Zürich, 02.06.2025
Space Data
ETH Zurich

Project 3 – MACHINE LEARNING

Lorenzo Arturo & Yara Luginbühl

CONTENTS

1 INTRODUCTION

The objective of this project is to evaluate and compare different architectures of convolutional neural networks, reflecting the evolution of CNNs in the field of image denoising [1]. Three architectures are tested on the same dataset of 19k noisy lunar surface images of size 128 x 128 px and compared in their ability to denoise the images. As a reference, a dataset of clean images will be used.

Three distinct architectures are explored and optimized. The baseline is a simple CNN to provide a point of comparison. The second is a U-Net, known for its effectiveness in image-to-image tasks. The third architecture leverages transformer-based structures to assess their potential in high-resolution denoising. Finally, U-Net and Transformer structures are combined to explore the potential of advanced architectures.

The code and outputs can be accessed at: https://github.com/YaraLuginbuehl/MachineLearningProject.

2 METHODOLOGY

2.1 Simple CNN

The baseline model implemented here is a simple convolutional neural network (CNN) designed for grayscale image denoising. It consists of two convolutional layers with simple ReLU activations, batch normalization to stabilize training and MSE as a loss function. The other relevant parameters of the model are given in the appendix in Table A.1.

2.2 U-NET

The U-Net architecture, introduced in 2015, is an improved version of a basic CNN. It consists of two paths, where the encoder path progressively downsamples the input to capture context and the decoder path upsamples to enable precise localization. Moreover, these two paths combine features from both paths over skip connections [2].

The same parameters as in the previous architecture were used for this model. The additional, U-NETspecific parameters are given in Table A.2. To understand the impact of four kinds of variables, an analysis structured in two parts was done: In the first run (Model 1 - 4) two different model depths (2 layers, 4 layers) and activation functions (Relu and Leaky Relu) were used. The model with the best result was then used in the second run (Model 5 - 8), where two different optimisers (Adam and AdamW) and learning rates (0.001 and 0.0005) were compared.

2.3 Transformer

This denoising model uses a Vision Transformer (ViT)-style encoder to better preserve fine-grained lunar features (e.g., craters, ridges, isolated hills) that basic CNNs often smooth out due to their local receptive fields and pooling. Instead of sliding convolution kernels, the image is divided into fixed patches and each is linearly embedded. This allows the Transformer to model global dependencies across all patches simultaneously — capturing both broad structures and subtle, spatially dispersed details (like small dark dots or textured highlands). By using self-attention, the model dynamically weighs relationships between all patches, rather than relying solely on spatial locality. This helps retain weak but meaningful contrasts that often vanish in CNN outputs. A lightweight CNN refinement block is applied after the image is reassembled to restore local continuity without over-smoothing. Parameters are depcited in Table A.5.

2.4 UNET-Transformer-hybrid

The hybrid U-Net Transformer architecture combines the strengths of convolutional and attention-based models for superior image denoising. U-Net contributes strong spatial precision via its encoder-decoder structure and skip connections, while the Transformer bottleneck introduces global context modeling by attending across all spatial patches simultaneously. This fusion mitigates U-Net's limited receptive field and the Transformer's weakness in fine-grained localization.

The processing parameters are given in Table A.5. After Transformer processing, the feature maps are reshaped and upsampled through a two-step transposed convolution decoder. The decoder merges local features from the encoder and global features from the Transformer, producing denoised images with improved structural fidelity and preserved fine details.

2.5 Metrics

To obtain a robust and reliable estimate of the performance, the different architectures were all run with a 5-fold cross validation. The quality of the models is then compared looking at the Mean Squared Error (MSE), the image quality metrics Peak SNR (PSNR) and Structural Similarity Index Measure (SSIM) as well as the run time and if relevant loss functions. For each fold, the values were calculated for each validation image and averaged. For a more qualitative analysis, the ouptut images as well as the difference of the clean versus predicted images were analysed.

