

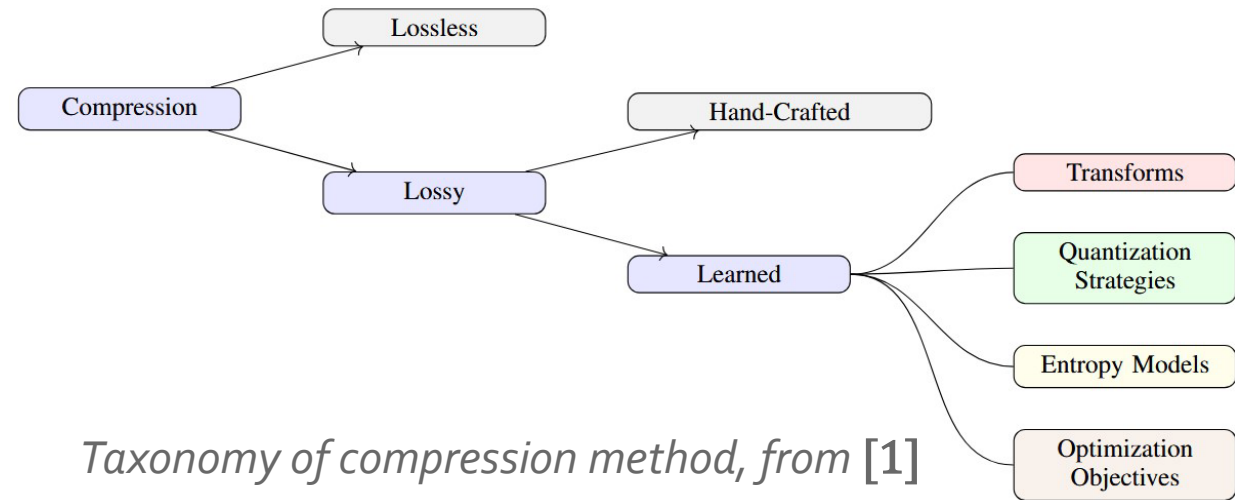
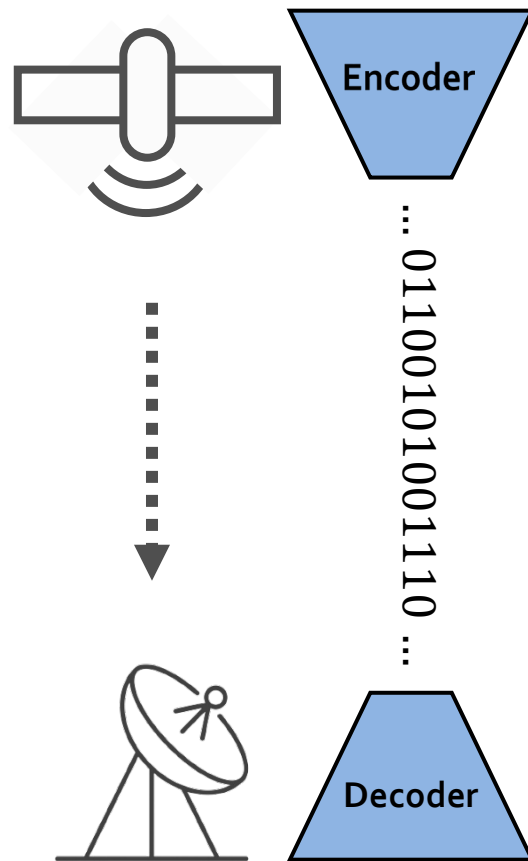
Onboard Machine Learning-based Compression of Synthetic Aperture Radar (SAR) Images Using FPGA/MPSoC Hardware

