



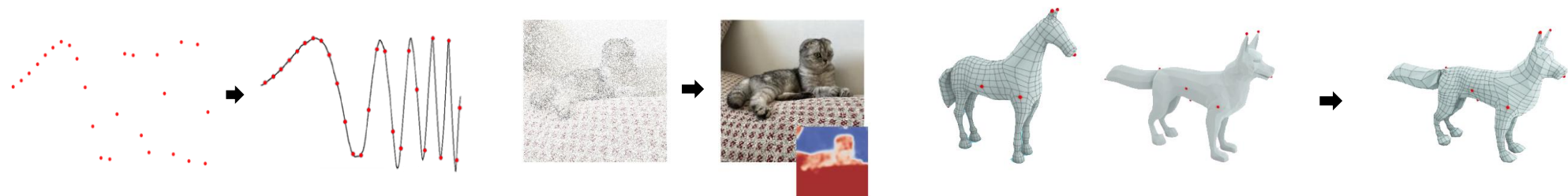
**The Blavatnik School
of Computer Science**
The Raymond and Beverly Sackler
Faculty of Exact Sciences
Tel Aviv University



SAPE: Spatially-Adaptive Progressive Encoding for Neural Optimization

*Amir Hertz¹, Or Perel¹,
Raja Giryes¹, Olga Sorkine-Hornung², Daniel Cohen-Or¹*

NeurIPS 2021



¹ Tel Aviv University

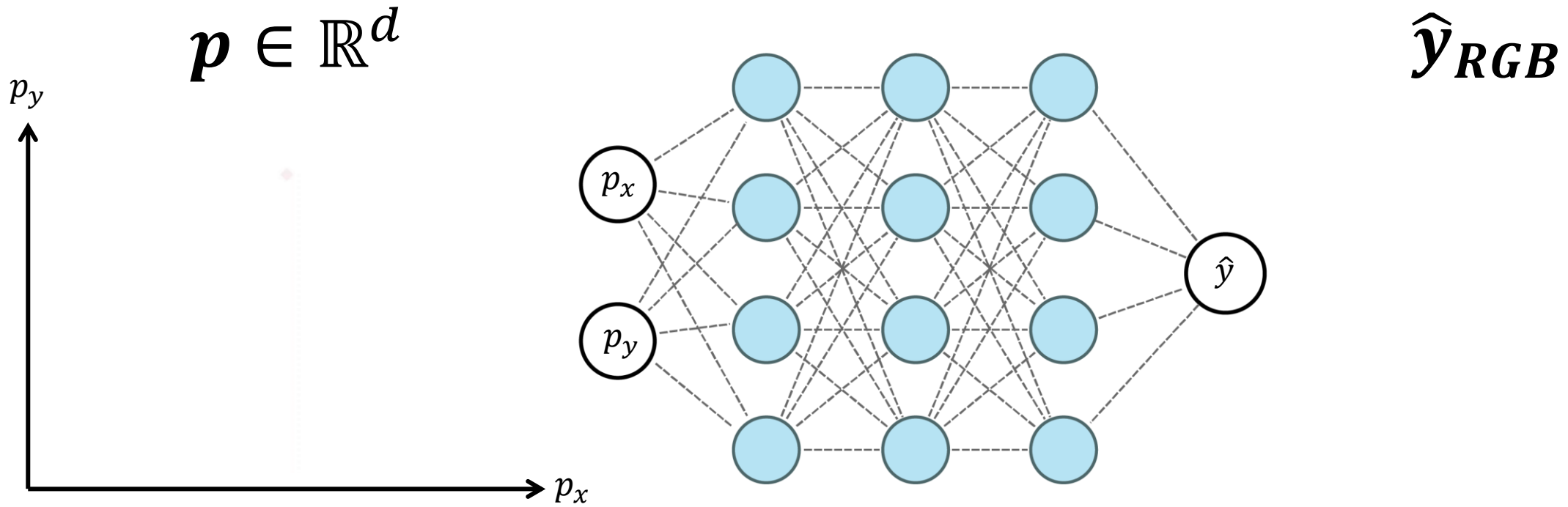
² ETH Zurich, Switzerland

Neural Implicit Functions

The image illustrates the concept of Neural Implicit Functions through several examples and diagrams:

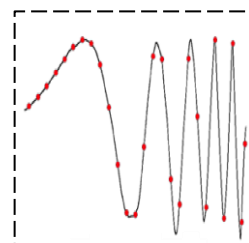
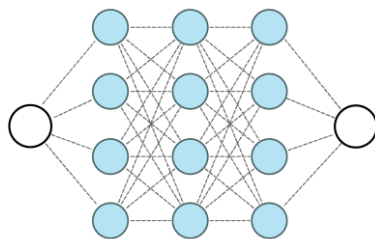
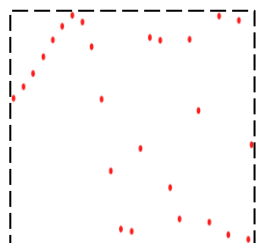
- Coordinate-based Neural Network:** A diagram labeled (a) shows a neural network with input features v (coordinates x and y) and a function $\gamma(v)$. The output is a vector y with components R , G , and B .
- Dragon Model:** A 3D model of a dragon is shown with a red grid overlay, representing the implicit function's output for a specific object.
- Starfish Texture Analysis:** Three columns of images show a starfish texture. The top row shows the original texture, and the bottom row shows the results of applying "Gradients" and "Laplacian" filters to the texture.
- Lamp Model:** A 3D model of a lamp is shown with a red grid overlay, similar to the dragon model.
- Ray-Tracing Diagram:** A diagram shows a 3D scene with a yellow excavator and a yellow lamp. Two rays, "Ray 1" and "Ray 2", are shown passing through the scene. A neural network F_θ is shown processing the rays to generate the final image.

