Optimizing Rank for High-Fidelity Implicit Neural Representations

Vanilla ReLU MLPs are secretly expressive.

Julian McGinnis[1,2], Florian Hölzl[3], Suprosanna Shit[4], Florentin Bieder[5], Paul Friedrich[5], Mark Mühlau[1], Björn Menze[4], Daniel Rueckert[1,2,†], Benedikt Wiestler[2,3,†]

[†]Shared senior authorship
[1]TUM [2]MCML [3]HPI [4]University of Zurich [5]University of Basel

The prevailing belief

Vanilla ReLU MLPs *are widely believed to be incapable of representing high-frequency content.*

In the last years, the INR community has primarily focused on developing **new architectural inventions** trying to overcome the **low-frequency bias of vanilla ReLU MLPs.**

CONTRIBUTION

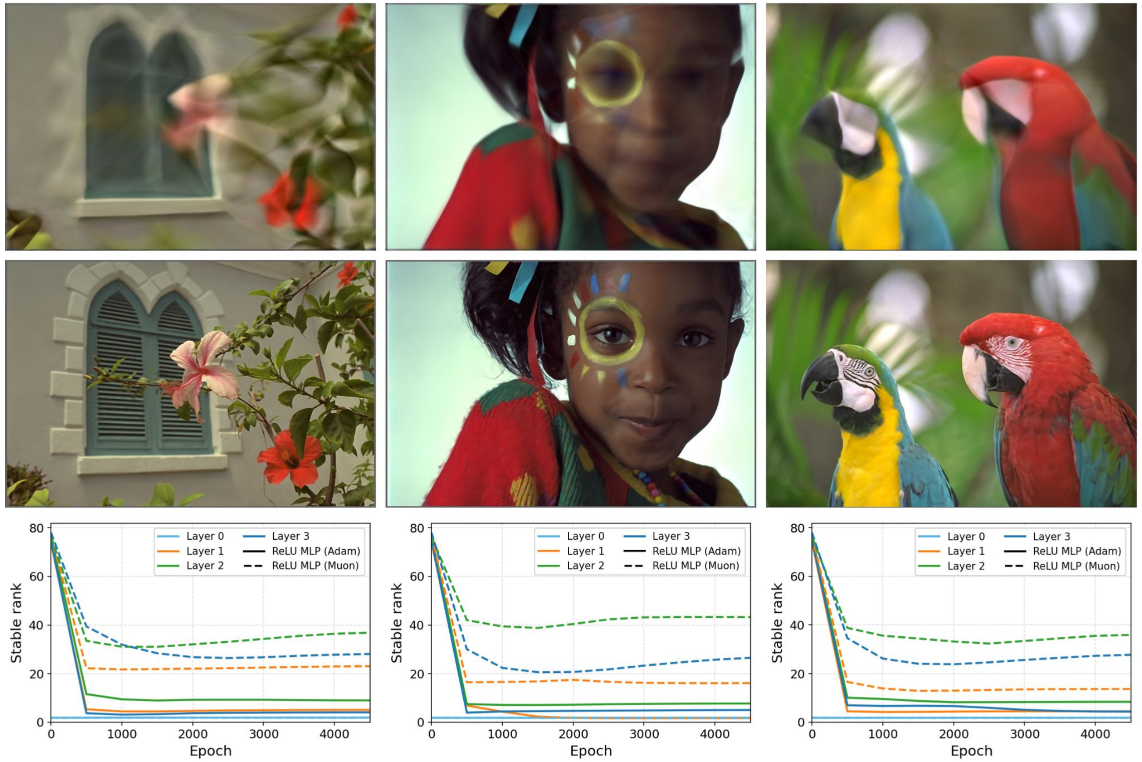
ReLU MLPs are secretly expressive

The problem isn't architecture - it is optimization!

We examine INRs using **a stable rank perspective** as a unifying lens **to explain existing architectural fixes.**

This allows us to explore a previously overlooked axis - **optimization-level interventions**.

The low-frequency bias of vanilla ReLU MLPs is not **an intrinsic architectural limitation** to learn high frequency content, but a **symptom** of stable-rank collapse during training with certain optimizers.