3 RESULTS

3.1 CNN

The metrics resulting from the simple CNN models averaged for the five folds are given in the following table (Table 3.3). An example image of the denoising as well as the loss functions can be found in the Appendix A.1.

	MSE	**PSNR [dB]**	**SSIM**	**Runtime [s]**
CNN_Model 1	0.0029 ± 0.0003	27.01 ± 0.14	0.8408 ± 0.0012	1276 ± 6

Table 3.1. *Mean ($\pm$ standard deviation) for MSE and image quality metrics of the simple CNN model.*

3.2 U-NET

The results of the UNET models with varying parameters averaged over the folds can be found in Tables A.3 and A.4. For the models with 4 layers (models 3 and 4) plateauing was apparent looking at the loss functions (see Figure A.4). Therefore, two layers and the leaky relu activation function (model 2) were used for the second batch of tests, since these parameters also slightly outperform the basic relu (model 1) in the image quality metrics. The results of the second run can be found in Table A.4. The best model resulting from these tests is model 5 with the Adam optimiser and a learning rate of 0.001, resulting in the following values. An image example as well as the loss function can be found in Appendix A.2.

	MSE	**PSNR [dB]**	**SSIM**	**Run time [s]**
UNET_Model 5	0.00227 ± 0.00005	28.45 ± 0.21	0.9072 ± 0.0013	968 ± 9

Table 3.2. *The best resulting means ($\pm$ standard deviations) for MSE and image quality metrics of the U-NET testing.*

3.3 Transformer

Quantitative results for the Transformer model are summarized below, averaged over 5 cross-validation folds. Additional outputs and performance figures are available in Appendix A.3.

	MSE	**PSNR [dB]**	**SSIM**	**Runtime [s]**
Transformer_Model 1	0.004042 ± 0.001377	25.40 ± 0.97	0.8120 ± 0.0133	86.59

Table 3.3. *Mean performance metrics (± standard deviation) for the Transformer model over 5 folds.*

3.4 UNET-Transformer-Hybrid

Results for the hybrid model are shown below, along with representative sample outputs and loss plots provided in Appendix A.4. The table presents evaluation metrics for a single trained model instance.

	MSE	**PSNR [dB]**	**SSIM**	**Runtime [s]**
UNET_Transformer_Model 1	0.001809 ± 0.000708	29.07 ± 1.31	0.8996 ± 0.0151	1687.087

Table 3.4. *Performance metrics for the hybrid U-Net Transformer model (single run).*

4 DISCUSSION

The U-NET architecture shows better results than the simple CNN: the loss decreased 20%, both image quality metrics increased and the run time was shorter. Although Figures A.1 and A.3 indicate that both models are not fully capable of depicting the crater structures, the CNN model shows a higher residual error, whereas the U-NET's are smoother and less intense (as reflected in the metrics). Both loss functions show expected decreases with fluctuations reflecting healthy learning (Figures A.5 and A.4). While the LRELU activation was able to improve the results of the 2-layer model, the impact of the other parameters was less pronounced: The optimiser did not have any impact and the learning rate's impact was ambiguous. This, as well as the reason and possible circumventions for the plateauing of the losses in the 4-layer model could be analysed in more detail using grid-tests and automated searches.

Figures A.7, A.8, and A.9 demonstrate that the Transformer model performs effective denoising with partial terrain structure preservation. However, it suffers from resolution loss and oversmoothing, which obscure finer surface details. While the overall reconstruction is acceptable, subtle features are noticeably diminished. By combining the strengths of standalone U-Net and Transformer models, the hybrid architecture achieves improved prediction accuracy. Comparing Figure A.9 (Transformer) with Figure A.15 (Hybrid) reveals a clear improvement in resolution, particularly for smaller terrain artifacts. Subtraction maps further highlight this difference in reconstruction fidelity—see Figures A.10 and A.16.