Neural Implicit Functions



Our Setting

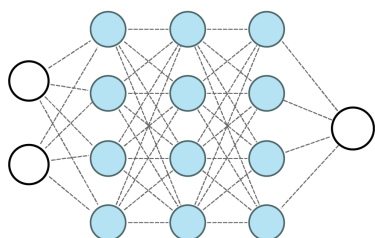
1D Coordinates



1D Signal
Magnitude



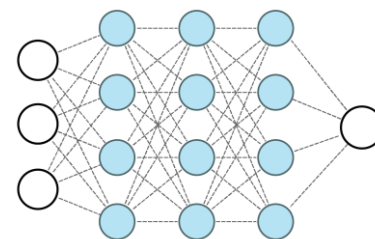
2D Coordinates



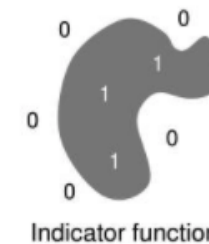
2D Pixel
Color



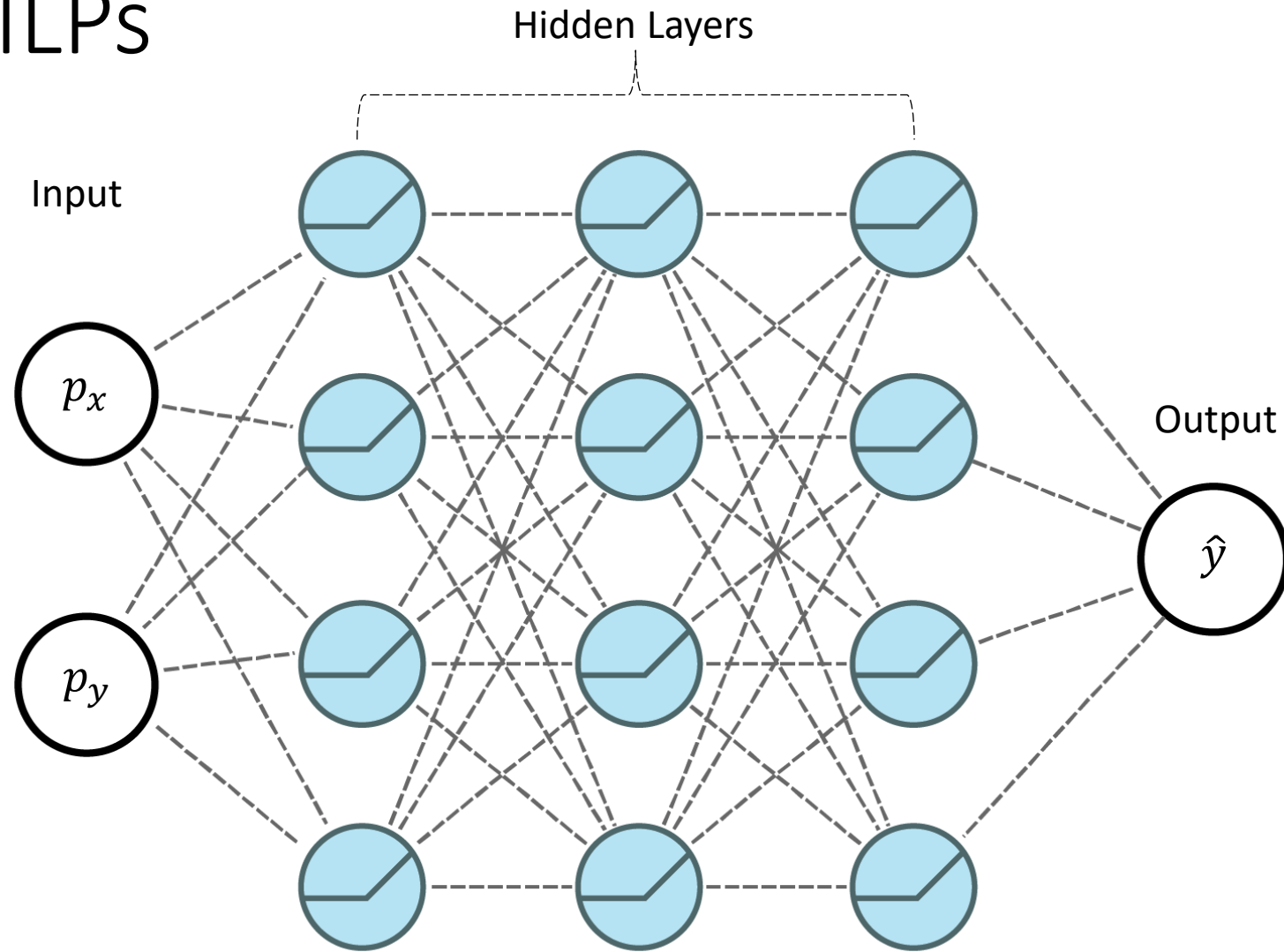
3D Coordinates



3D Volume
Occupancy



ReLU MLPs

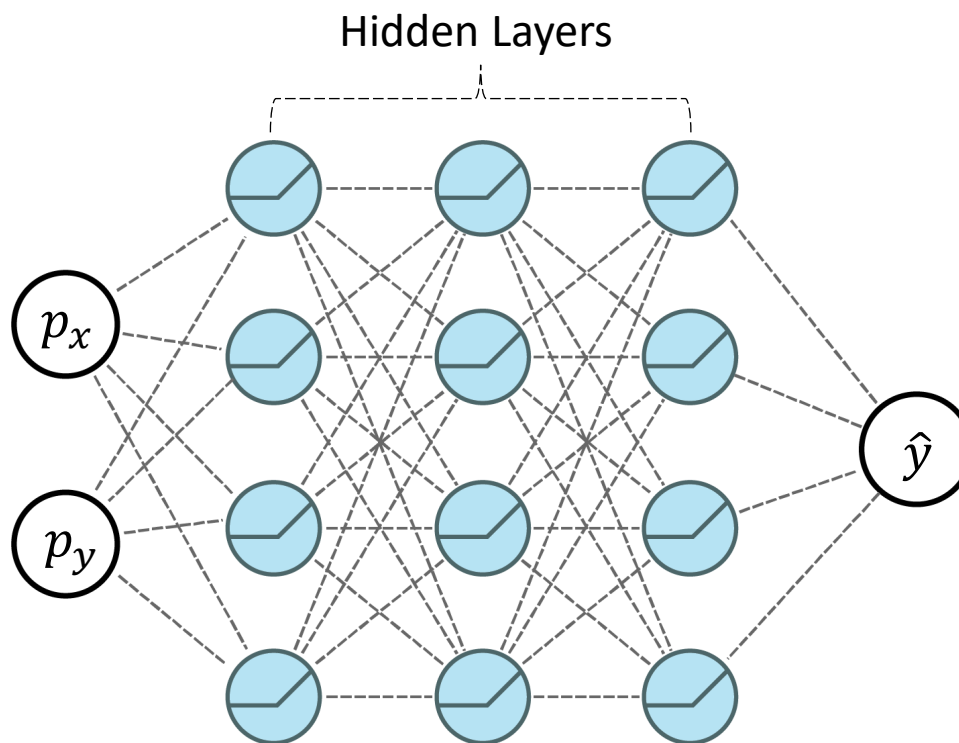
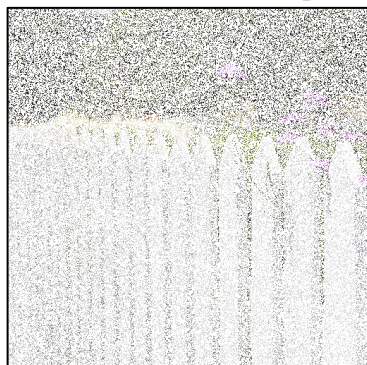