23/06/2025, LPS25, Vienna

Cédric Léonard, Francescopaolo Sica, Martin Schulz

Wissen für Morgen

Relaxing memory and bandwidth constraints onboard



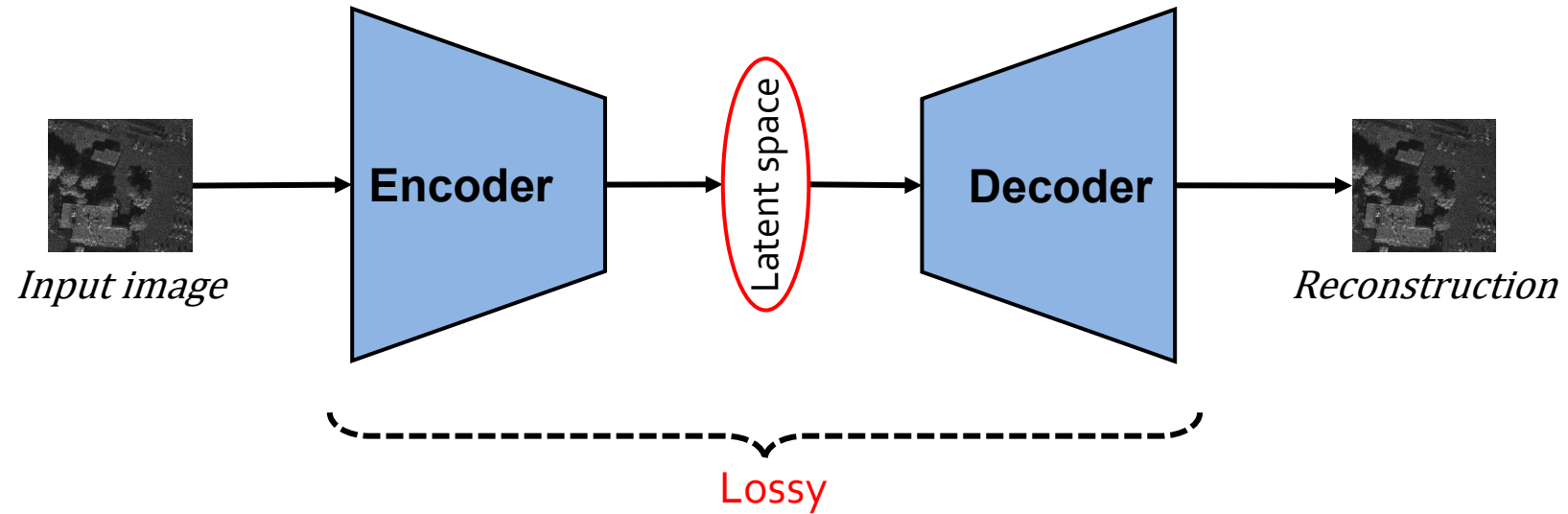
Taxonomy of compression method, from [1]



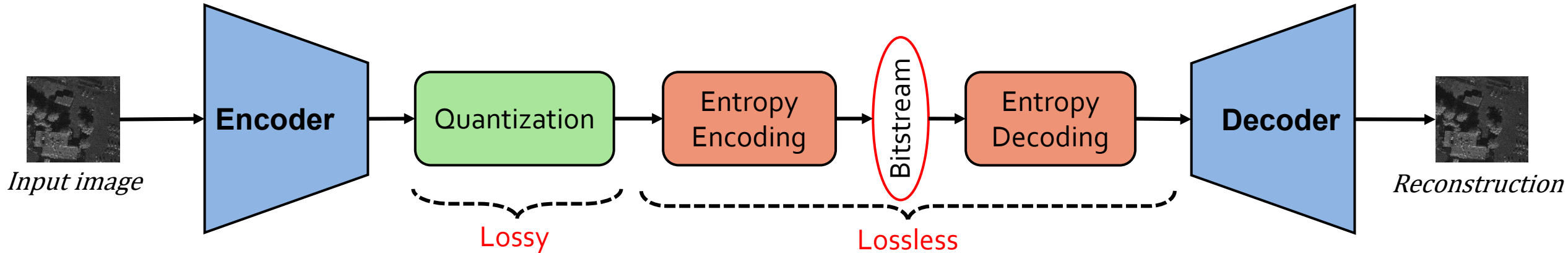
Overview of SAR focusing flow, adapted from [2]

1. Gomes, C., Wittmann, I., ... Albrecht, C. M. (2025). Lossy Neural Compression for Geospatial Analytics: A Review. *arXiv*.
2. Mandapati, S., Balss, U., & Breit, H. (2024). Real Time Floating Point SAR Focusing on FPGA. *Proceedings of the European Conference on Synthetic Aperture Radar, EUSAR*, 60–65.