Among the tested models, the UNET-Transformer-Hybrid architecture achieved the lowest MSE (0.001809) and highest PSNR (29.07 dB), indicating superior reconstruction fidelity and noise suppression. It also reached a high SSIM score of 0.8996, closely matching the UNET's structural accuracy while significantly outperforming the Transformer model (SSIM = 0.8120). In contrast, the Transformer model exhibited the fastest runtime (86.59 s) but at the cost of lower image quality, with the highest MSE (0.0040) and lowest PSNR (25.40 dB). This highlights a clear trade-off between denoising performance and computational efficiency across the architectures. A prominent round artifact near the top middle edge of the image A.9 shows a clear qualitative difference between the models. In the Transformer difference map A.10, this region appears closer to white, indicating a more accurate reconstruction relative to the clean image. In contrast, the Hybrid model underestimates this structure, as shown by the stronger red shading, suggesting that the Transformer better captures large-scale artifacts due to its global attention mechanism.

BIBLIOGRAPHY

[1] M. Elad, B. Kawar, and G. Vaksman, “Image denoising: The deep learning revolution and beyond – a survey paper –,” 2023. [Online]. Available: https://arxiv.org/abs/2301.03362

[2] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, 2015, pp. 234–241.

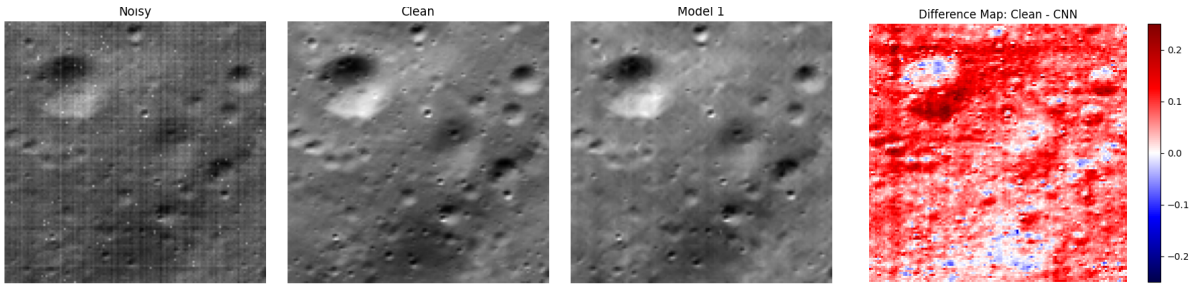
A APPENDIX

The images shown here are included for completeness. They can be found in the code submitted with this report for more detailed viewing.

A.1 Simple CNN

Parameter	Value
Nr of Layers	2
Conv2d Channels	(64, 128)
BatchNorm Channels	(64, 128)
Kernel Size	3 x 3
Activation	ReLU
Optimizer	Adam
Learning Rate	0.0005
Loss Function	MSE Loss
Epochs	50
Batch Size	32
Data Split	80/20

Table A.1. *Parameters of the simple CNN model*



(a) *Example of image correction* **(b)** *Difference*

Figure A.1. *Resulting Images Simple CNN*

Figure A.2. *Loss Function*

A.2 U-NET

Parameter	Value
Variable parameters	Nr of layers, activation fct, optimiser and learning rate
Symmetric Model	Yes
Skip Connections	Yes
Pooling Type	MaxPooling (2×2)
Upsampling Method	Transposed Convolution (2×2, stride 2)
Bottleneck Channels	128
Encoder Channels	2 layers: (64, 128), 4 layers: (32, 64, 128, 256)
Decoder Channels	2 layers: (128, 64), 4 layers: (256, 128, 64, 32)
BatchNorm Channels	same as Conv2d Channels

