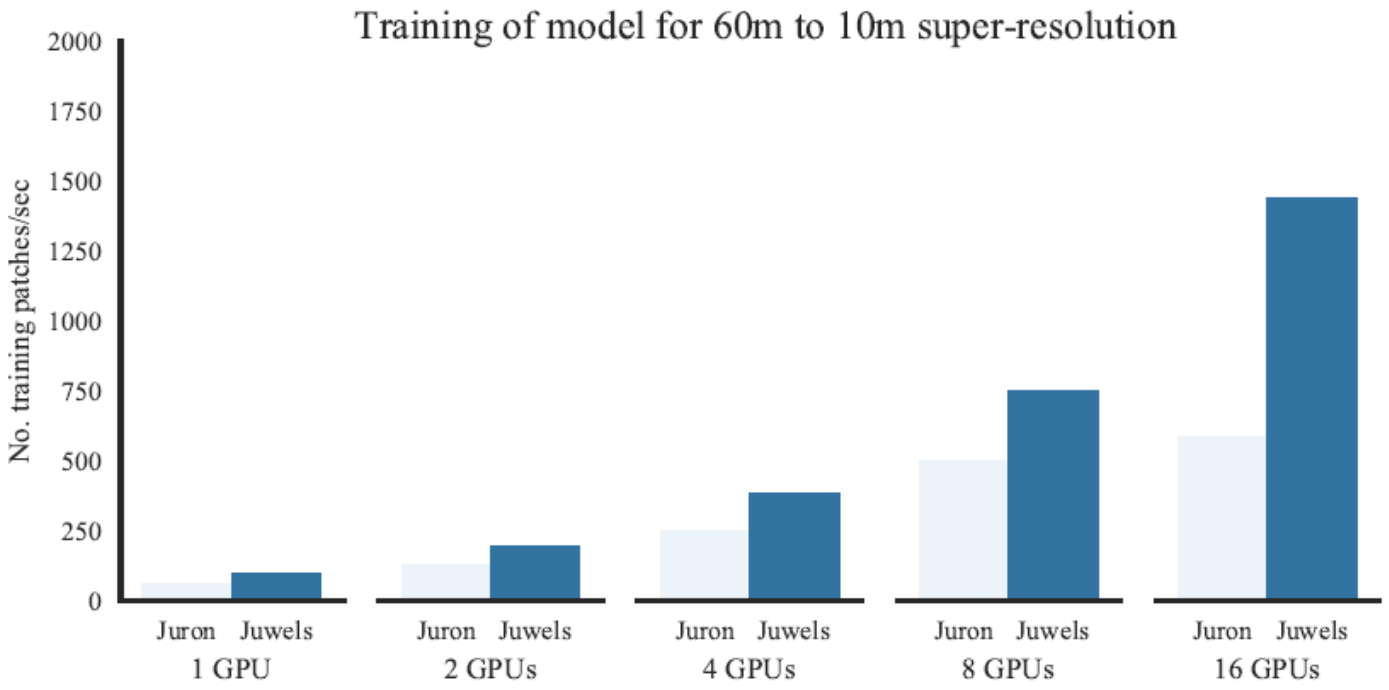
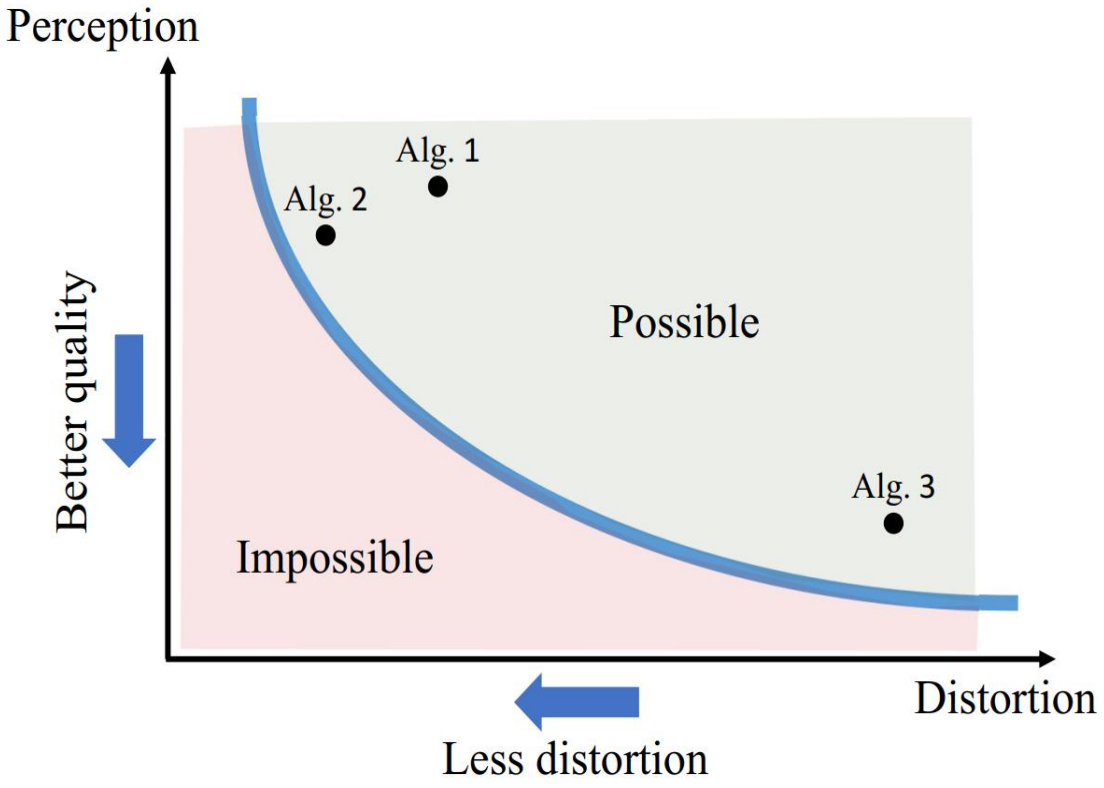
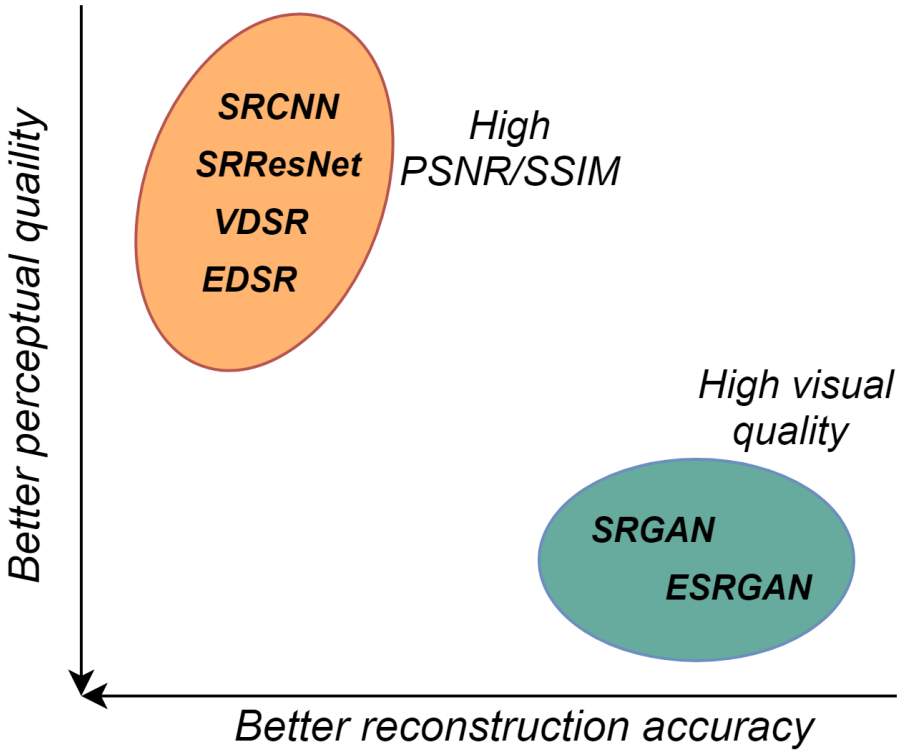


Figure 6.17: Data throughput when training $\mathcal{S}_{6\times}^{L1C}$ on Juron and Juwels.

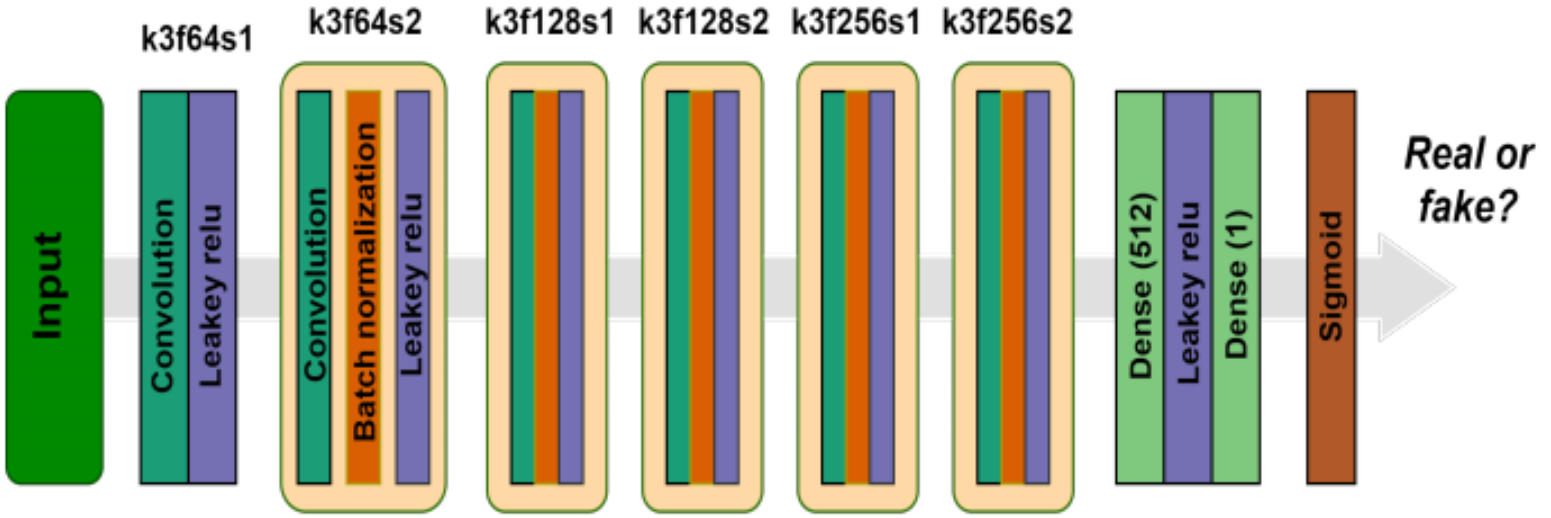
Why we use GANs ?





- Two objectives in disagreement with each other:
  - reconstruction accuracy, as measured by PSNR, SSIM, RMSE etc.
  - visual quality, as rated by human observe
- Two distinct research trends in image super-resolution

Discriminator



Method to stabilize a GAN:

1: pretrain the generator

2: balance the network complexity of the generator and discriminator

3: the generator and the discriminator use the different learning rate

4: balance the adversarial loss and the content loss

5: Try to use different GAN loss functions

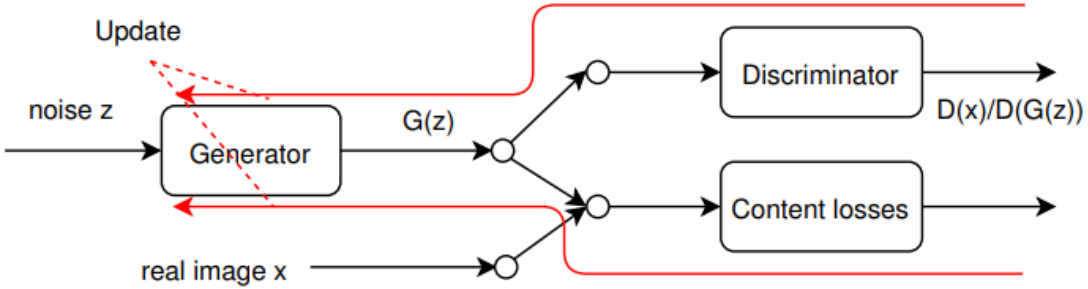


Figure 2.5: The process of updating the generator in a GAN. The architecture of GANs varies a lot. This figure illustrates a GAN architecture widely used in image super-resolution. As the red arrow shows, the generator G is penalized by both losses from Discriminator D and content losses. It means the generator G learns not only to fool the discriminator D, but also to minimize the distance between its output $G(z)$ and the real image x.