Neural compression: autoencoder

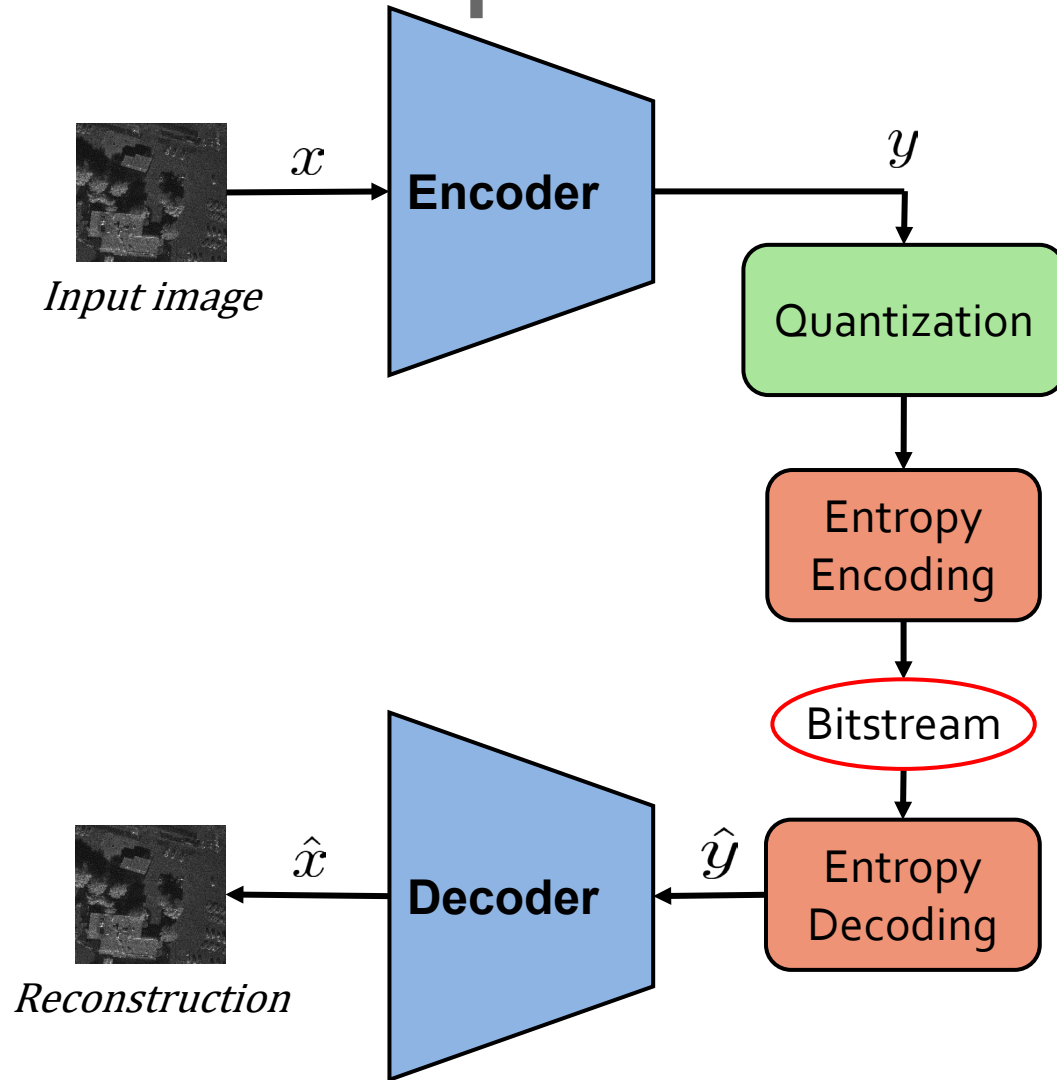


Neural compression: entropy modeling



Ballé, J., Laparra, V., & Simoncelli, E. P. (2017). End-to-end Optimized Image Compression (No. arXiv:1611.01704). *arXiv*.