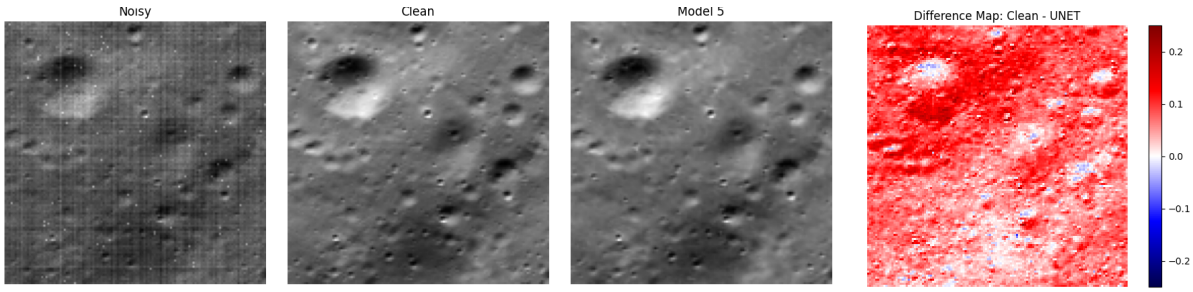
Table A.2. *Additional U-Net-specific Parameters*

	Layers	Activation	MSE	PSNR [dB]	SSIM	Runtime [s]
UNET_Model 1	2	Relu	0.00232 ± 0.00127	28.4 ± 0.3	0.906 ± 0.003	975 ± 4
UNET_Model 2	2	LeakyRelu	0.00228 ± 0.00004	28.42 ± 0.19	0.9076 ± 0.0015	984 ± 4
UNET_Model 3	4	Relu	0.00223 ± 0.00015	28.53 ± 0.06	0.9153 ± 0.0006	1209 ± 4
UNET_Model 4	4	LeakyRelu	0.00219 ± 0.00015	28.45 ± 0.15	0.9146 ± 0.0008	1229 ± 4

Table A.3. *Run 1: The resulting values of the UNET architecture for varying numbers of layers and activation functions.*

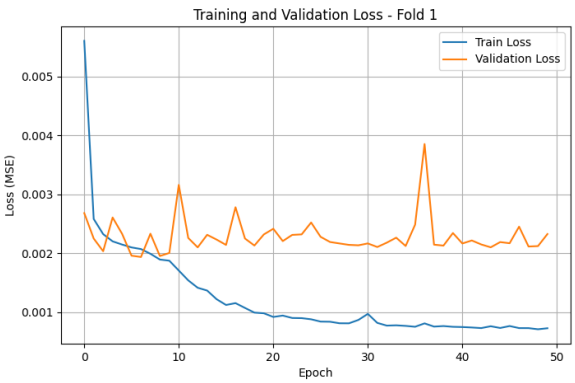
	Optimiser	LR	MSE	PSNR	SSIM	Run time
UNET_Model 5	Adam	0.001	0.00227 ± 0.00005	28.45 ± 0.21	0.9072 ± 0.0013	968 ± 9
UNET_Model 6	Adam	0.0005	= Model 2			
UNET_Model 7	AdamW	0.001	0.00227 ± 0.00005	28.45 ± 0.21	0.9072 ± 0.0013	985 ± 6
UNET_Model 8	AdamW	0.0005	0.00228 ± 0.00004	28.42 ± 0.19	0.9076 ± 0.0015	984 ± 6

Table A.4. *Run 1: The resulting values for the UNET architecture for varying optimisers and learning rates.*