ReLU MLPs

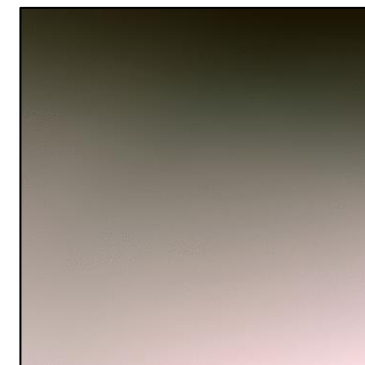
Ground Truth y



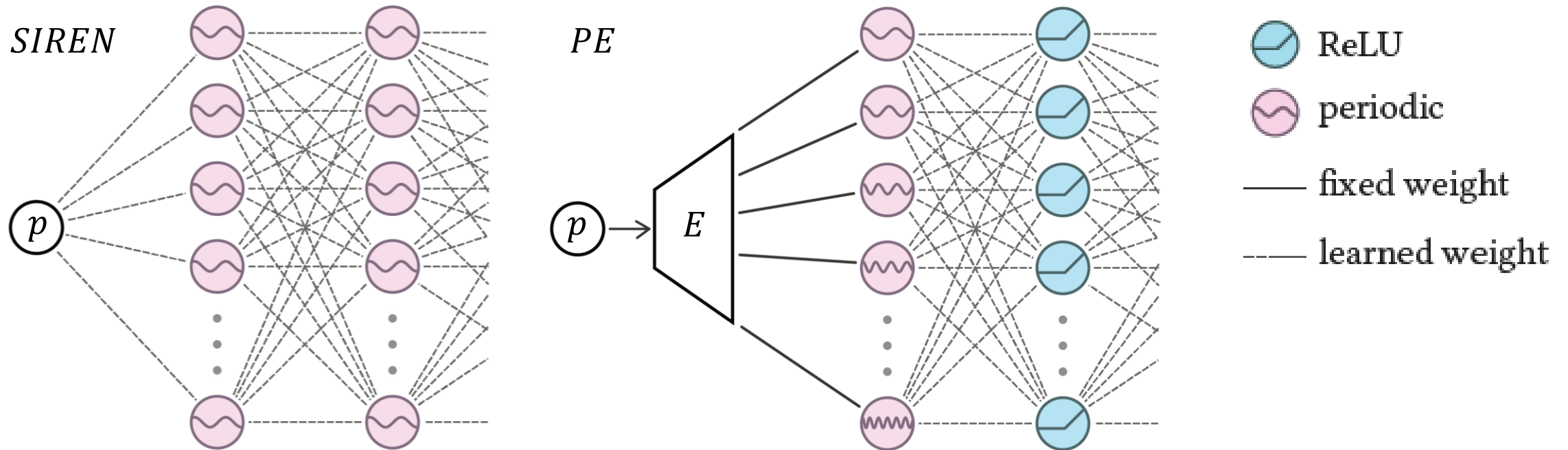
Coord / Intensity
Samples $(\mathbf{p}, \mathbf{y}_p)$



Output \hat{y}



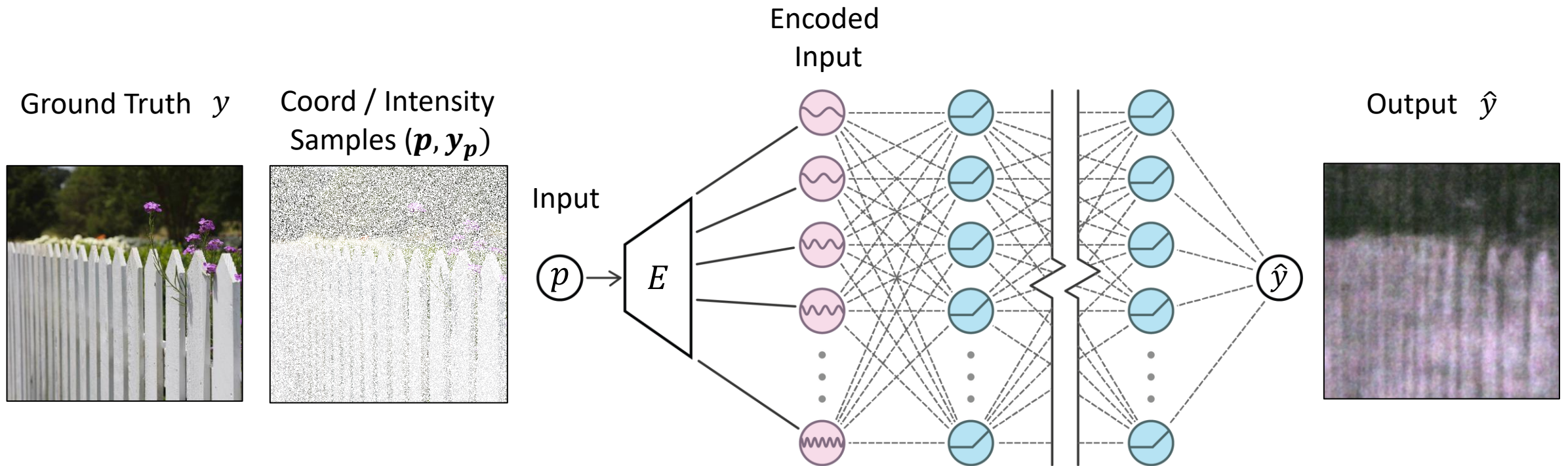
Positional Encodings



NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis, Mildenhall et al. 2020

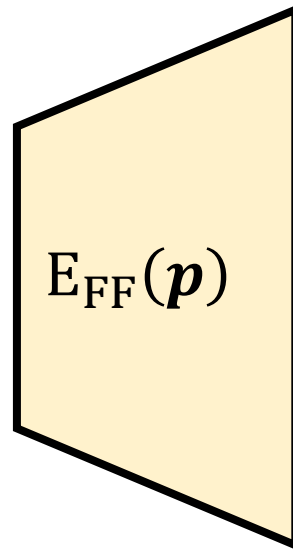
Implicit Neural Representations with Periodic Activation Functions, Sitzmann et al. 2020

Positional Encoding



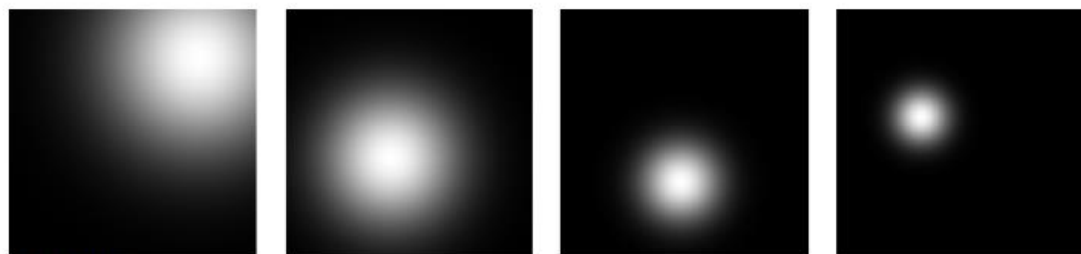
Positional Encoding

$$\mathbf{p} \in \mathbb{R}^d$$

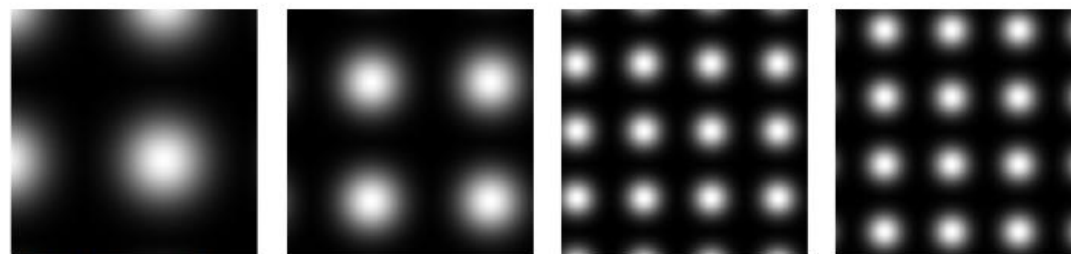


$$\begin{bmatrix} \cos(2\pi \mathbf{b}_1^\top \mathbf{p}) \\ \sin(2\pi \mathbf{b}_1^\top \mathbf{p}) \\ \dots \\ \dots \\ \cos(2\pi \mathbf{b}_n^\top \mathbf{p}) \\ \sin(2\pi \mathbf{b}_n^\top \mathbf{p}) \end{bmatrix}$$