Neural compression: end-to-end optimization



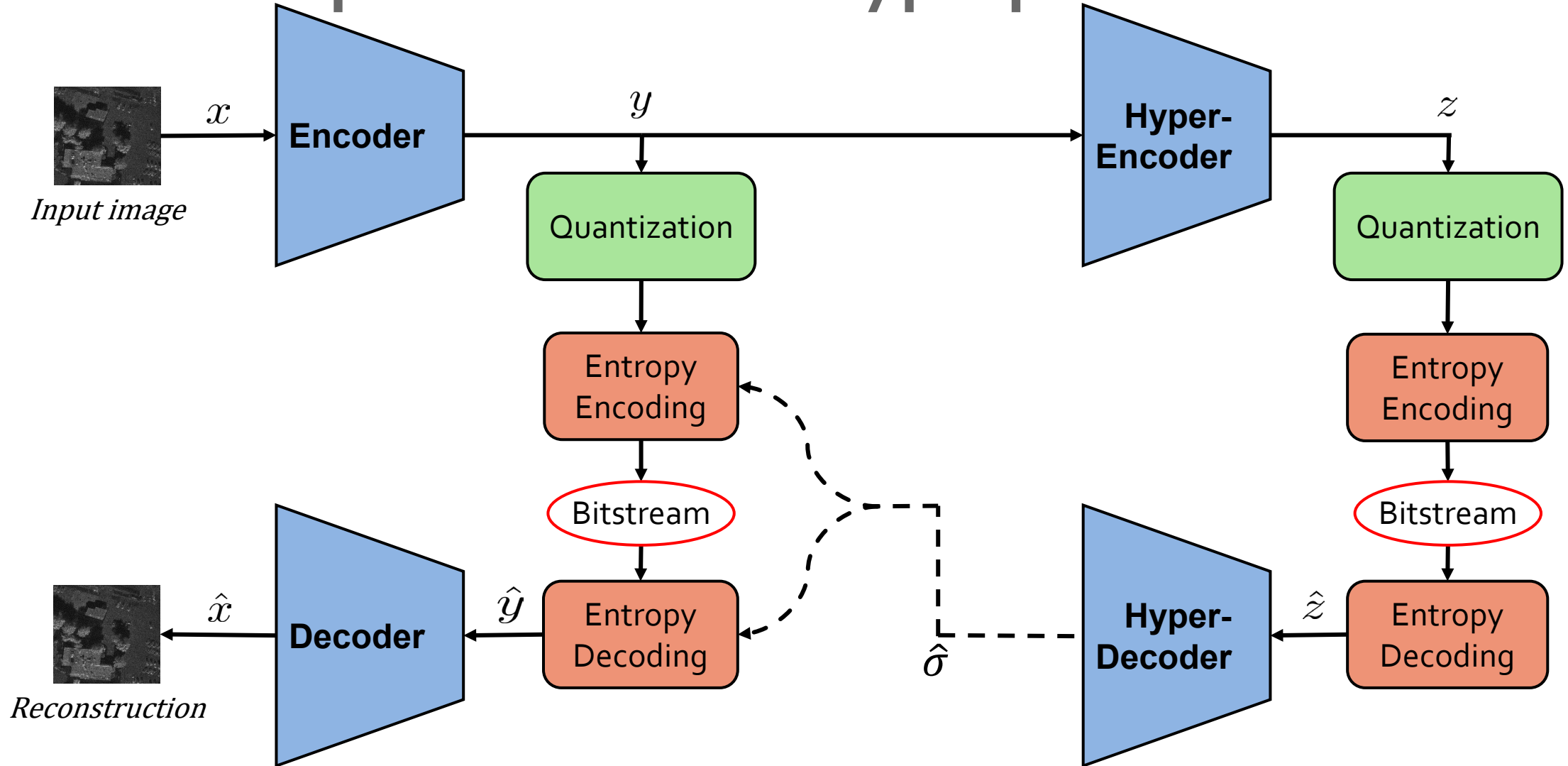
End-to-end optimization

$$\mathcal{L} = \mathcal{R} + \lambda \mathcal{D}$$

- \mathcal{R} is the rate in bits-per-pixel (*bpp*)
- \mathcal{D} is the distortion, e.g., Mean Squared Error
- λ is a Lagrangian multiplier