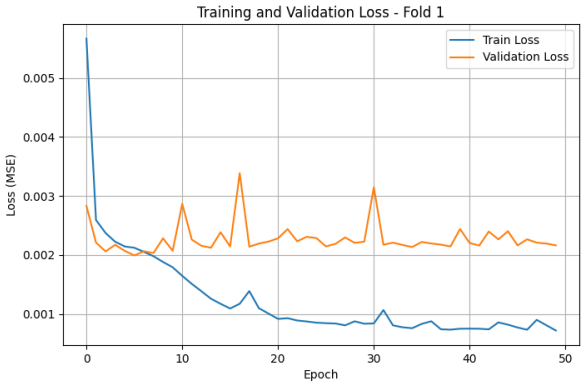


(a) *Example of image correction* **(b)** *Difference*

Figure A.3. *Resulting Images UNET_Model 5*



(a) *Model 3 Loss Functions (first fold)*



(b) *Model 4 Loss Functions (first fold)*

Figure A.4. *The loss functions of the two 4 layer UNET models.*

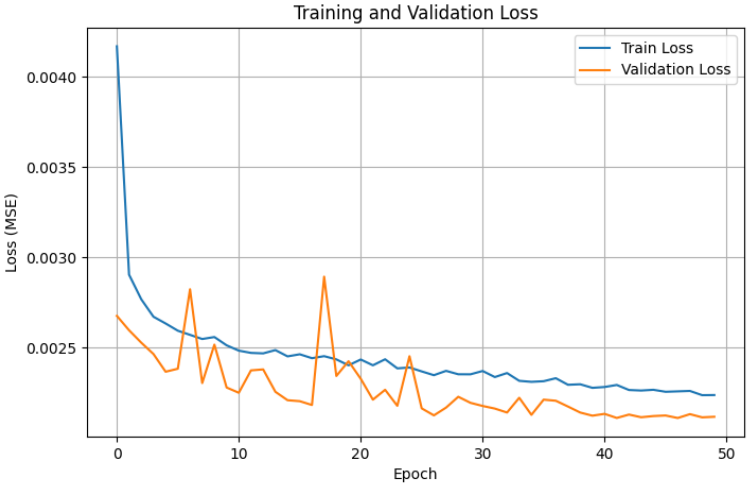


Figure A.5. *Loss Function UNET_Model5*

A.3 Transformer

Parameter	Value
Input Image Size	128×128
Patch Size	8×8
Patch Vector Dimension	$128 \times 8 \times 8 = 8192$
Latent Embedding Dimension	256
Transformer Encoder Layers	2
Number of Attention Heads	8
Feedforward Network Dimension	512

Table A.5. *Model and Transformer parameters*

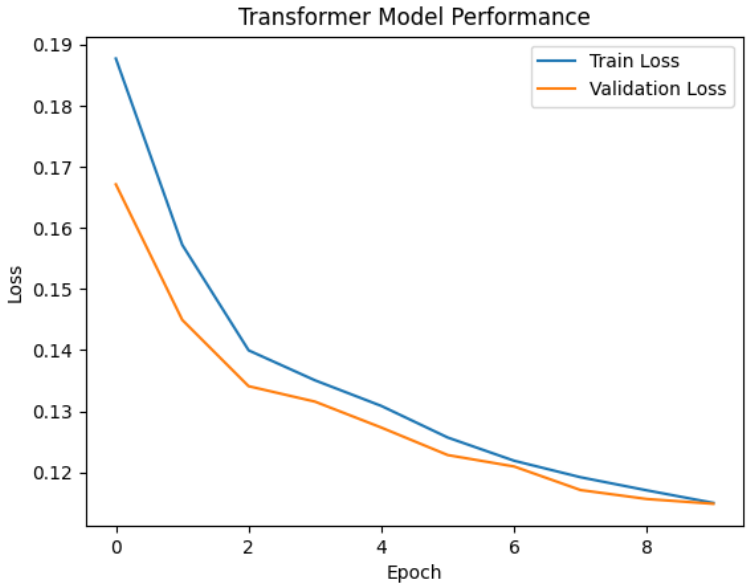


Figure A.6. *Training vs. validation loss for the Transformer model: Consistent convergence across 10 epochs with minimal signs of overfitting, indicating stable training behavior.*