Positional Encodings



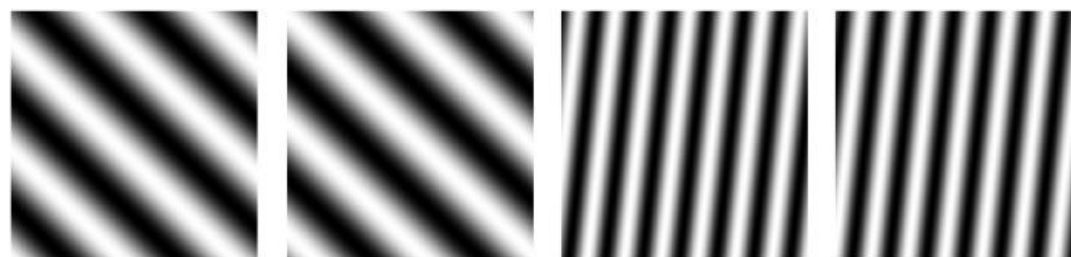
RBF



RBF grid



Regular Positional encoding



Fourier features

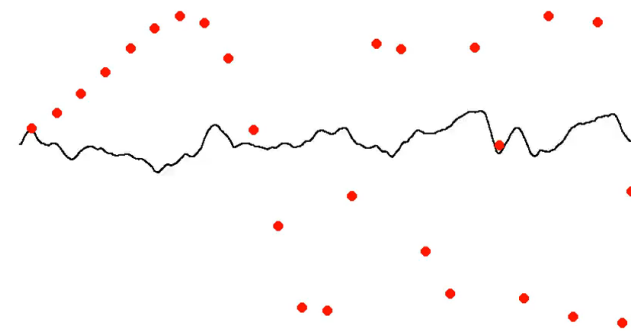
What's the problem here?