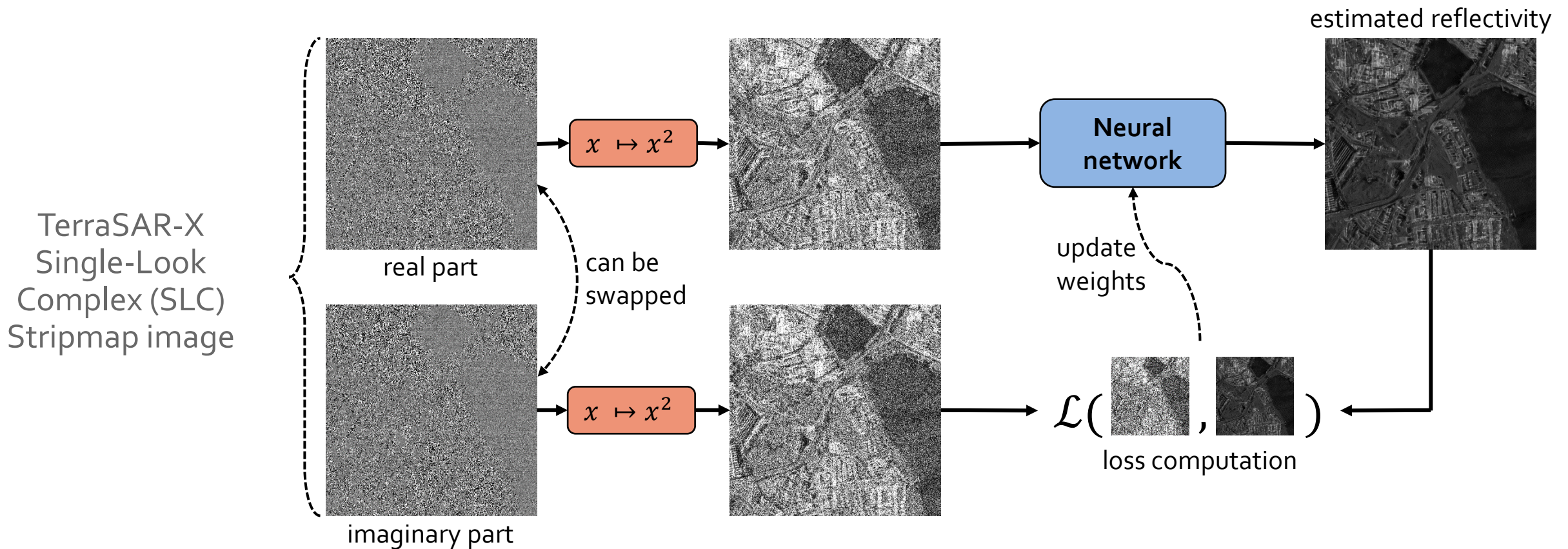
Ballé, J., Laparra, V., & Simoncelli, E. P. (2017). End-to-end Optimized Image Compression (No. arXiv:1611.01704). *arXiv*.

Neural compression: scale hyperprior



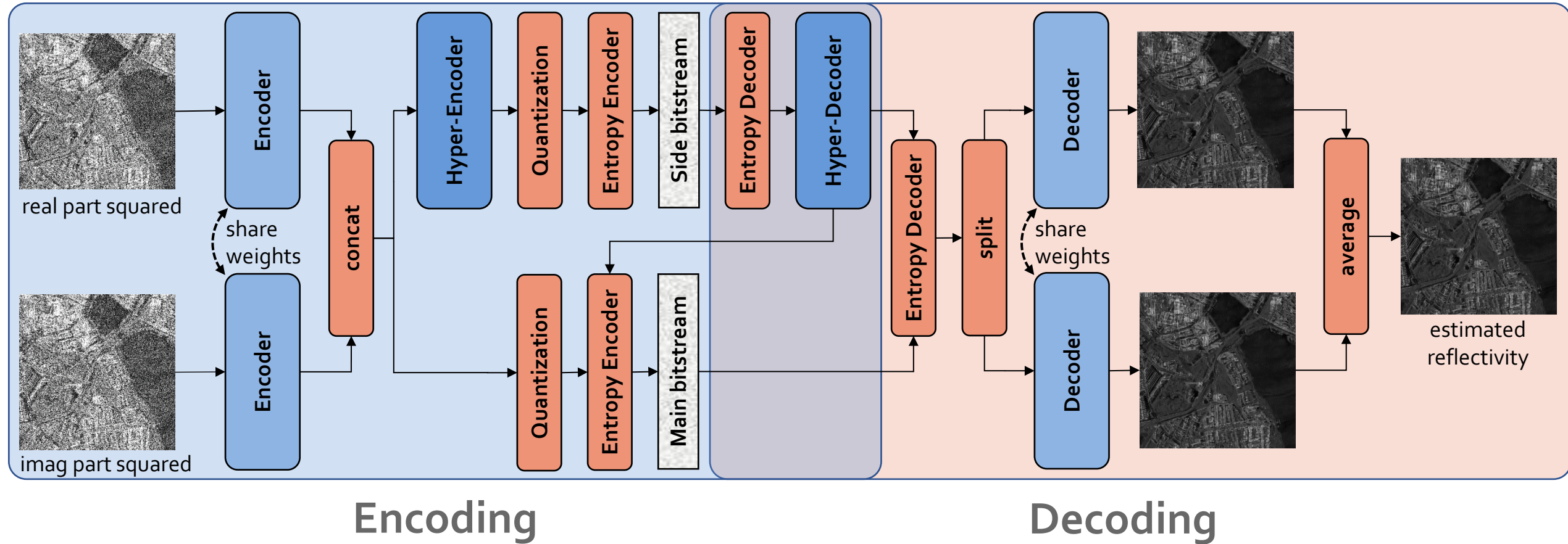
Ballé, J., Minnen, D., Singh, S., Hwang, S. J., & Johnston, N. (2018). Variational image compression with a scale hyperprior. *International Conference on Learning Representations*.