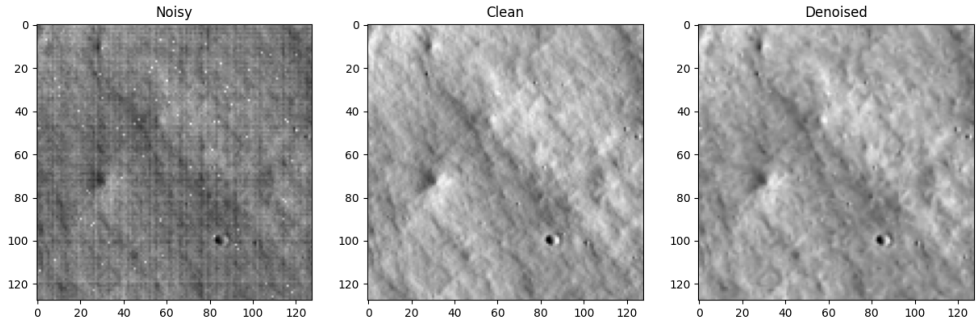


Figure A.7. *Transformer output on a training sample (0)*

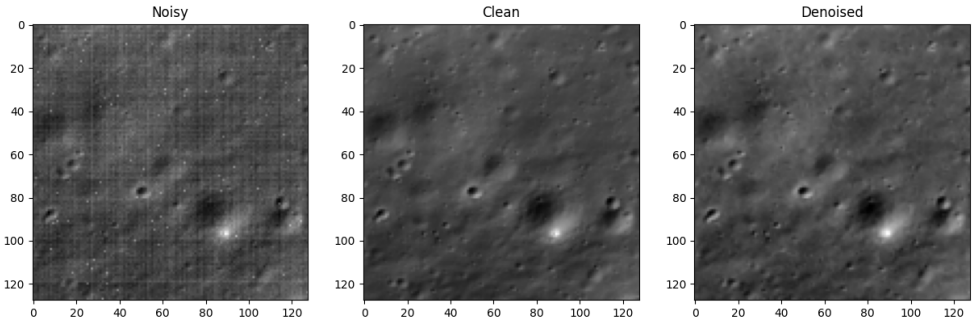


Figure A.8. *Transformer output on a training sample (1)*

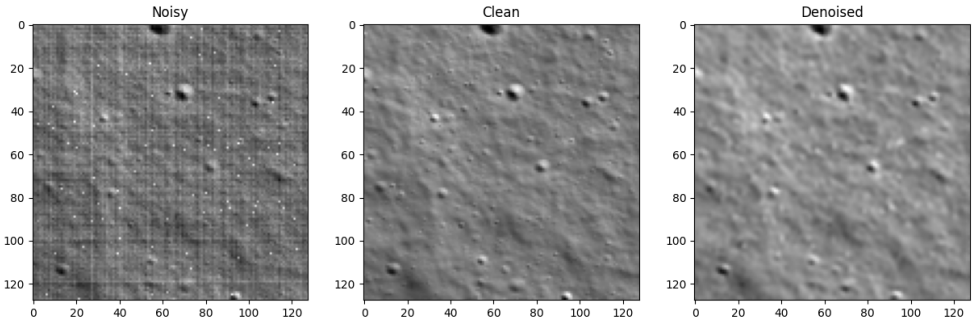


Figure A.9. *Transformer output on a validation sample (1)*

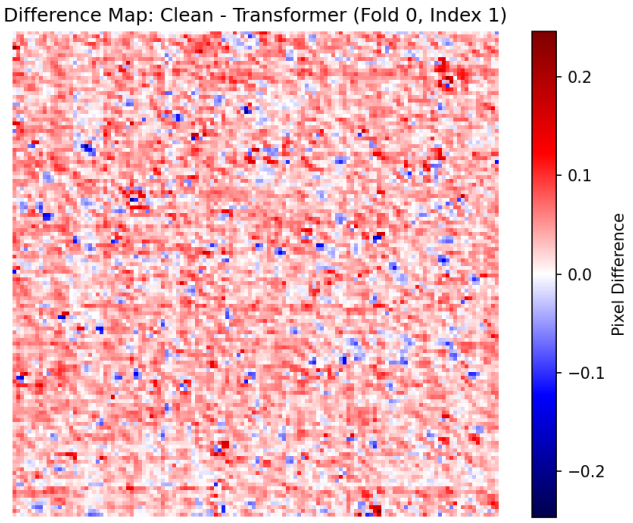


Figure A.10. *Difference map between clean image and Transformer output (Fold 0, Sample 1).*