Adversarial losses

GAN type	Loss expression
GAN with Hinge loss[18]	$\hat{J}_{Hingle}(D) = \frac{1}{N}\sum_{n=0}^{N}[max(0, 1 - D(x_n))] + \frac{1}{N}\sum_{n=0}^{N}[max(0, 1 + D(G(z_n)))]$
	$\hat{L}_{Hingle}(G) = \hat{L}_{WGAN-GP}(G) = \hat{L}_{WGAN}(G) = -\frac{1}{N}\sum_{n=0}^{N}[D(G(z_n))]$
Relativitistic GAN [21]	$\hat{J}_{relativ}(D) = -\frac{1}{N}\sum_{n=0}^{N}[\log(D(x_n) - D(G(z_n)))]$
	$\hat{L}_{relativ}(G) = \sum_{n=0}^{N}[\log(D(G(z_n)) - D(x_n))]$

Adversarial losses

GAN type	Loss expression
Wasserstein GAN [22]	$\hat{J}_{WGAN}(D) = -\frac{1}{N}\sum_{n=0}^{N}[D(x_n)] + \frac{1}{\sum}_{n=0}^{N}[D(G(z_n))]$
	$\hat{L}_{WGAN}(G) = -\frac{1}{N}\sum_{n=0}^{N}[D(G(z_n))]$
Wasserstein GAN + Gradient Penalty [23]	$GP = \lambda\frac{1}{N}\sum_{1}^{N}(\|\nabla_{\tilde{x}_n} D(\tilde{x}_n)\| - 1)^2, \tilde{x}_n = tG(z_n) + (1-t)x_n$
	$\hat{J}_{WGAN-GP}(D) = \hat{J}_{WGAN}(D) + \lambda GP$
	$\hat{L}_{WGAN-GP}(G) = \hat{L}_{WGAN}(G) = -\frac{1}{N}\sum_{n=0}^{N}[D(G(z_n))]$

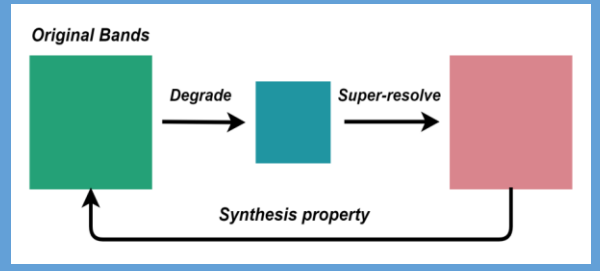
Adversarial losses

GAN type	Loss expression
Vanilla GAN [25]	$\hat{J}(D) = -\frac{1}{N}\sum_{n=0}^{N}[\log D(x_n)] - \frac{1}{N}\sum_{n=0}^{N}[\log(1 - D(G(z_n)))]$
	$\hat{L}(G) = \frac{1}{N}\sum_{n=0}^{N}[\log(1 - D(G(z_n)))]$

Distributed Super-resolution of L1C 20m bands (synthesis)

	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
pre-trained $\mathcal{S}_{2\times}^{L1C}$	**34.99**	**36.19**	**0.75**	**1.03**	**0.9336**	**55.7756**	**56.3358**
+ WGAN-GP	39.86	35.08	0.88	1.17	0.9269	54.6377	55.7148
+ vanilla GAN	61.28	31.18	1.39	1.93	0.8664	50.9394	50.7452
+ relativistic GAN	98.35	26.99	2.19	3.13	0.7860	46.7731	46.5310
+ hinge loss GAN	56.07	31.95	1.32	1.76	0.8824	51.7295	51.7466

Table 6.17: The effect of four adversarial losses on pre-trained $\mathcal{S}_{2\times}^{L1C}$ in sense of synthetic property. The definition of the adversarial loss in vanila GAN, relativistic GAN, WGAN-GP, and GAN with hinge loss is explained in see Section 5.2.2.

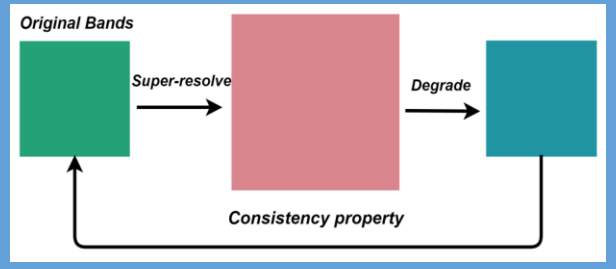


- Worse synthesis property with a discriminator.

Distributed Super-resolution of L1C 20m bands (consistency)

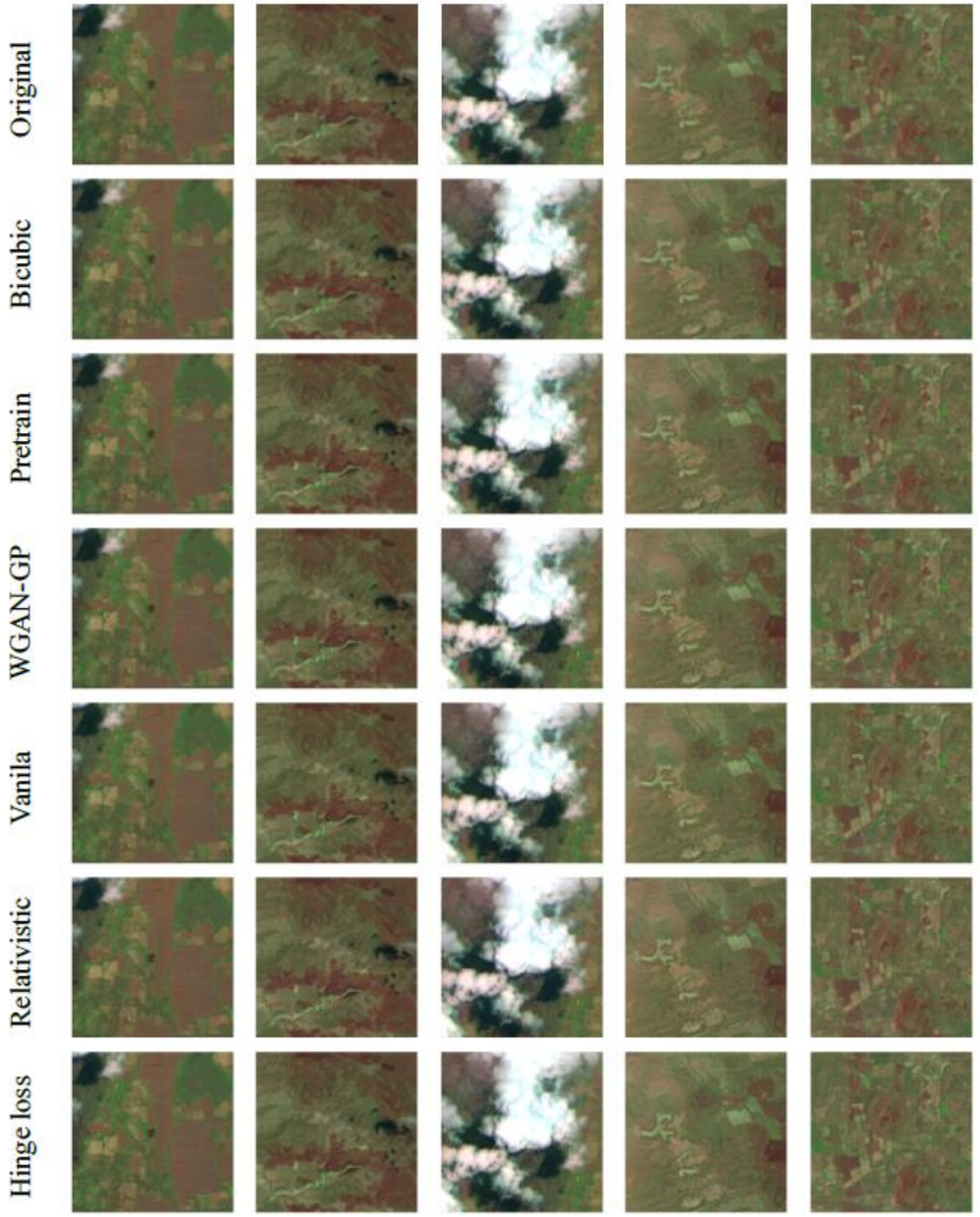
	RMSE	SRE	SAM	ERGAS	SSIM	PSNR	bPSNR
pre-trained $\mathcal{S}_{2\times}^{L1C}$	**4.28**	**55.11**	**0.07**	**0.1251**	**0.9901**	**74.9196**	**75.4208**
+ WGAN-GP	6.56	50.98	0.12	0.19	0.9893	70.8014	71.2999
+ vanilla GAN	45.94	33.67	1.09	1.39	0.9352	53.2586	53.1678
+ relativistic GAN	58.34	31.12	1.47	1.89	0.8979	51.3161	51.3626
+ hinge loss GAN	35.33	35.51	0.94	1.23	0.9528	55.6350	55.7038

Table 6.18: The effect of adversarial losses on pre-trained $\mathcal{S}_{2\times}^{L1C}$ in sense of consistency property.

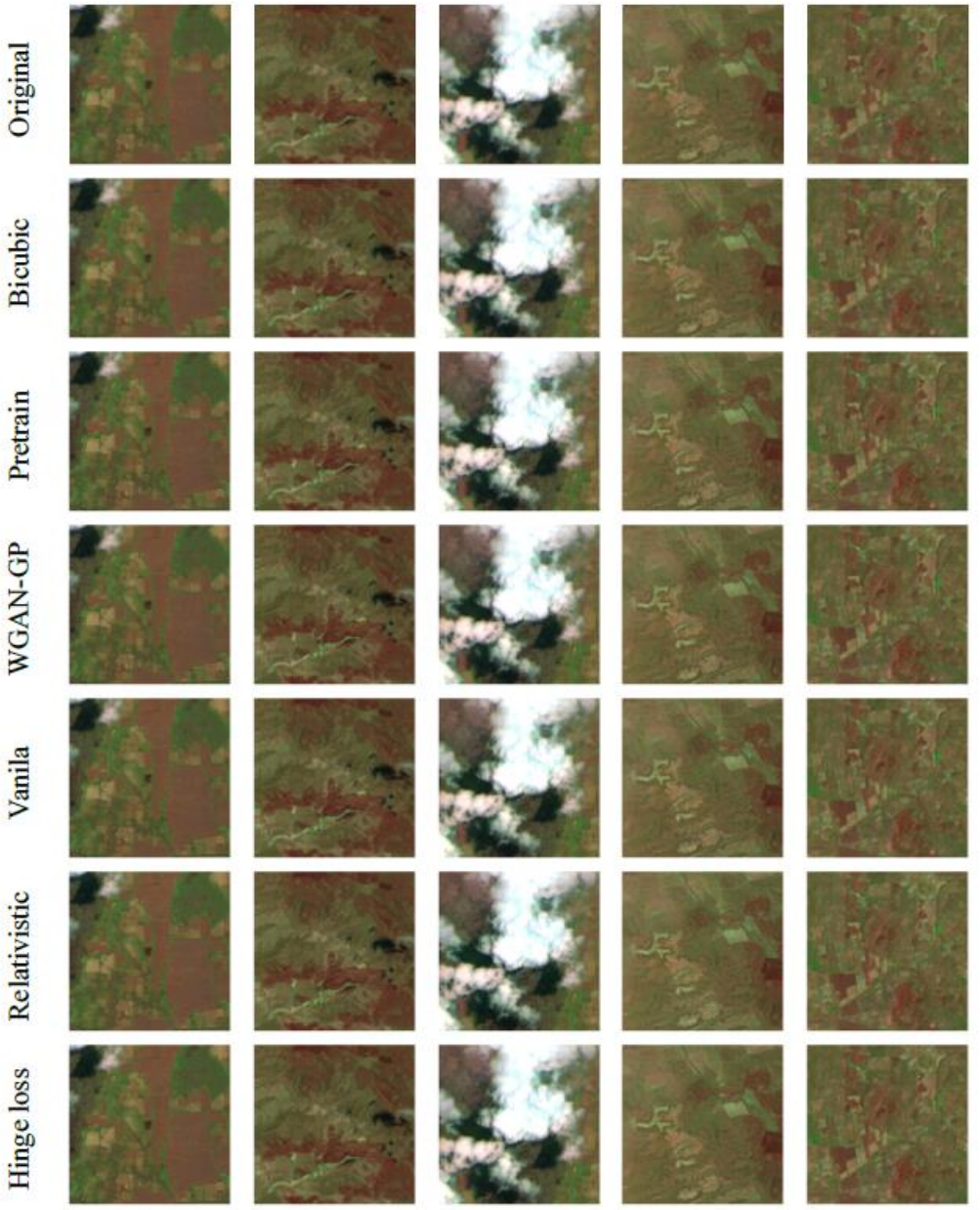


- Worse consistency with the discriminator.