Despeckling SAR using MERLIN



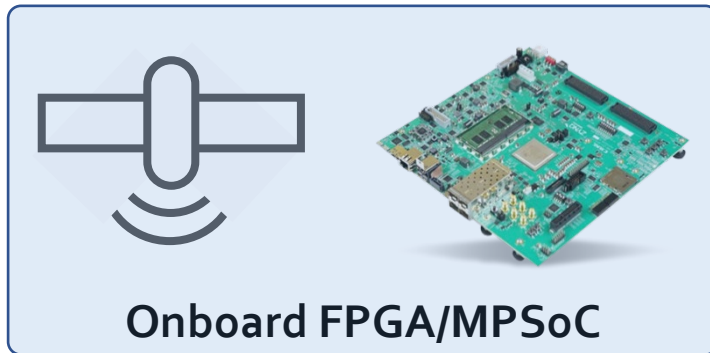
Dalsasso, E., Denis, L., & Tupin, F. (2022). As if by magic: Self-supervised training of deep despeckling networks with MERLIN. *IEEE Transactions on Geoscience and Remote Sensing*

Complete framework for inference

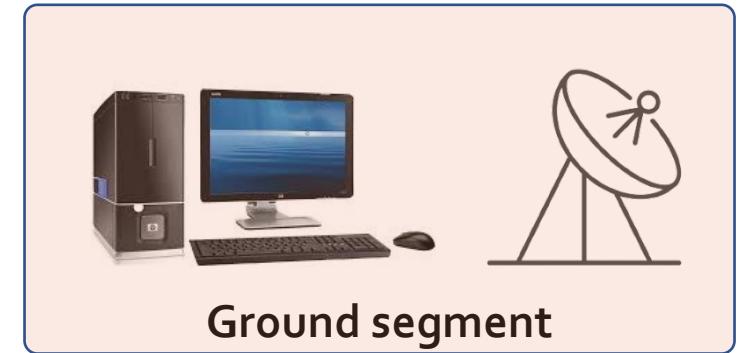
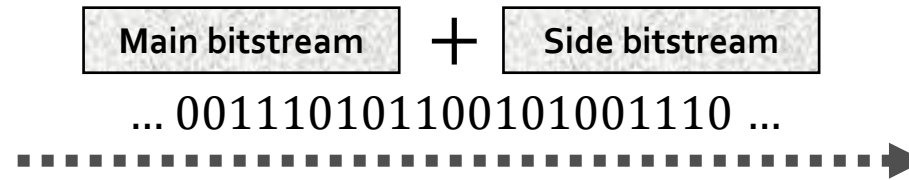


Amao-Oliva J., Foix-Colonier N., & Sica F. (2024). Joint compression and despeckling by SAR representation learning. *ISPRS Journal of Photogrammetry and Remote Sensing*.