A.4 UNET-Transformer-Hybrid

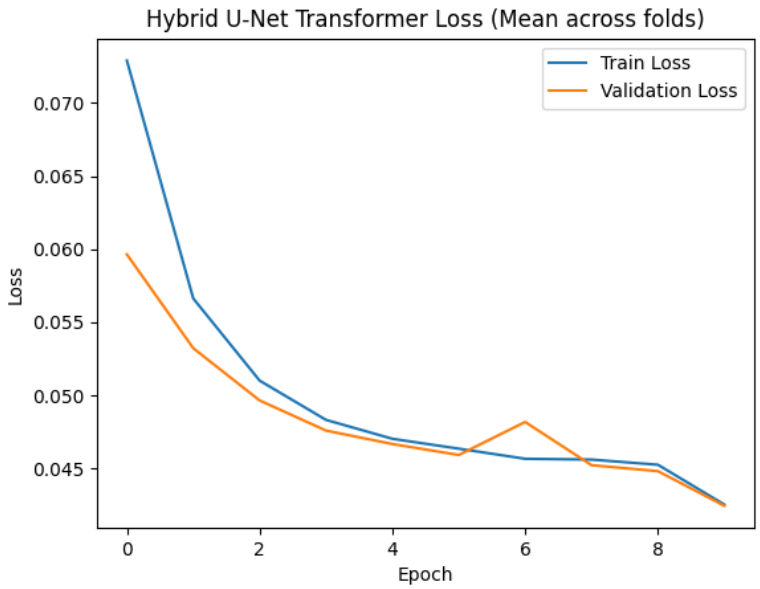


Figure A.11. *Mean training and validation loss over all folds using NaN-aware averaging. Shows the overall trend of the hybrid model.*

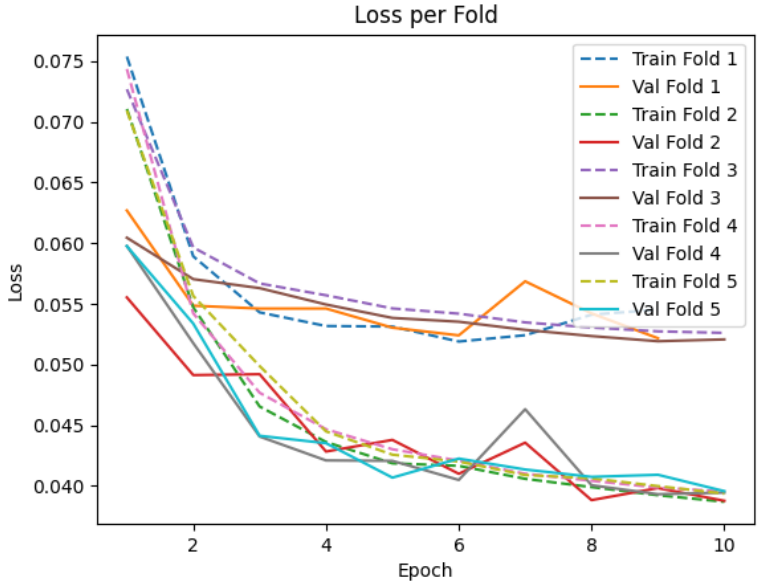


Figure A.12. *Fold-wise training and validation losses from 5-fold cross-validation. Visualizes performance variability per fold.*