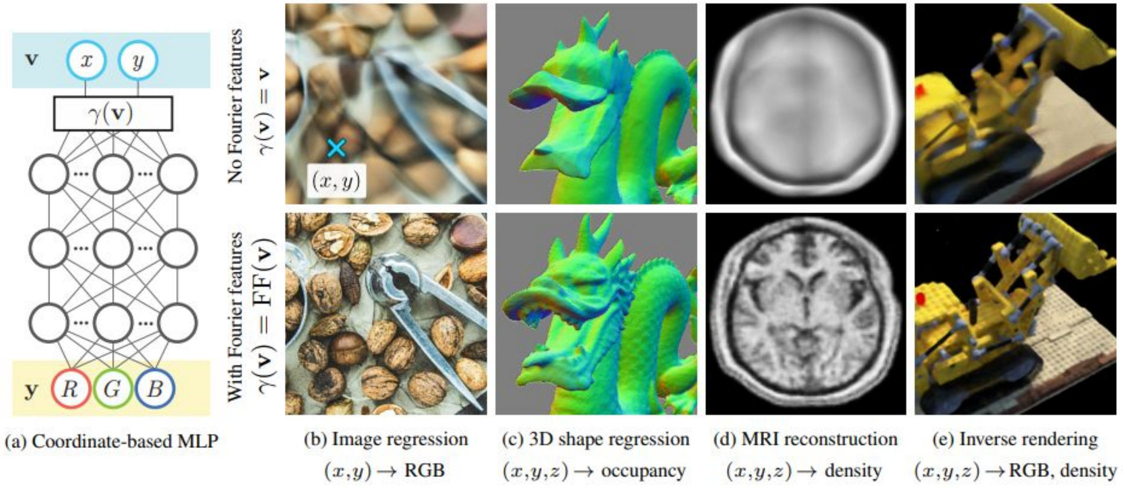


Lifting the stable rank at the input

Fourier Features (Tancik et al.)

Mechanism

Map coordinates x ⟶ γ(x) with sine-/cosine encoding to expand the input.



[1]

Result

Rank grows with the number of frequency bands and encoding dimensions lifting the first-layer bound.

Limitation

Fourier Features are static and the bandwidth σ must be tuned per task, adding an additional hyperparameter.

Holding the bound with depth

SIREN (Sitzmann et al.) [2]

Mechanism

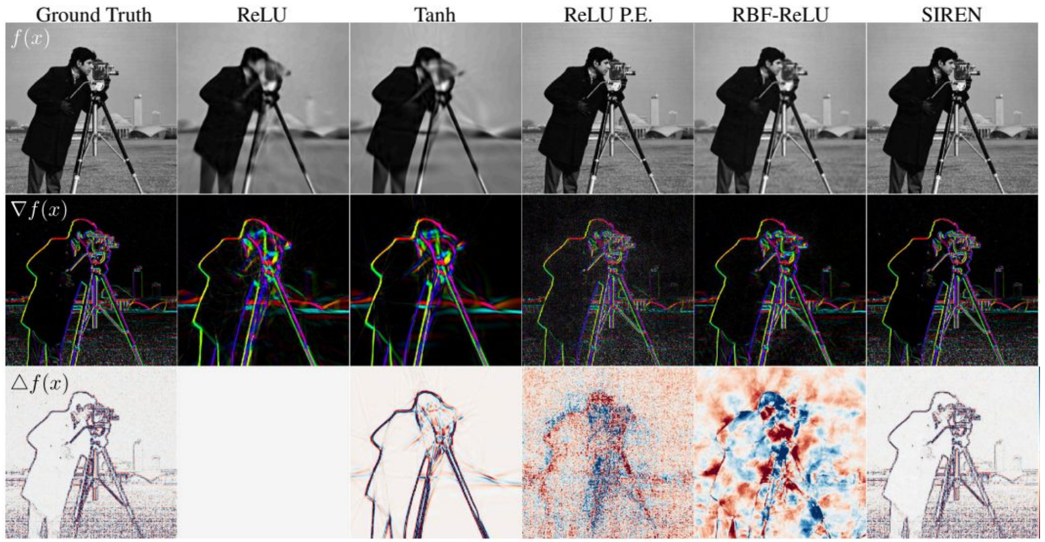
Directly act on the activations - use e.g. periodic sin(·) instead oF ReLU activations to resist rank decay through depth.

Result

Maintains rank deeper into the network sustaining the same bound across layers rather than only at the input.

Limitation

Requires specialized initialization and extra hyperparameter tuning. The factor ω must be selected for first and hidden layers.