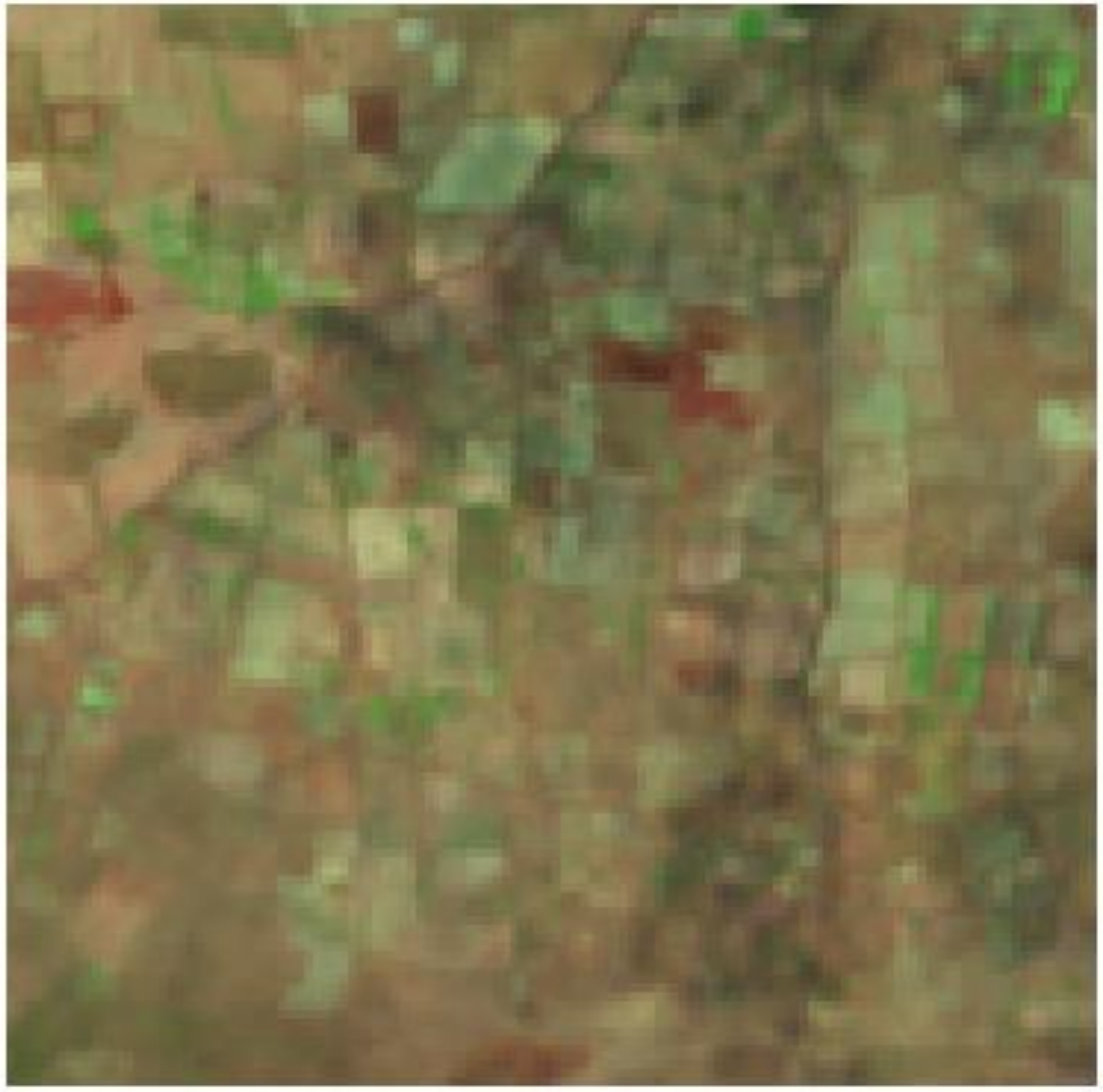
L1C 20m bands



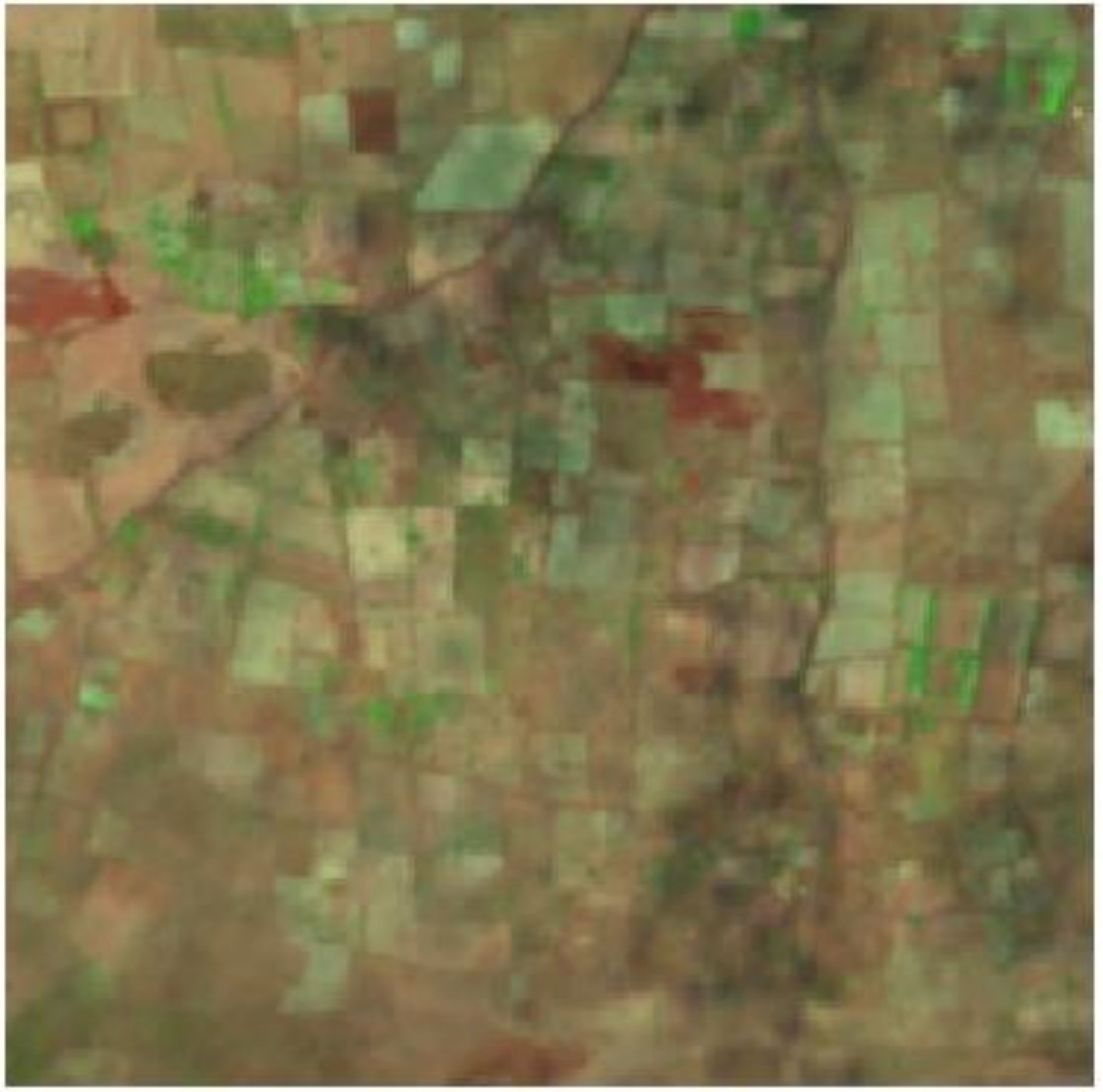
L2A 20m bands



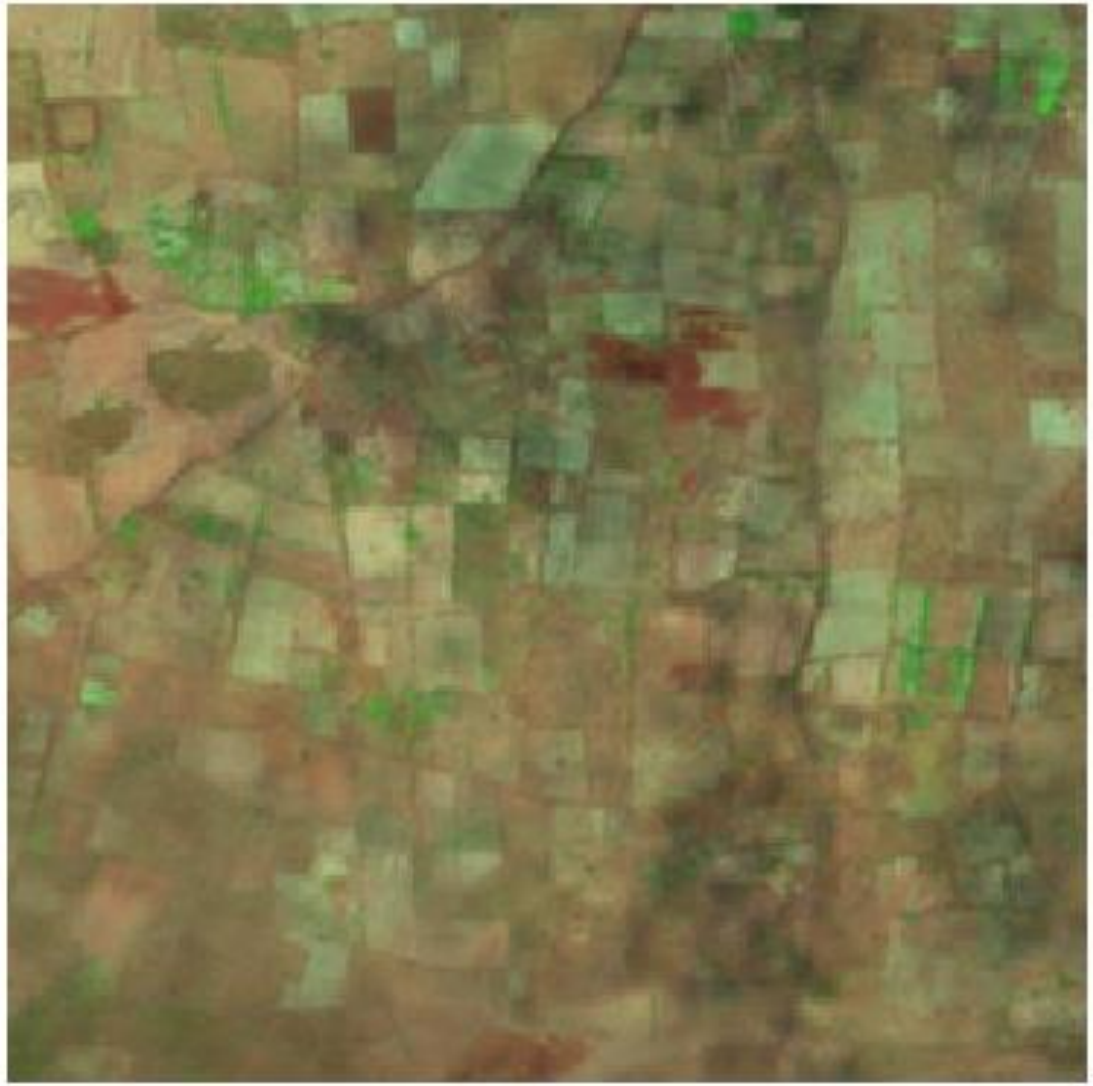
Original bands



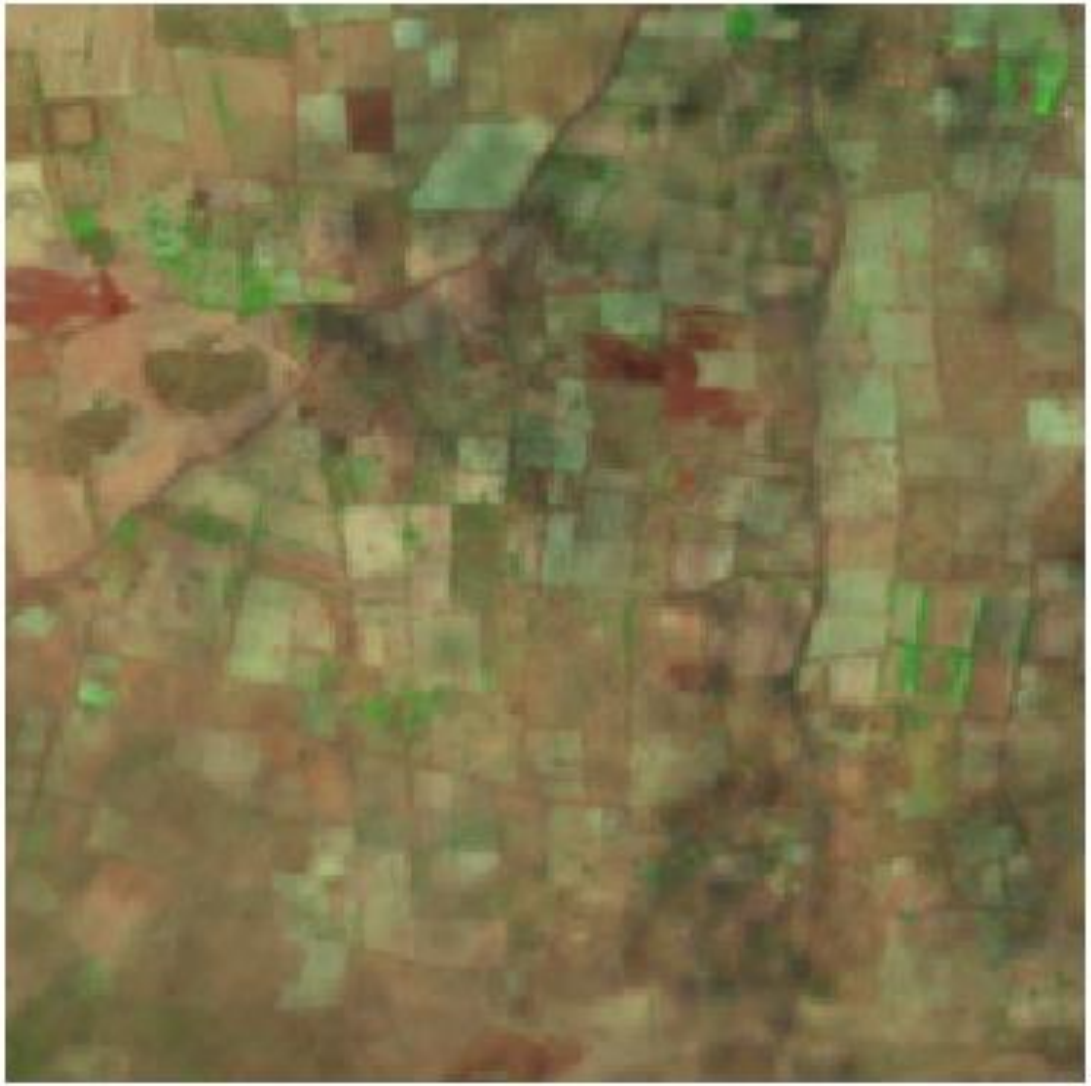
Generator only



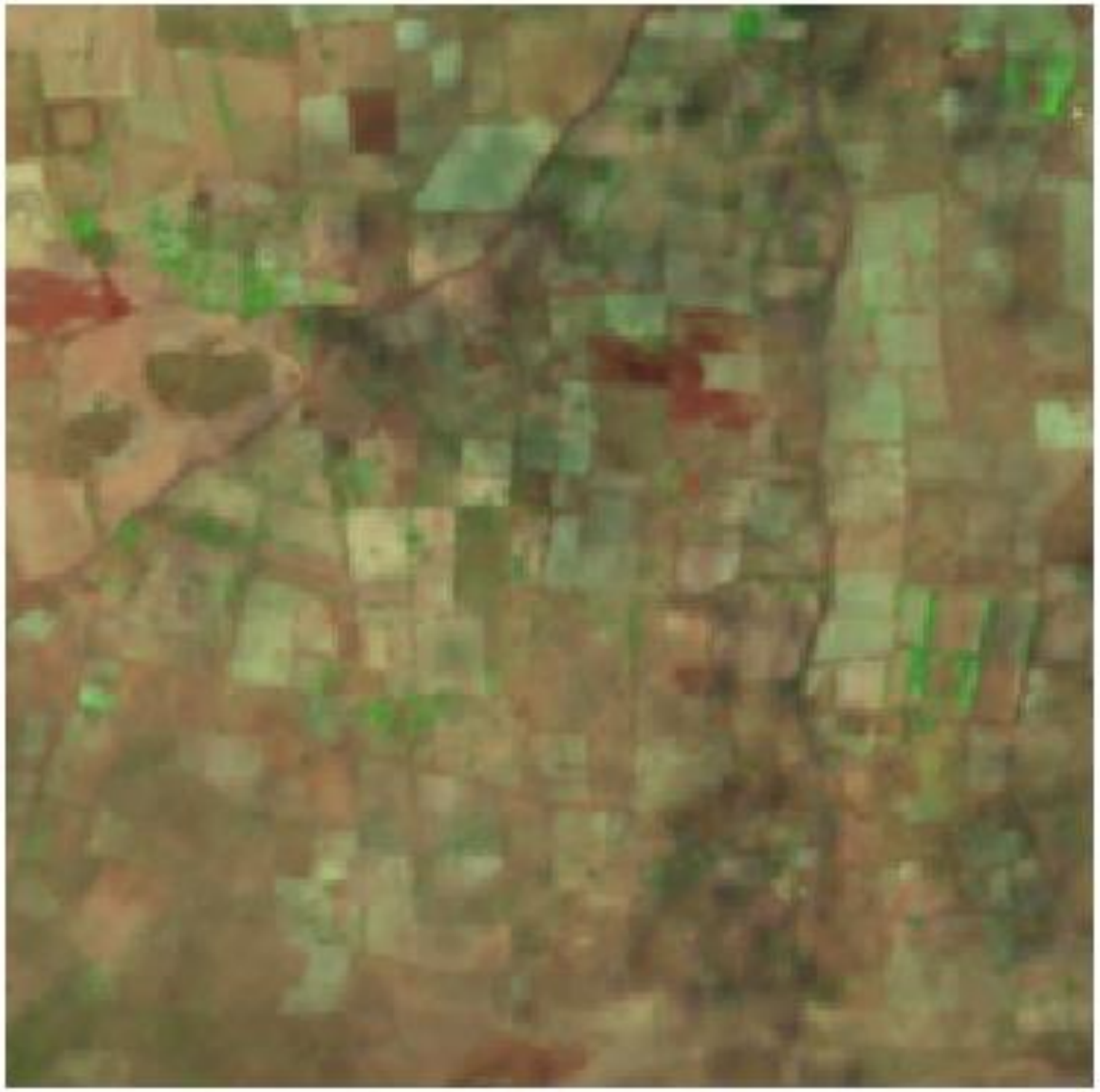
Relativistic GAN



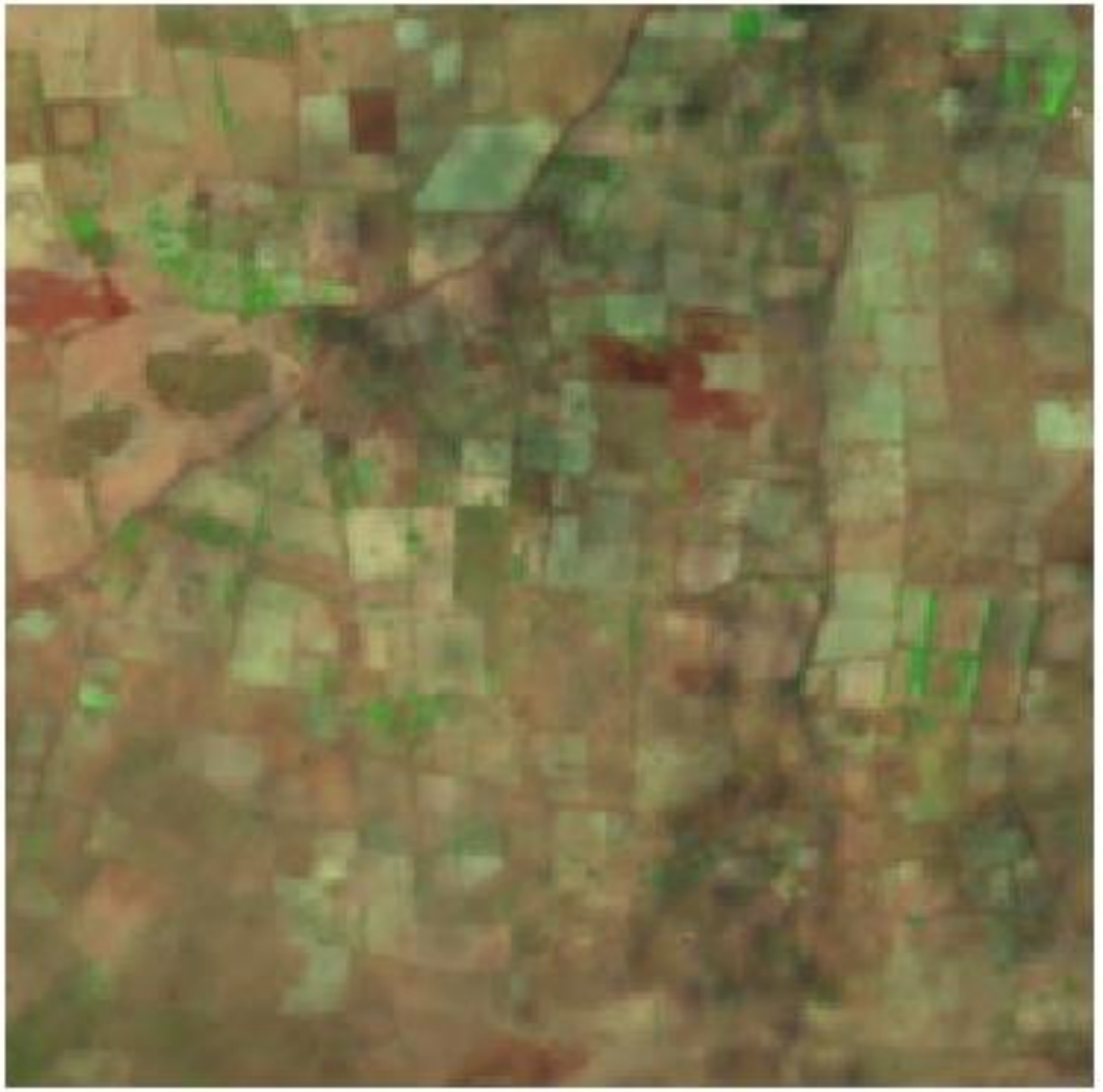
Vanila GAN



WGAN-GP



GAN with hinge loss



Conclusion

1. A model based on self-attention mechanism and residual learning was proposed and state-of-the-art performance is achieved on several metrics.
2. A comprehensive evaluation framework is proposed, we are the first to evaluate a learning-based RS super-resolution model with both synthetic and consistency properties on both level-1C and level-2A data
3. We scale up the training process to a distributed HPC system installed in JSC. Distributed deep learning is thus shown to significantly speed up the training and keep the model performance intact.
4. The effects of adversarial losses on super-resolution of large-scale multi-spectral RS observations are studied.