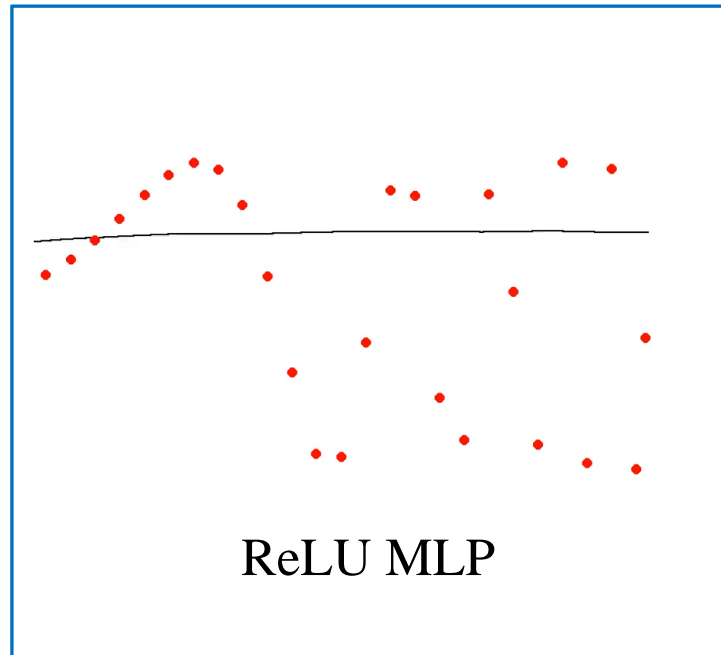
FFN $\sigma = 5$



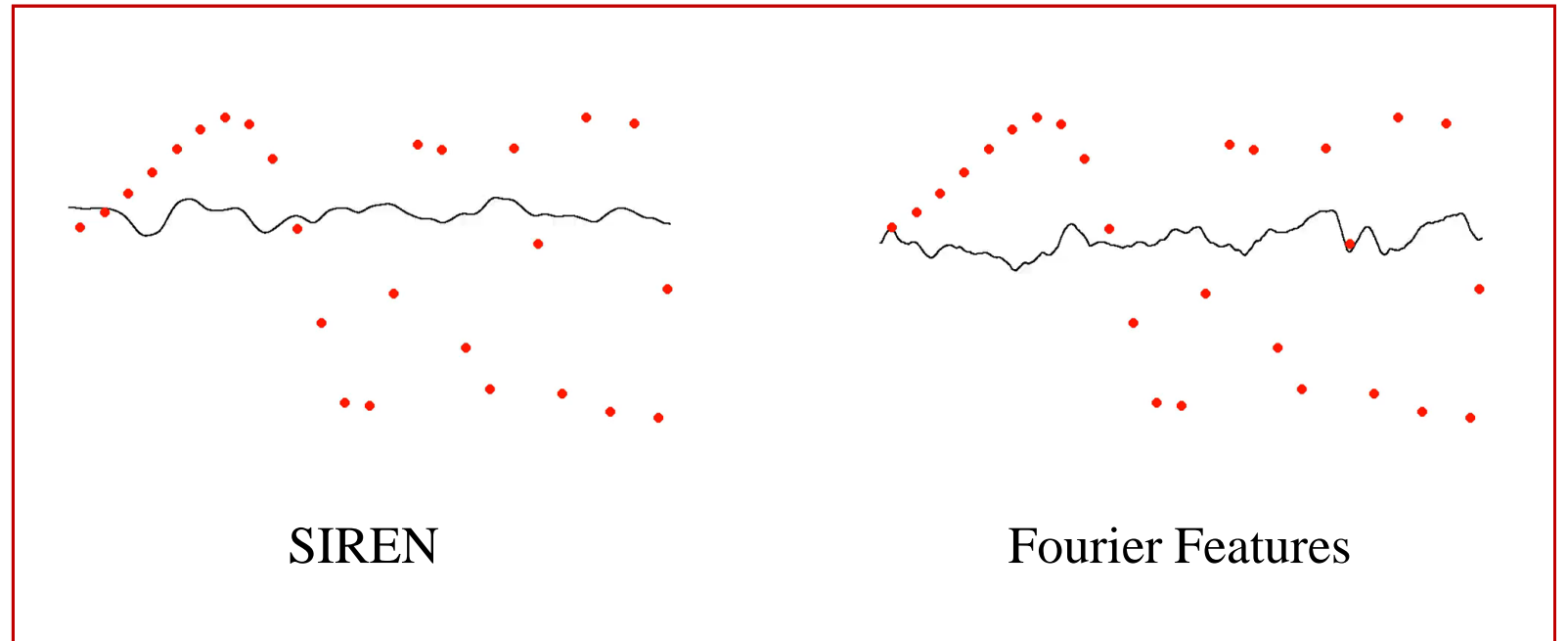
FFN $\sigma = 25$



ReLU MLP vs. Positional Encoding



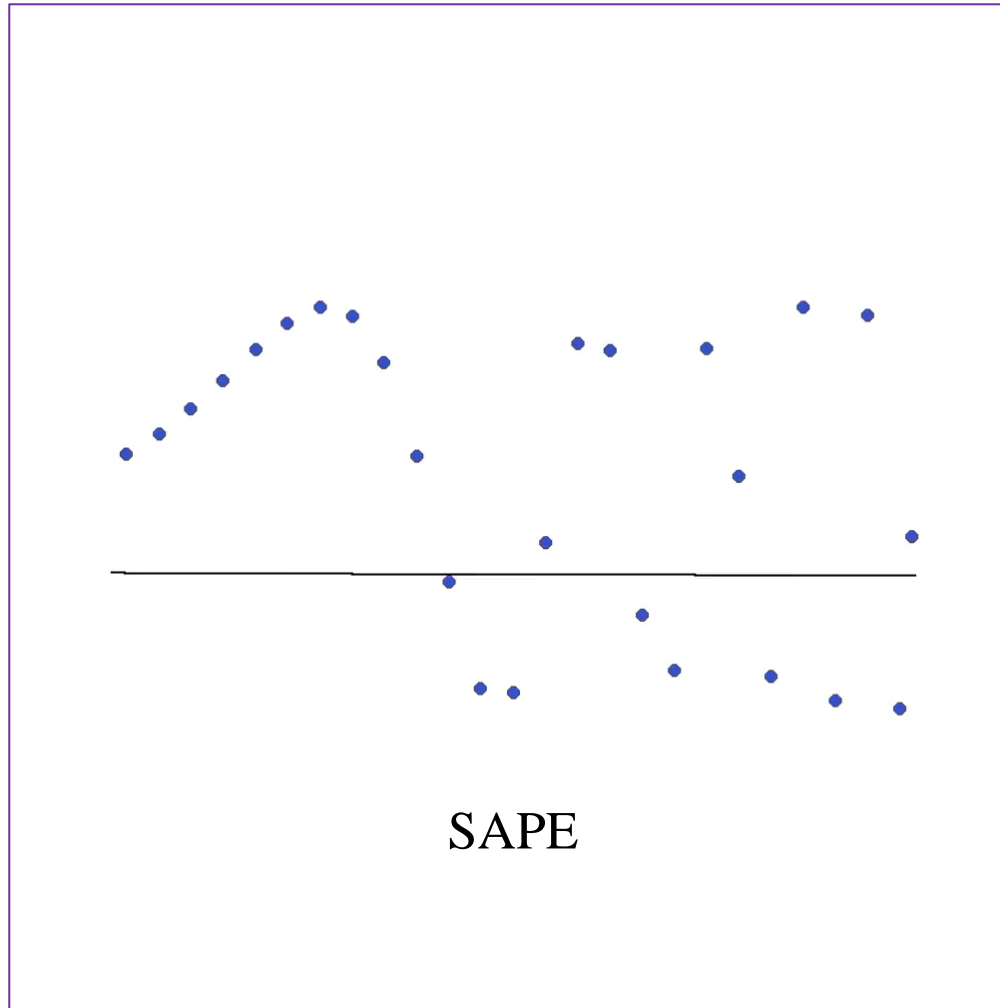
Spectral Bias
Low Frequencies
Globally Stable