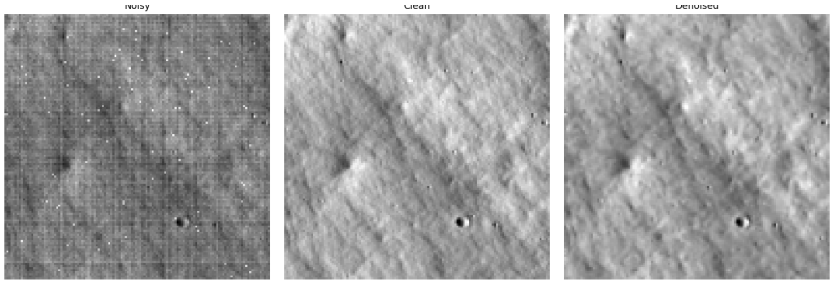


Figure A.13. *Hybrid model prediction vs. ground truth (Fold 1, Sample 0).*

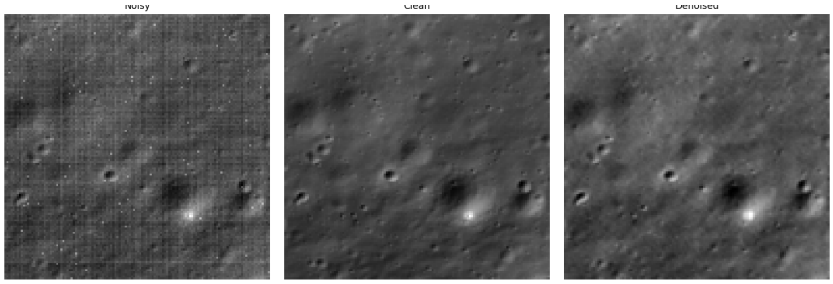


Figure A.14. *Hybrid model prediction vs. ground truth (Fold 3, Sample 0).*

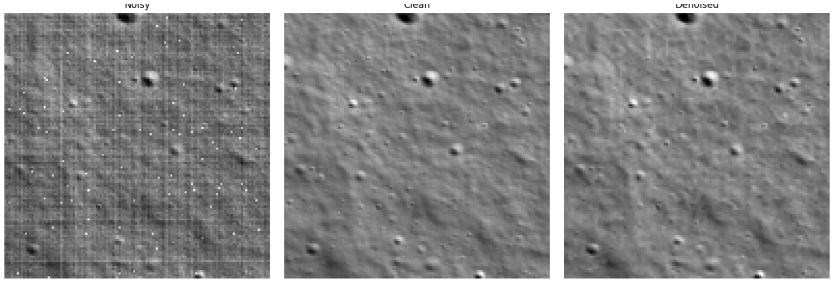


Figure A.15. *Hybrid model prediction vs. ground truth (Fold 0, Sample 1).*

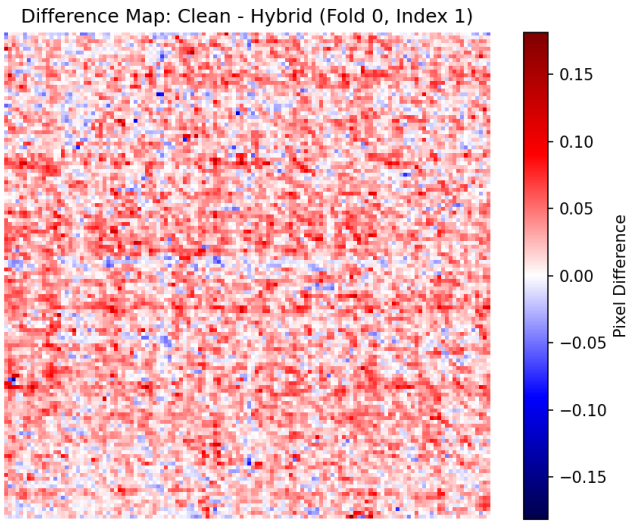


Figure A.16. *Difference map between clean image and Hybrid output (Fold 0, Sample 1).*