R.Zhang, G.Cavallaro, J.Jitsev "super-resolution of large volumes of sentinel-2 images with high performance distributed deep learning" in Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2020 (submitted)

Future directions

1: multi-scale super-resolution generator
2: improve the generator with the four super-resolution paradigms
3: a discriminator for remote sensing images super-resolution
4: stronger learning rate scaling, e.g., warm up or heuristic.
5: benefits of image super-resolution, cloud removal or land cover classification
6: balance between the content loss and adversarial loss, generator and discriminator
7: study the spectral distortion caused by band fusion
8: addressing the worse consistency of large scale super-resolution
…

Questions

References

- [1] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In European conference on computer vision, pages 184-199. Springer, 2014.
- [2] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1646-1654, 2016
- [3] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeplyrecursive convolutional network for image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1637-1645, 2016.
- [4] Christian Ledig, Lucas Theis, Ferenc Husz´ar, 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
- [5] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 136-144, 2017

References

- [6] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced superresolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV), 2018.
- [7] Junfeng Yang, Xueyang Fu, Yuwen Hu, Yue Huang, Xinghao Ding, and John Paisley. Pannet: A deep network architecture for pan-sharpening. In The IEEE International Conference on Computer Vision (ICCV), Oct 2017.
- [8] Xiangyu Liu, Yunhong Wang, and Qingjie Liu. Psgan: a generative adversarial network for remote sensing image pan-sharpening. In 2018 25th IEEE International Conference on Image Processing (ICIP), pages 873-877. IEEE, 2018.
- [18] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attentiongenerative adversarial networks. arXiv preprint arXiv:1805.08318, 2018.

References

- [9] Charis Lanaras, Jos´ e Bioucas-Dias, Silvano Galliani, Emmanuel Baltsavias, and Konrad Schindler. Super-resolution of sentinel-2 images: Learning a globally applicable deep neural network. ISPRS Journal of Photogrammetry and Remote Sensing, 146:305-319, 2018
- [10] Qunming Wang, Wenzhong Shi, Peter M Atkinson, and Eulogio Pardo-Ig´ uzquiza. A new geostatistical solution to remote sensing image downscaling. IEEE Transactions on Geoscience and Remote Sensing, 54(1):386-396, 2015.
- [11] Charis Lanaras, Jos´ e Bioucas-Dias, Emmanuel Baltsavias, and Konrad Schindler. Super-resolution of multispectral multiresolution images from a single sensor. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 20-28, 2017
- [17] Jens Leitloff and Felix M. Riese. Examples for CNN training and classification on Sentinel2 data. http://doi.org/10.5281/zenodo.3268451, 2018.

References

- [12] Nicolas Brodu. Super-resolving multiresolution images with band independent geometry of multispectral pixels. IEEE Transactions on Geoscience and Remote Sensing, 55(8):4610-4617, 2017.
- [13] Darren Pouliot, Rasim Latifovic, Jon Pasher, and Jason Duffe. Landsat super-resolution enhancement using convolution neural networks and sentinel-2 for training. Remote Sensing, 10(3):394, 2018.
- [14] Zhu, Xi, Yang Xu, and Zhihui Wei. "Super-Resolution of Sentinel-2 Images Based on Deep Channel-Attention Residual Network." *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019.
- [15] Goyal P, Dollár P, Girshick R, Noordhuis P, Wesolowski L, Kyrola A, Tulloch A, Jia Y, He K. Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677. 2017 Jun 8.
- [16] https://sentinel.esa.int/documents/247904/685211/Sentinel-2_L1C_Data_Quality_Report

References

- [19] Pitch Patarasuk and Xin Yuan. Bandwidth optimal allreduce algorithms for clusters of workstations. Journal of Parallel and Distributed Computing, 69(2):117–124, 2009.
- [20] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677, 2017.
- [21] Alexia JolicoeurMartineau. The relativistic discriminator: a key element missing from standard gan. arXiv preprint arXiv:1807.00734, 2018.
- [22] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In International conference on machine learning, pages 214–223, 2017.
- [23] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In Advances in neural information processing systems, pages 5767–5777, 2017.

References

- [24] Tao Dai, Jianrui Cai, Yongbing Zhang, ShuTao Xia, and Lei Zhang. Second order attention network for single image superresolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 11065–11074, 2019.
- [25] Ian Goodfellow, Jean PougetAbadie, Mehdi Mirza, Bing Xu, David WardeFarley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.
- [26] https://step.esa.int/main/third-party-plugins-2/sen2cor/
- [27] Alexander Sergeev and Mike Del Balso. Horovod: fast and easy distributed deep learning in tensorflow. arXiv preprint arXiv:1802.05799, 2018.

Related work overview

1: Nature image super-resolution

SRCNN[1]
VDSR [2]
DRCN [3]
SRResNet [4]
EDSR [5]
A special group based on GANs:
SRGAN [4]
ESRGAN [6]
…

Can not be applied directly!

2: Pan-sharpening

PSGAN [7]
PanNet [8]
…

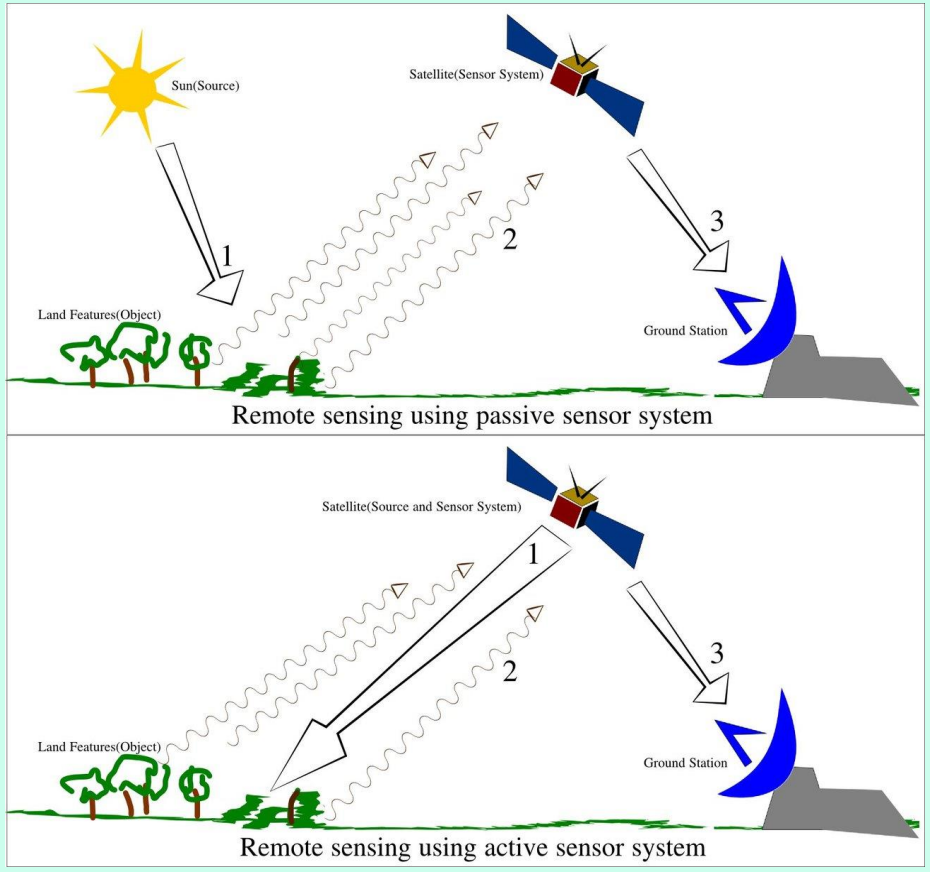
No panchromatic

3: Sentinel-2 super-resolution

DSen2 [9]
ATPRK [10]
SupReME [11]
Superres [12]
Sen2ResNet [13]
Sen2CA [14]
…

BASELINE

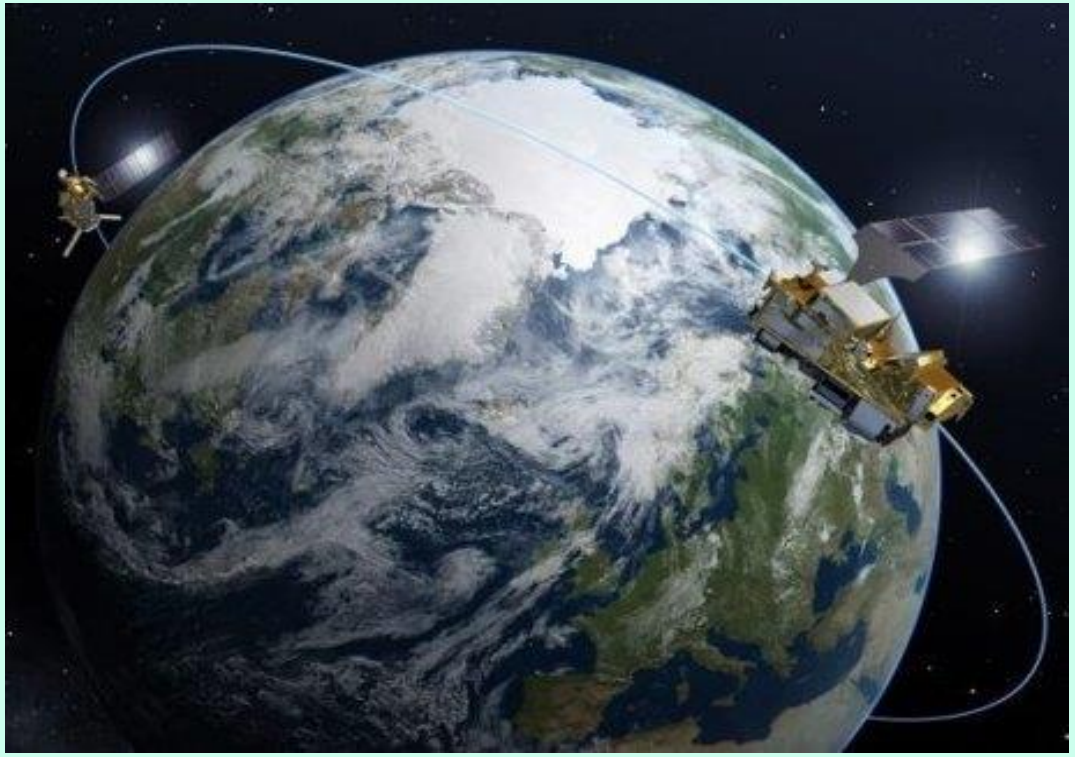
Remote sensing



https://en.wikipedia.org/wiki/Remote_sensing

- Acquire high-quality earth observation (EO)
- **Passive** remote sensing
- **Active** remote sensing