[2]

Acting on the update directly

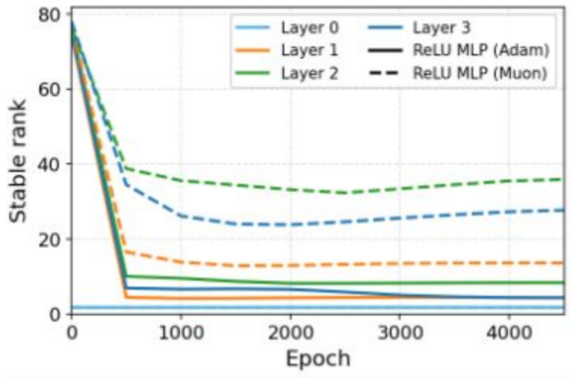
Using Muon (Jordan et al.) with orthogonalized, near-maximal-rank updates

Explored Mechanisms:

- **BatchNorm [4]** elevates activation rank through normalization
- **Rank Pre-Training [5]** elevates stable rank prior to INR optimization
- **Muon [3]** orthogonalizes each hidden-layer update, Ortho(U) = PQ^T, forcing it to high stable rank regardless of the activations.

Results:

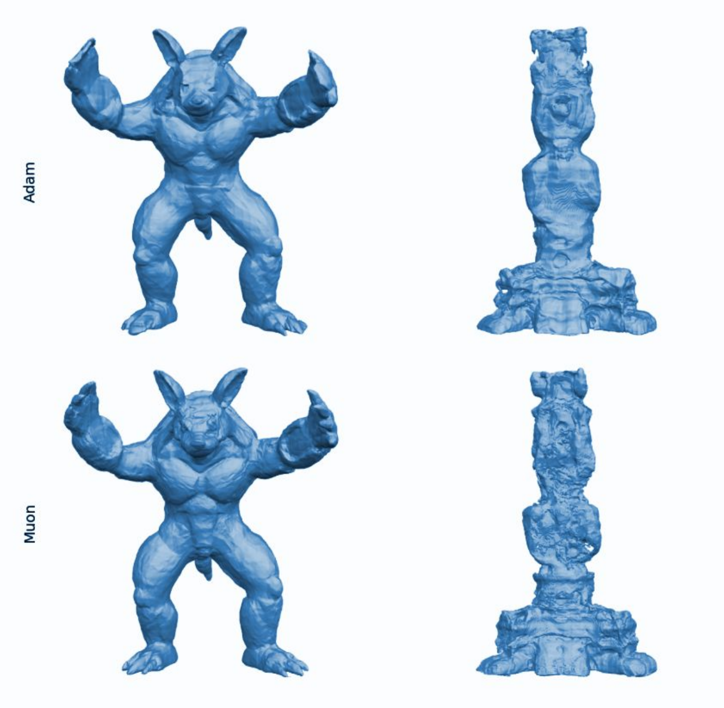
- **BatchNorm** holds rank but impedes learning
- **Rank pre-training** helps but does not enforce rank during training
- Muon as an update-level fix acts as the INR game-changer!



Elevating INR Expressiveness (Images, Audio, SDFs)

Table 2. **Audio overfitting performance (bach.wav).** Mean $\pm$ std over 5 seeds. Best-performing optimizer per row in **bold**.

Model	SNR (↑)		SI-SNR (↑)	
	Adam	Muon	Adam	Muon
ReLU MLP	0.02 $\pm$ 0.00	**0.14** $\pm$ 0.01	−23.52 $\pm$ 0.39	**−14.88** $\pm$ 0.47
ReLU FFN	8.88 $\pm$ 0.40	**21.43** $\pm$ 1.11	8.31 $\pm$ 0.45	**21.40** $\pm$ 1.11
Gauss MLP	8.76 $\pm$ 0.63	**12.85** $\pm$ 0.32	8.16 $\pm$ 0.73	**12.64** $\pm$ 0.34
Gauss FFN	37.80 $\pm$ 0.44	**46.80** $\pm$ 0.91	37.80 $\pm$ 0.44	**46.80** $\pm$ 0.91
SIREN	37.92 $\pm$ 0.19	**47.46** $\pm$ 0.68	37.92 $\pm$ 0.19	**47.46** $\pm$ 0.68
WIRE	3.65 $\pm$ 0.82	**15.52** $\pm$ 0.60	1.23 $\pm$ 1.39	**15.62** $\pm$ 0.63
FINER	27.22 $\pm$ 0.65	**36.48** $\pm$ 0.81	27.21 $\pm$ 0.65	**36.48** $\pm$ 0.81