Execution environment: onboard FPGA/MPSoC



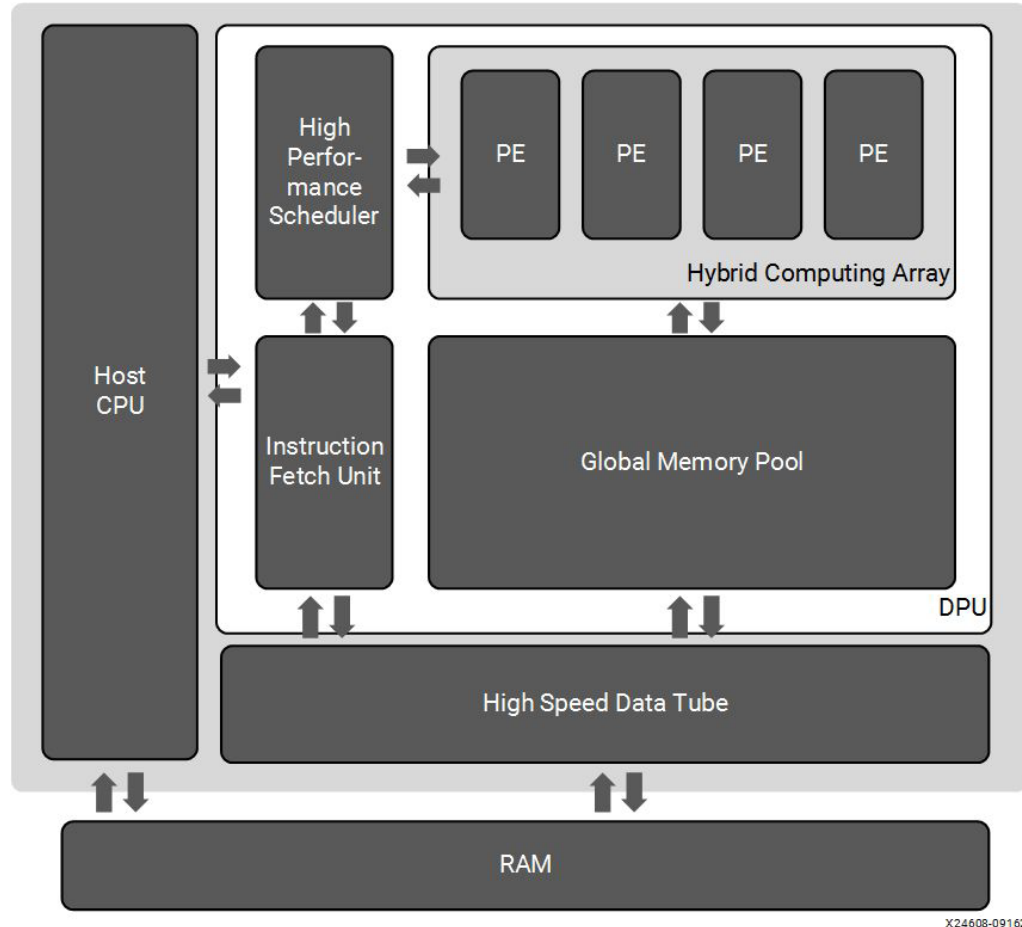
Encoding



Decoding

- CPUs / GPUs have hardened general-purpose hardware units
- FPGAs are reconfigurable integrated circuits
- AMD Zynq UltraScale+ is a Multiprocessor System-on-Chip (MPSoC)
 - FPGA Programming Logic
 - CPU Processing System

Designing FPGAs using automated deployment



Deployment steps:

- Inspection
- Quantization
- Compilation
- Execution

Requires network adaptation:

- Custom operations
- Convolutions simplification

Vitis AI Deep-Learning Processing Unit, DPUCZDX8G schema from [1]

1. AMD. (2019). *Vitis™ AI Software*. [Vitis AI overview v3.5 User Guide (UG1414)].

Preliminary Results

Input



CR = 1

Reconstruction ($\lambda = 25$)
 $bpp \approx 10.16$



CR \approx 6.31

MERLIN
 $bpp = 32$



CR = 2



Preliminary Results

Input



CR = 1

Reconstruction ($\lambda = 1$)
 $bpp \approx 1.64$



CR \approx 39.02

MERLIN
 $bpp = 32$



CR = 2



Preliminary Results

Input



CR = 1

Reconstruction ($\lambda = 0.05$)
 $bpp \approx 0.13$



CR \approx 4830

MERLIN
 $bpp = 32$



CR = 2



Future steps

- Tweaks needed to improve reconstruction quality
- Adapt the network for the DPU on the FPGA
- Quantitative evaluation of despeckling



**Deutsches Zentrum
für Luft- und Raumfahrt**
German Aerospace Center



Chair of Computer Architecture and Parallel Systems
TUM School of Computation, Information and Technology
Technical University of Munich

Takeaways

- Saving memory, bandwidth, and on-ground processing
- Small compressed model \Rightarrow Minimal payload
- Can be paired with detection methods for real-time monitoring



**Munich School for
Data Science**
HELMHOLTZ · TUM · LMU



Thanks for listening