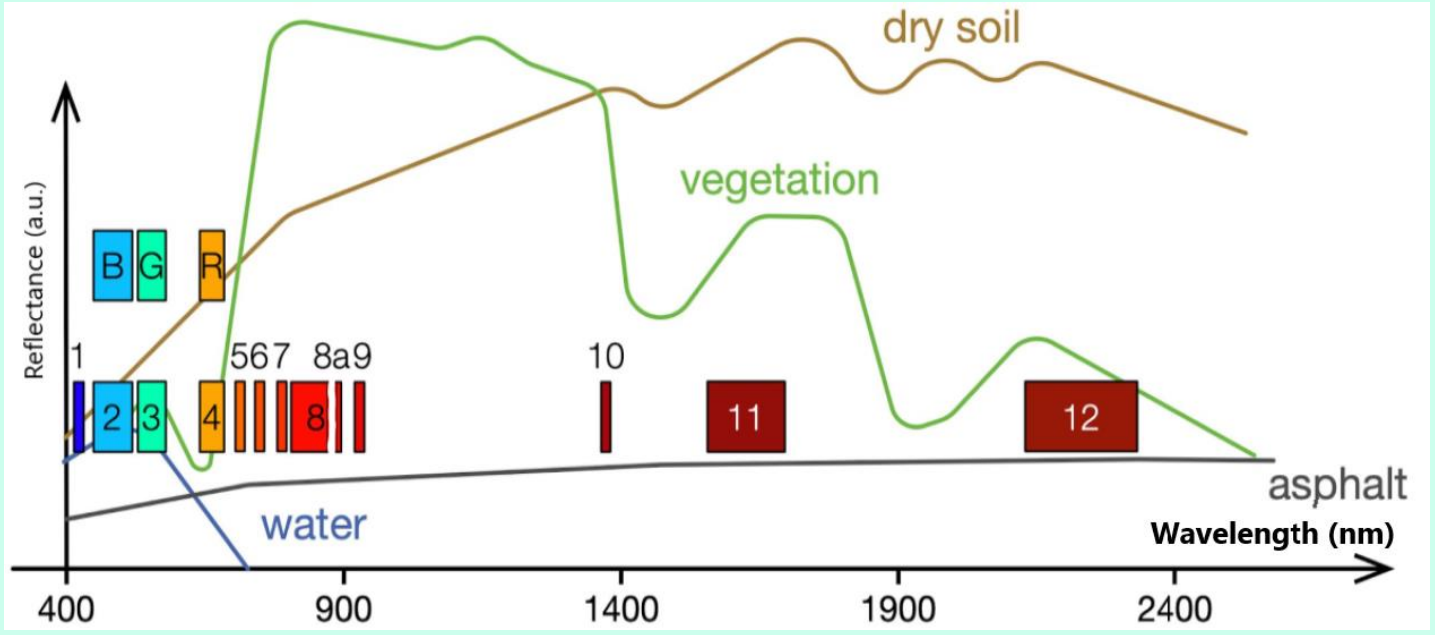
Sentinel-2A&2B



https://www.esa.int/Enabling_Support/Operations/Sentinel-2_operations

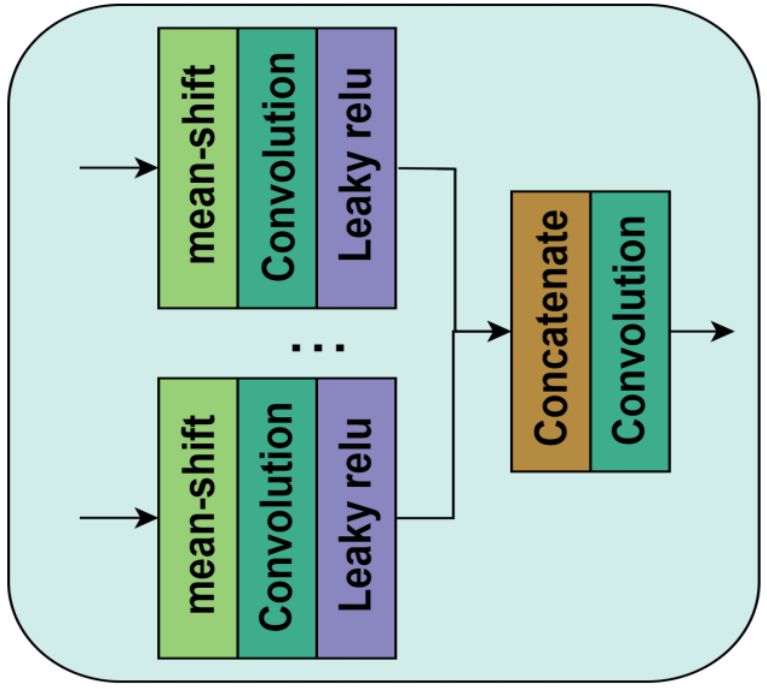
- European Space Agency's Copernicus program.
- Two satellites placed in the same sun synchronous orbit and phased at 180 degree.

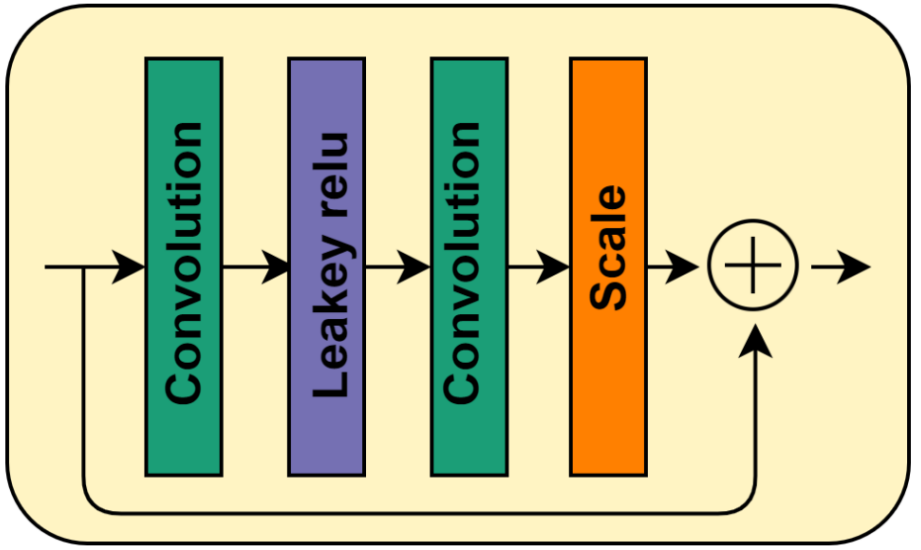
Sentinel-2 images



Leitloff et.al [17]

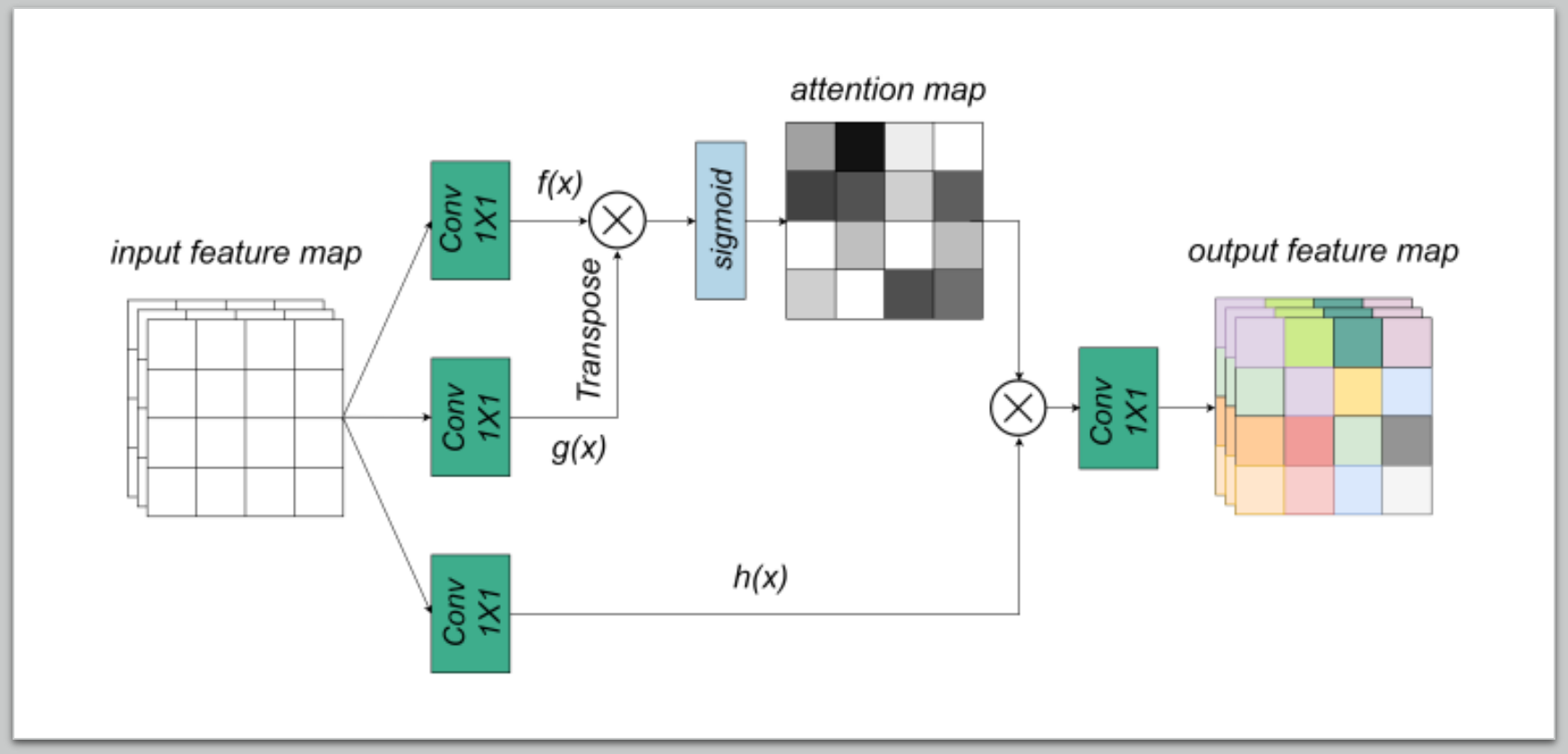
- Different bands have different usecases
- Combination can be used as a spectral signature.
- B10 is exclude due to poor radiometric quality and across-track artifacts.
- Multiple processing levels (0, 1A, 1B, 1C, 2A)





Residual blocks and band fusion module

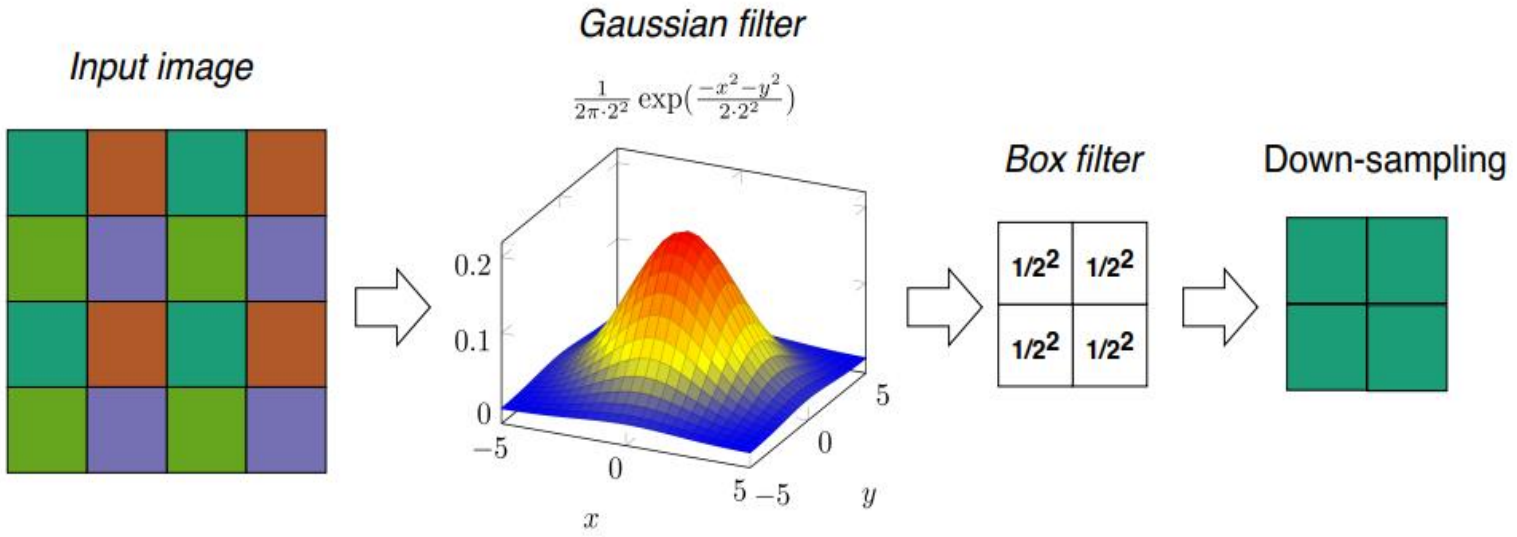
1. Mean-shift is used to suppress large brightness changes across all patches.
2. Scaling with a constant is used instead of batch normalization to speed up training.



Self-attention module [18]

1: Simple matrix multiplication;
2: To capture the long-range dependencies over the entire input feature maps;

Degradation filters



- MTF of Sentinel-2 [16]
- Gaussian blurry filter is used to emulate the MTF of the Sentinel-2 image sensor
- Box filter is used to down-sample the images.
- DIV2K: Bicubic interpolation without blurry filters.