Positional Encoding
High Frequencies
Fits Delicate Details

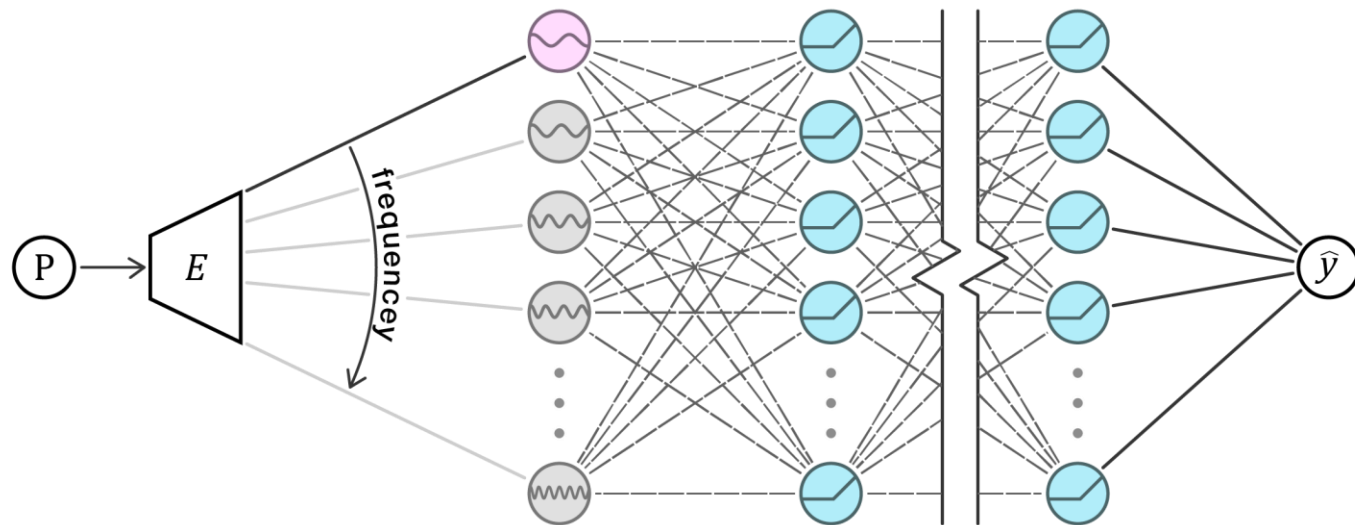
SAPE: The Best of Both Worlds

Spectral Bias
Low Frequencies
Globally Stable

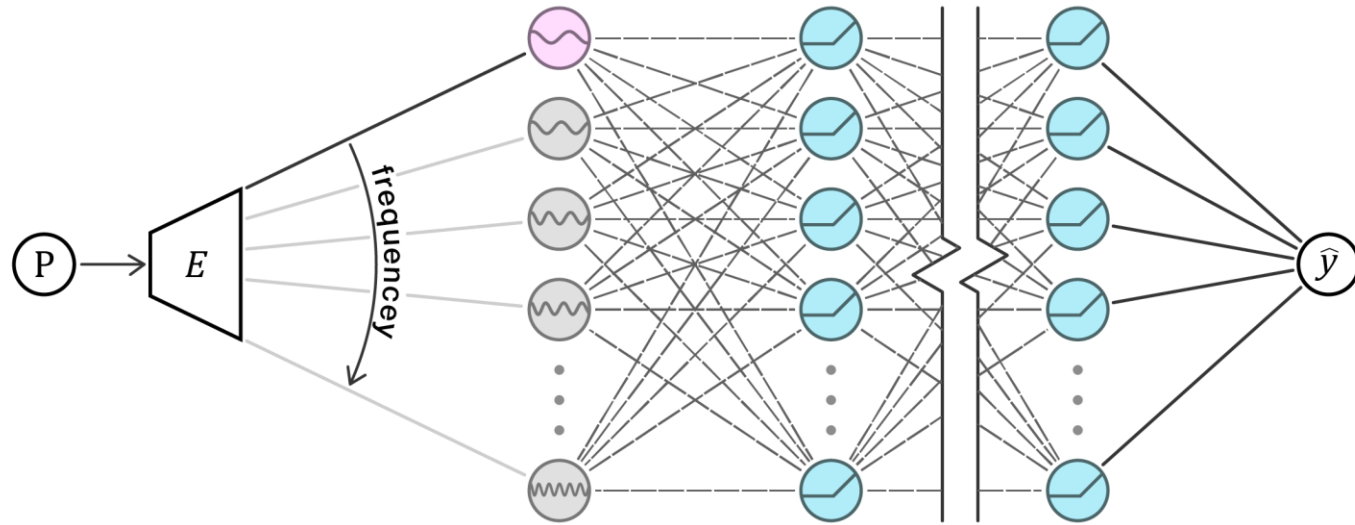


Positional Encoding
High Frequencies