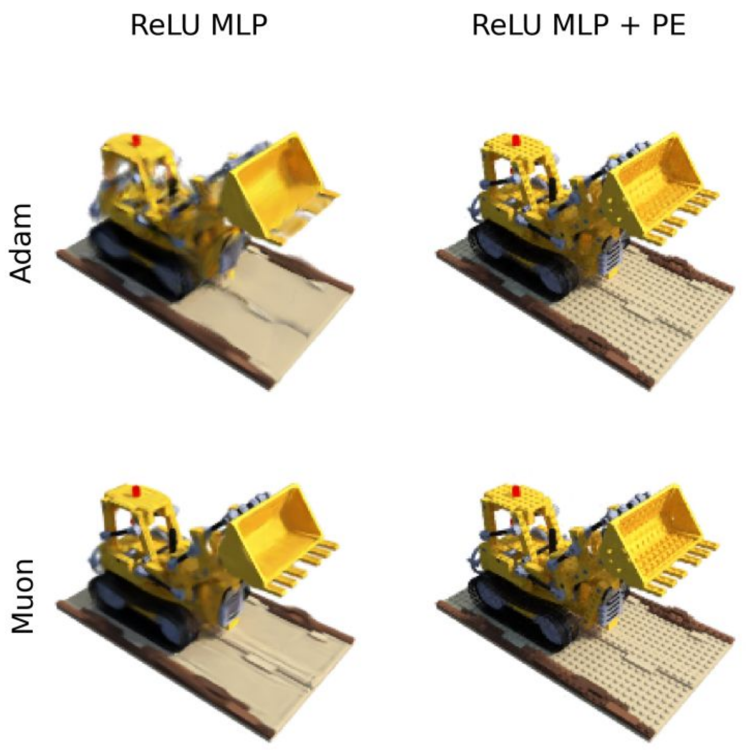
→ Muon uniformly boosts INR performance

→ Optimization-level interventions are additive to input-/ and activation-level interventions

Spectral Bias in Inverse Problems (NeRF, CT, SISR)

Table 4. **Sparse-view CT reconstruction performance**. Quantitative comparison of CT reconstruction quality for different architectures using the Adam optimizer versus the Muon optimizer. We report the mean and standard deviation for PSNR, SSIM, and LPIPS over 5 seeds. The best-performing optimizer for each architecture is highlighted in **bold**.

Model	PSNR (↑)		SSIM (↑)		LPIPS (↓)	
	Adam	Muon	Adam	Muon	Adam	Muon
ReLU MLP	25.86 ± 0.35	**32.93** ± 0.31	0.640 ± 0.014	**0.831** ± 0.004	0.528 ± 0.026	**0.251** ± 0.005
ReLU FFN	28.64 ± 0.60	**33.28** ± 0.27	0.756 ± 0.025	**0.839** ± 0.003	0.347 ± 0.058	**0.156** ± 0.010
Gauss MLP	26.01 ± 0.50	**27.35** ± 0.43	0.483 ± 0.031	**0.524** ± 0.023	0.405 ± 0.026	**0.340** ± 0.026
Gauss FFN	18.76 ± 1.03	**20.81** ± 0.18	0.177 ± 0.035	**0.225** ± 0.006	0.883 ± 0.074	**0.763** ± 0.014
SIREN	30.58 ± 0.43	**33.69** ± 0.12	0.803 ± 0.009	**0.850** ± 0.002	0.316 ± 0.011	**0.225** ± 0.003
WIRE	26.26 ± 0.36	**29.51** ± 0.72	0.471 ± 0.019	**0.644** ± 0.046	0.432 ± 0.017	**0.251** ± 0.039
FINER	31.93 ± 0.50	**32.92** ± 0.21	0.820 ± 0.016	**0.822** ± 0.006	0.251 ± 0.019	**0.215** ± 0.005



→ Optimization-level interventions may change the spectral bias

DISCUSSION

Conclusion

⟶ Vanilla ReLu MLPs are secretly expressive - if we challenge state of the art optimization
⟶ When designing INRs consider the third axis - optimization-level interventions

Limitations

⟶ Stable Rank is a useful diagnostic framework - but we don’t know optimal target ranks
⟶ Muon, as an exemplary instantiation of optimization-level interventions, improves but does not fully close the gap between vanilla ReLU MLPs and highly expressive activation functions such as SIREN and FINER

Future Work

⟶ INRs are a great testbed - can we scale to other model families and applications ?

EXPLORE OUR WORK

THANK YOU FOR JOINING!

For more details checkout our page and code!








Citations

[1] Tancik et al. *Fourier features let networks learn high frequency functions in low dimensional domains,* NeurIPS 2020.

[2] Sitzmann et al. *Implicit neural representations with periodic activation functions, NeurIPS 2020*

[3] Jordan et al. Muon: An optimizer for hidden layers in neural networks, 2024. https://kellerjordan.github.io/posts/muon

[4] Cai Z, Zhu H, Shen Q, Wang X, Cao X. Batch normalization alleviates the spectral bias in coordinate networks. CVPR 2024

[5] Daneshmand H, Kohler J, Bach F, Hofmann T, Lucchi A. Batch normalization provably avoids ranks collapse for randomly initialised deep networks. NeurIPS 2020