Variant: Brightness-equalized PSNR

$$\mathcal{PSNR} = 10 \cdot \log\left(\frac{MAX^2}{\frac{1}{|N|}\sum_{n \in N}\left(HR_n - \hat{HR}_n\right)^2}\right)$$

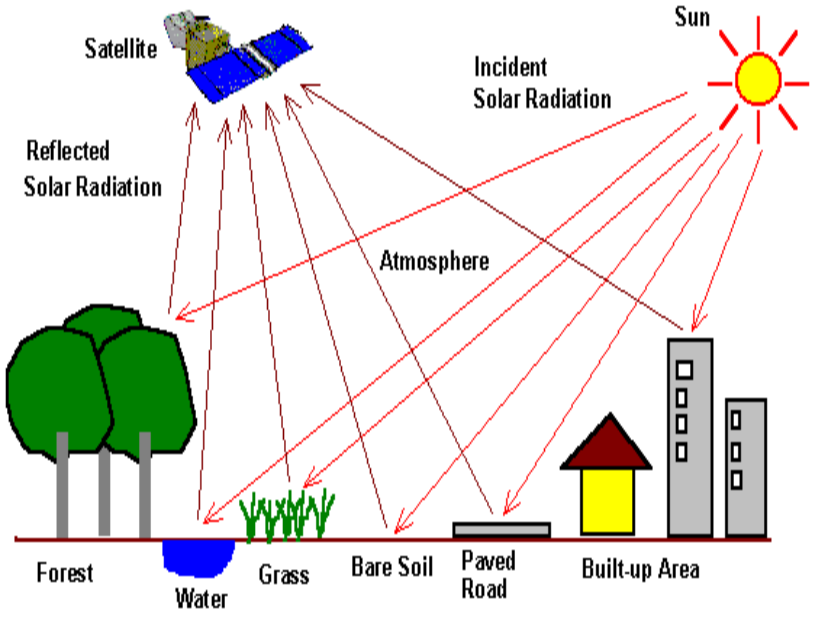
$$b = \frac{1}{N}\left(\sum_{x,y \in HR} HR(x,y) - \hat{HR}(x,y)\right)$$

$$cMSE(HR, \hat{HR}) = \frac{1}{N}\sum_{x,y \in HR}\left(HR(x,y) - \left(\hat{HR}(x,y) + b\right)\right)^2$$

$$\mathcal{BPSNR}(HR, \hat{HR}) = -10\log_{10}\left(\frac{MAX^2}{cMSE(HR, \hat{HR})}\right)$$

L1C-L2A conversion by Sen2cor

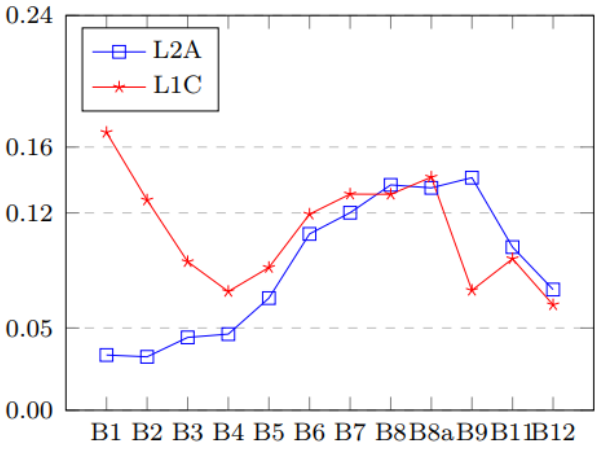
L1C: *Top of atmosphere*
L2A: *Bottom of atmosphere*



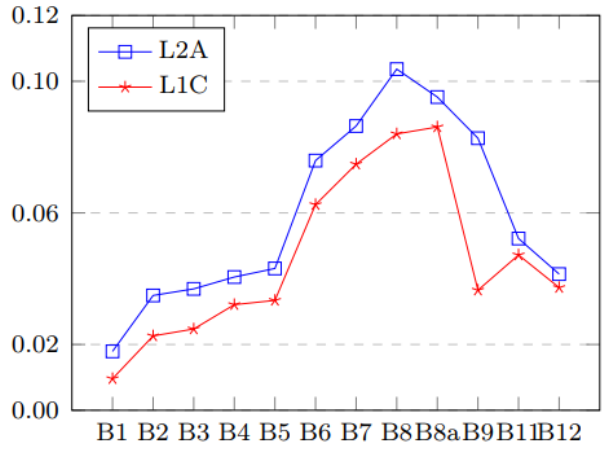
https://www.cimel.fr/solutions/

(a) Level-1C

(b) Level-2A



(c) Mean of reflectance in each band.

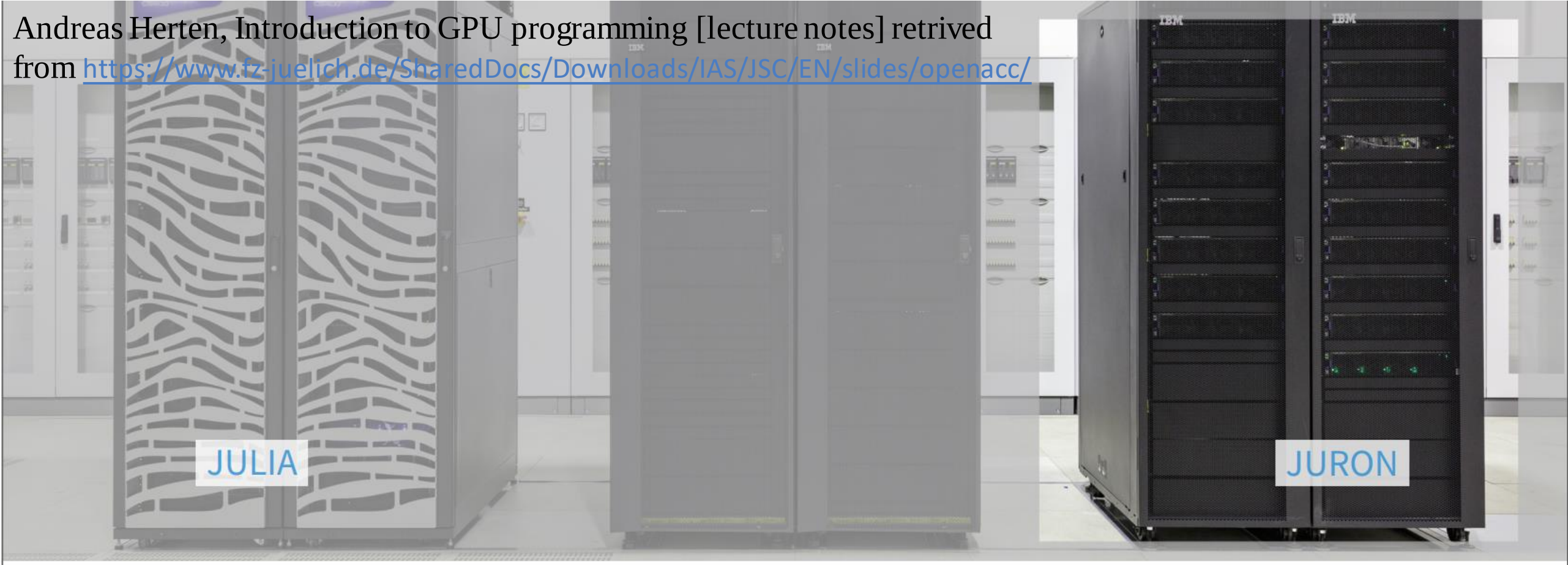


(d) Standard deviation of reflectance in each band.

Distributed learning

- Short introduction of the supercomputers in JSC
- Synchronous data parallism
- Ring-reduction mechanism
- (modified) learning rate linear scaling rule
- Result analysis

Andreas Herten, Introduction to GPU programming [lecture notes] retrived from https://www.fz-juelich.de/SharedDocs/Downloads/IAS/JSC/EN/slides/openacc/



JURON – A Human Brain Project *Prototype*

- 18 nodes with IBM POWER8*NVL* CPUs (2 × 10 cores)
- Per Node: 4 NVIDIA Tesla P100 cards (16 GB HBM2 memory), connected via NVLink
- GPU: 0.38 PFLOP/s peak performance

Andreas Herten, Introduction to GPU programming [lecture notes] retrived from https://www.fz-juelich.de/SharedDocs/Downloads/IAS/JSC/EN/slides/openacc/

JURECA – Jülich's Multi-Purpose Supercomputer

- 1872 nodes with Intel Xeon E5 CPUs (2 × 12 cores)
- 75 nodes with 2 NVIDIA Tesla K80 cards (look like 4 GPUs)
- JURECA Booster: 1640 nodes with Intel Xeon Phi *Knights Landing*
- 1.8 (CPU) + 0.44 (GPU) + 5 (KNL) PFLOP/s peak performance (Top500: #44)
- Mellanox EDR InfiniBand

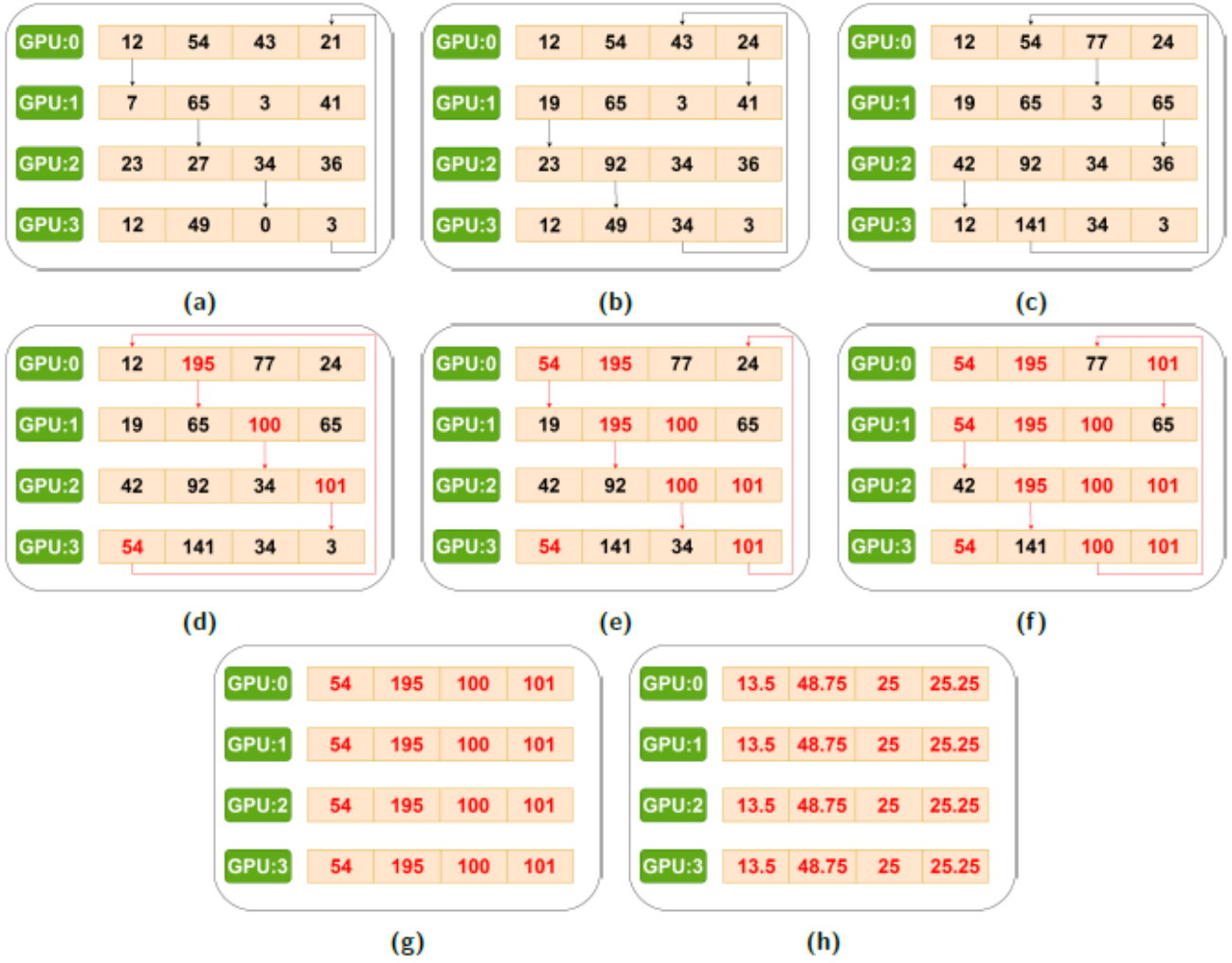


Figure 5.6: Illustration of gradient ring-reduction step by step in a cluster with 4 GPUs. Buffers in each GPU are used to synchronize gradients. In this example, each GPU is equipped with 4 buffers (depicted as yellow grids) and each buffer is of size for 1 gradient. The objective of ring-reduction is to average and synchronize gradients over the 4 GPUs. The address (grid index) of each gradient decides the model parameter it refers to, *e.g.*, 12, 7, 23, 12 in the first grid in Figure (a) are gradients for a same model parameter. We get the sum of all gradients in Figure (g), and each GPU calculates the average by dividing with the number of GPUs in Figure (h).