Fits Delicate Details

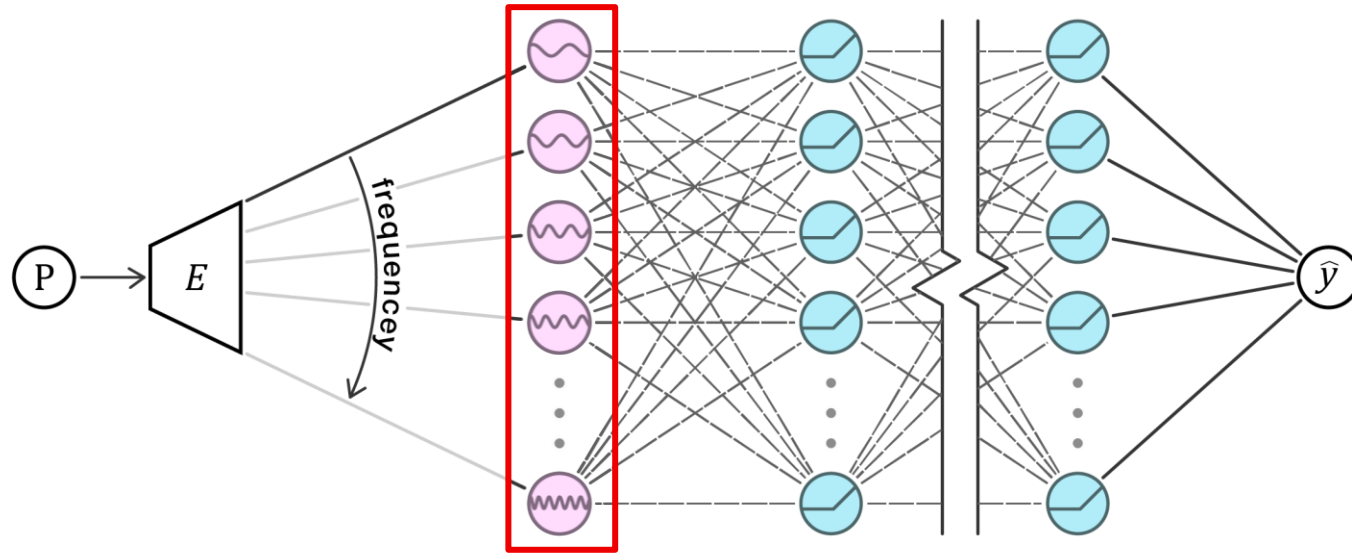
SAPE



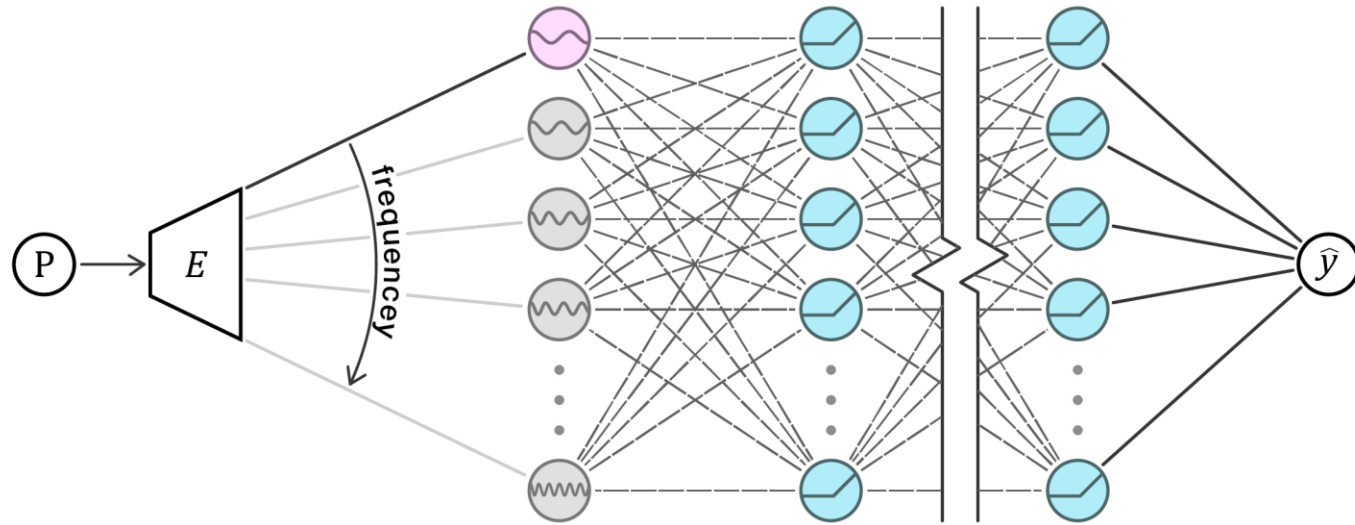
Progressive Encoding



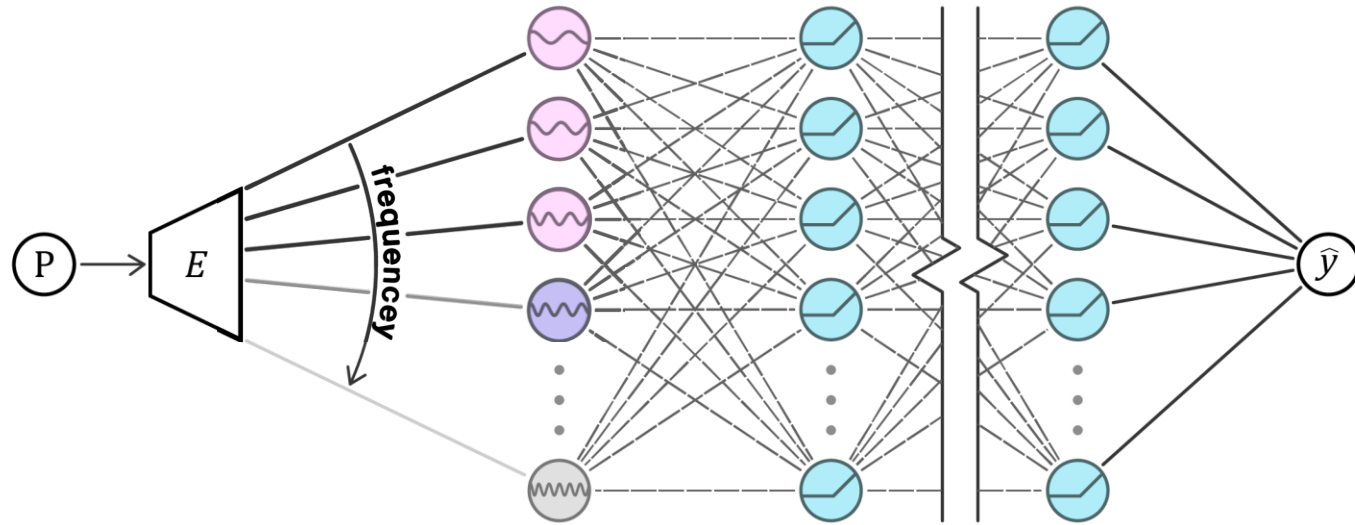
Progressive Encoding



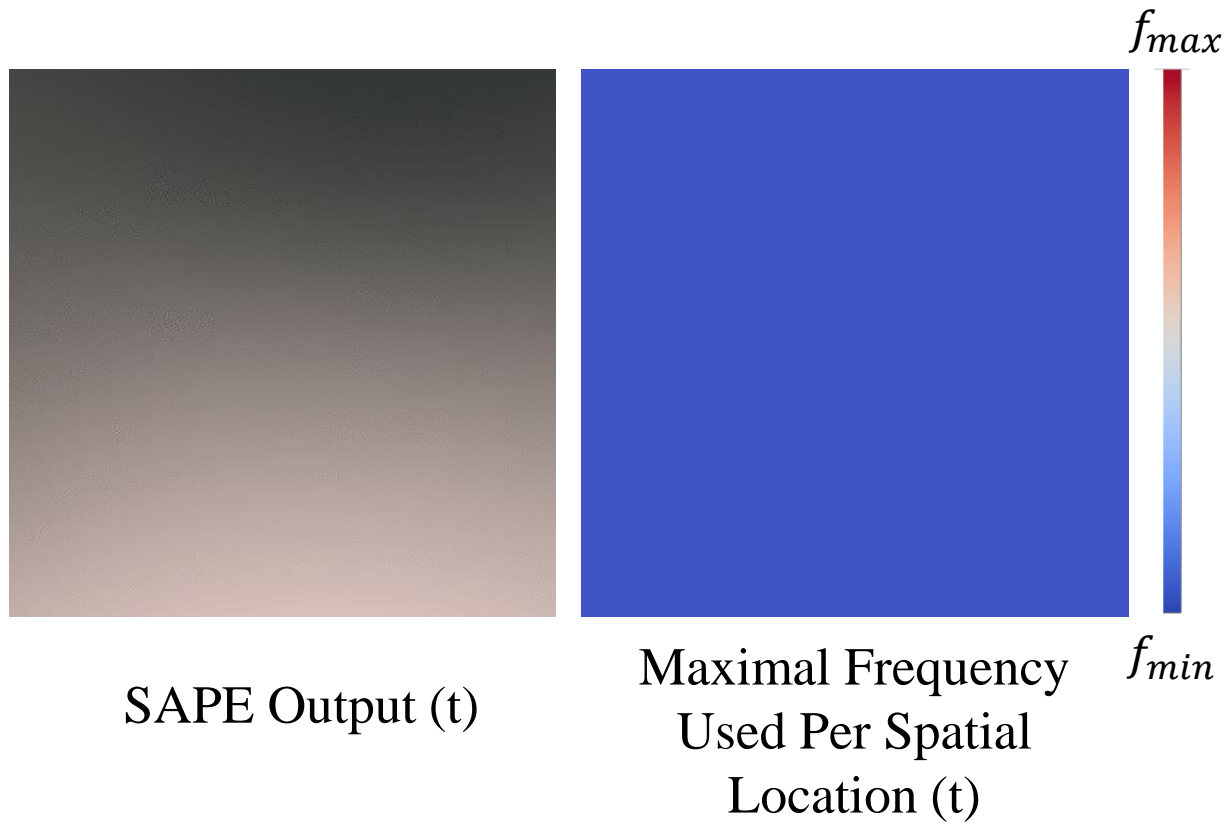
Progressive Encoding