Ring-reduction mechanism [19]

- *n* nodes ==> *n* chunks
- First *(n-1)* iteration: each machine sends one chunk of data to its neighbor and the received chunk is added to corresponding chunk.
- Second *(n-1)* iteration: each machine sends one chunk of data to its neighbor and the received chunk replace the corresponding chunk.
- Reduce the volume of transmitted data (4x6 VS 16x2)
- Bandwidth optimal
- Open source library Horovod

Super-resolution of L1C 20m bands (synthesis)

	B5	B6	B7	B8a	B11	B12
	RMSE					
Bicubic	101.23	117.29	129.52	137.73	127.65	118.73
ATPRK	89.4	119.1	136.5	147.4	113.3	91.7
SupReME	48.1	70.2	78.6	82.9	76.5	61.7
Superres	50.2	66.6	76.8	82.0	66.9	54.5
DSen2	27.74	32.68	36.07	38.02	36.22	34.55
$\mathcal{S}_{2\times}^{L1C}$	**27.48**	**32.27**	**35.58**	**37.46**	**35.56**	**33.68**
	SRE					
Bicubic	25.46	25.69	25.69	25.73	25.94	25.70
ATPRK	26.6	26.9	26.7	26.6	24.7	22.7
SupReME	31.2	31.0	31.0	31.2	27.9	26.1
Superres	31.3	31.7	31.4	31.4	29.1	27.2
DSen2	36.15	36.33	36.37	36.49	36.45	35.97
$\mathcal{S}_{2\times}^{L1C}$	**36.26**	**36.44**	**36.49**	**36.62**	**36.66**	**36.22**

Table 6.6: Per-band performance of super-resolving each $20m$ band in B by pre-trained model $\mathcal{S}_{2\times}^{L1C}$ in sense of synthesis property. Our method has achieved the best result consistently over all metrics and over all $20m$ bands.

- RMSE is highly correlated to pixel intensity (B6, B7, B8a).
- The performance of SRE is more balanced than RMSE

Super-resolution of L1C 20m bands (consistency)

	B5	B6	B7	B8a	B11	B12
	RMSE					
Bicubic	28.50	33.23	36.78	38.95	35.50	32.73
DSen2	6.34	6.20	6.53	6.63	6.10	5.67
$\mathcal{S}_{2\times}^{L1C}$	**4.31**	**4.58**	**4.77**	**4.86**	**4.40**	**4.09**
	SRE					
Bicubic	36.47	36.65	36.63	36.70	37.14	37.01
DSen2	50.06	51.49	51.81	52.21	52.64	52.42
$\mathcal{S}_{2\times}^{L1C}$	**53.21**	**54.20**	**54.63**	**55.00**	**55.61**	**55.43**

Table 6.9: Per-band performance of super-resolving each $20m$ band in B by pre-trained model $\mathcal{S}_{2\times}^{L1C}$ in sense of consistency property.

- Consistency property is also correlated to pixel intensity (B6, B7, B8a)

Super-resolution of L2A 20m bands (synthesis)

	B5	B6	B7	B8a	B11	B12
	RMSE					
Bicubic	133.87	149.90	160.27	165.71	152.44	142.35
DSen2	41.01	45.58	48.57	50.23	47.91	46.14
DSen2-L2A	37.07	41.99	44.97	46.24	43.51	41.63
$\mathcal{S}_{2\times}^{L2A}$	**36.14**	**40.86**	**43.64**	**44.83**	**42.00**	**40.07**
	SRE					
Bicubic	23.74	24.19	24.34	24.52	24.94	24.84
DSen2	33.91	34.39	34.57	34.73	34.73	34.30
DSen2-L2A	34.59	34.99	35.16	35.39	35.57	35.22
$\mathcal{S}_{2\times}^{L2A}$	**34.83**	**35.26**	**35.45**	**35.69**	**35.91**	**35.59**

Table 6.14: Per-band performance of super-resolving each $20m$ band in B by pre-trained model $\mathcal{S}_{2\times}^{L2A}$ in sense of synthetic property.

- Similar to L1C super-resolution, RMSE is highly correlated to pixel intensity (B6, B7, B8a).
- Better performance on RMSE and SRE over all spectral bands.

Super-resolution of L2A 20m bands (consistency)

	B5	B6	B7	B8a	B11	B12
	RMSE					
Bicubic	37.89	42.61	45.66	47.03	42.55	39.36
DSen2-L2A	8.95	9.64	10.14	10.27	9.15	8.32
$\mathcal{S}_{2\times}^{L2A}$	**6.33**	**6.03**	**6.27**	**6.44**	**5.75**	**5.30**
	SRE					
Bicubic	34.75	35.16	35.29	35.49	36.14	36.15
DSen2-L2A	48.08	48.66	48.90	49.21	50.06	50.32
$\mathcal{S}_{2\times}^{L2A}$	**51.49**	**52.96**	**53.16**	**53.30**	**54.15**	**54.20**

Table 6.16: Per-band performance of super-resolving each $20m$ band in B by pre-trained model $\mathcal{S}_{2\times}^{L2A}$ in sense of consistency property.

- Better performance on RMSE and SRE over all L2A spectral bands

Super-resolution of L1C 60m bands (synthesis)

	B1	B9	B1	B9
	RMSE		SRE	
Bicubic	169.54	158.12	22.43	19.79
ATPRK	162.9	127.4	22.8	18.0
SupReME	114.9	56.4	25.2	24.5
Superres	107.5	92.9	24.8	20.8
DSen2	29.28	27.51	37.25	34.44
$\mathcal{S}_{6\times}^{L1C}$	**27.60**	**26.18**	**37.77**	**34.95**

Table 6.8: Per-band performance of super-resolving each $60m$ band by pre-trained model $\mathcal{S}_{6\times}^{L1C}$ in sense of synthesis property, including $B1$ and $B9$. Our method achieved the best result consistently over all evaluation metrics and over all $60m$ bands.

- Better RMSE and SRE over both 60m spectral bands

Perception-oriented example: SRGAN[4]

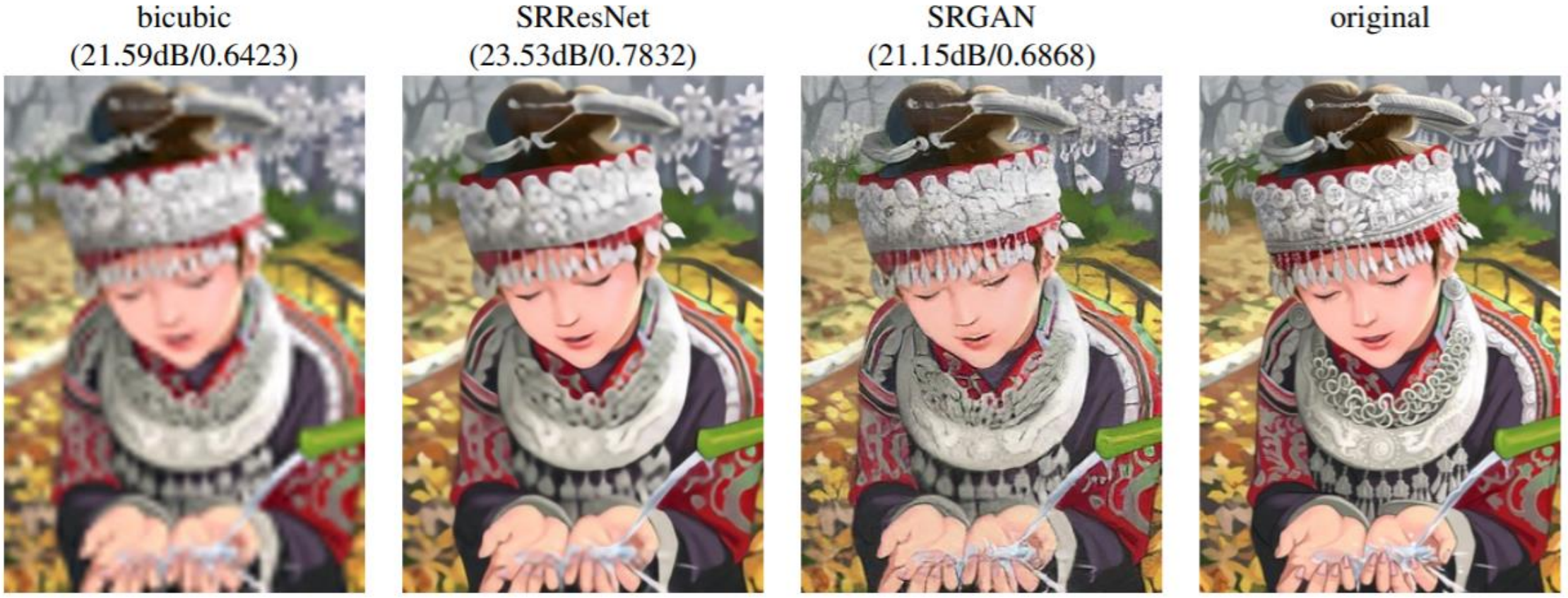


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

Alternating update in a GAN

Algorithm 5.2.1 Minibatch stochastic gradient descent training of GANs. The number of steps to update the discriminator, k, is a hyper-parameter. For WGAN and WGAN-gp, $k = 1$ is used. For other types of GAN, $k = 1$ is used.

for number of training epochs **do**
  for k steps **do**
    - Sample a minibatch of HR ground truth $\{x_1, ..., x_n\}$.
    - Run generator to create a minibatch of fake output $\{G(z_1), ..., G(z_n)\}$.
    - Update the discriminator by descending its stochastic gradient: $\mathcal{L}_D$.
  end for
  - Sample a minibatch of LR input $\{z_1, ..., z_n\}$.
  - Update the generator by descending its stochastic gradient: $\mathcal{L}_G$
end for

Method to stabilize a GAN:

1: pretraind the generator with the entire dataset

2: (k is introduced to balance the generate and discriminator optimization)

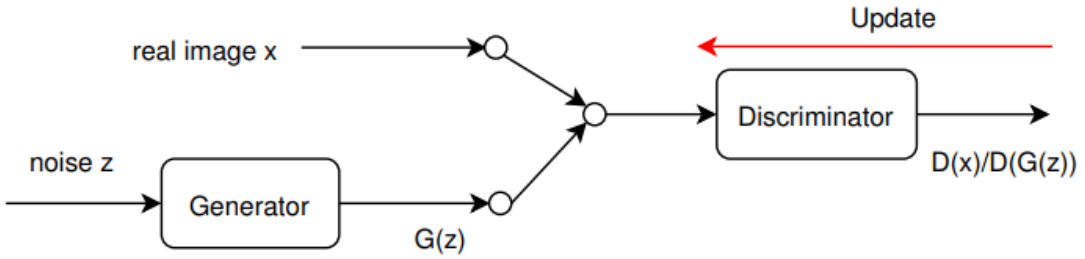


Figure 2.6: Illustration of updating the discriminator in a GAN. The discriminator D learns to distinguish the generated data $G(z)$ and the real data x. So in each iteration, the discriminator is updated to enlarge the difference between $D(x)$ and $D(G(z))$.

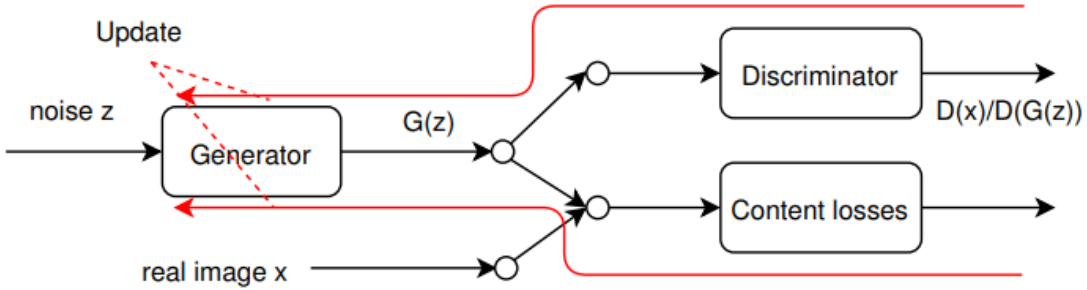


Figure 2.5: The process of updating the generator in a GAN. The architecture of GANs varies a lot. This figure illustrates a GAN architecture widely used in image super-resolution. As the red arrow shows, the generator G is penalized by both losses from Discriminator D and content losses. It means the generator G learns not only to fool the discriminator D, but also to minimize the distance between its output $G(z)$ and the real image x.

GAN-base SR architecture

Discriminator: 1) the loss to update itself;
2) the adversarial loss;

Content losses: Distance between outputs of the generator and the ground truth