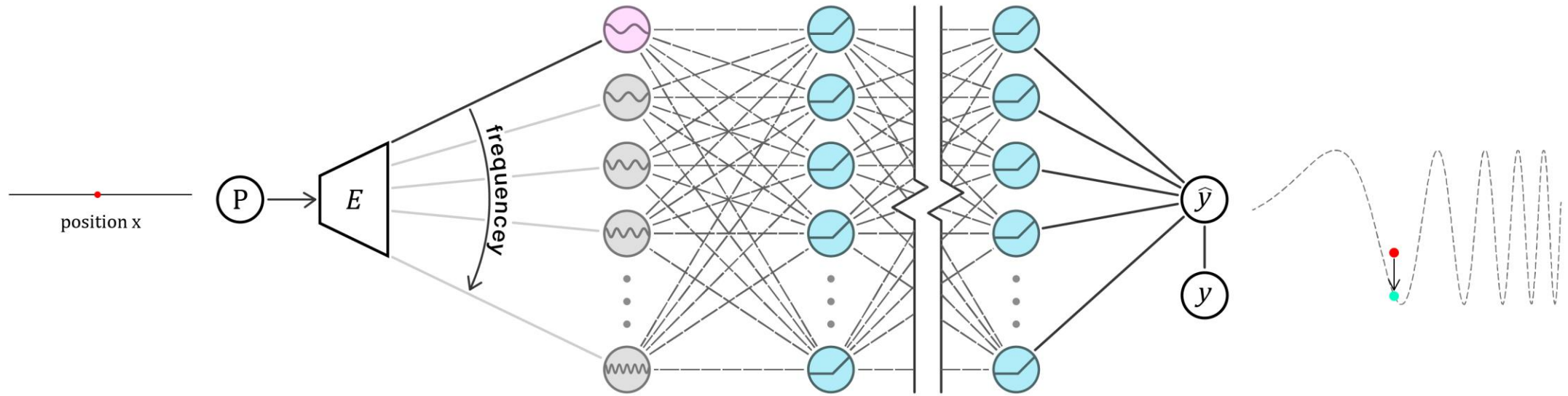
Progressive Encoding



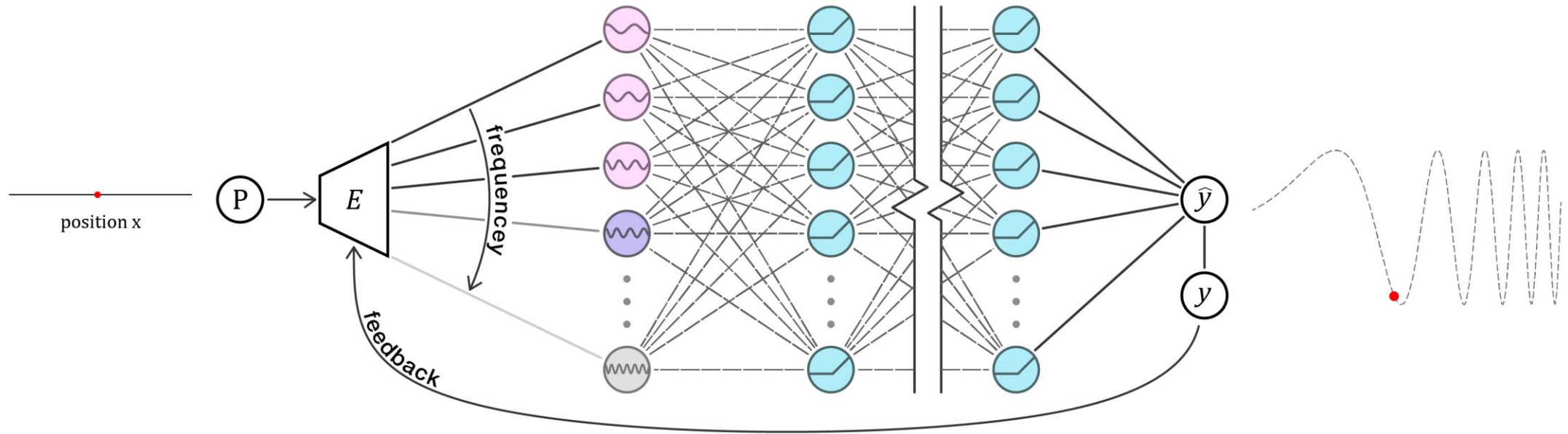
Spatially Adaptive



Spatially Adaptive



Spatially Adaptive

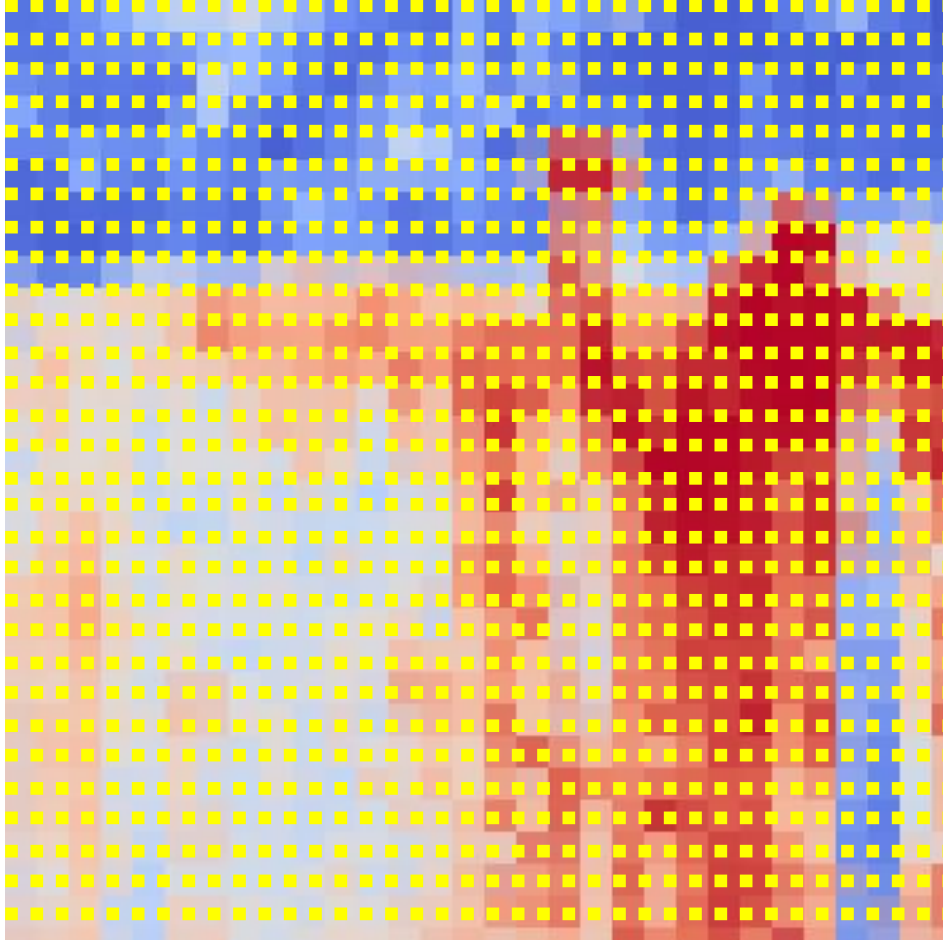
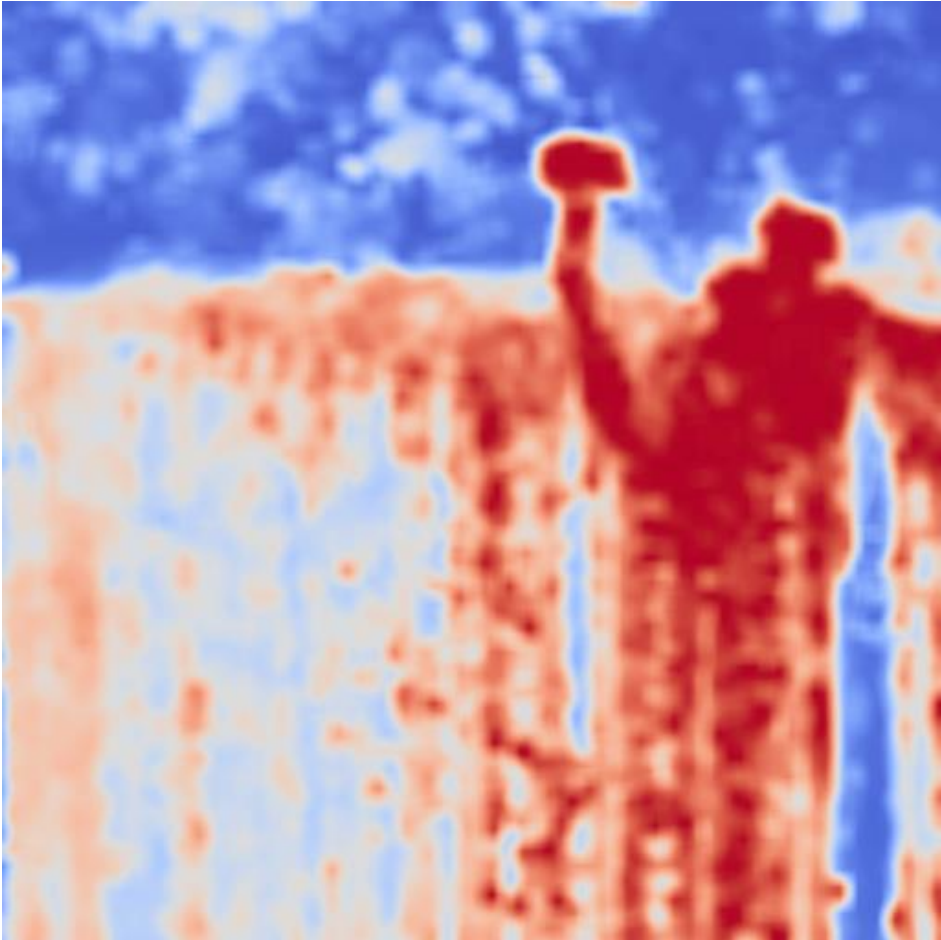


Spatially Adaptive

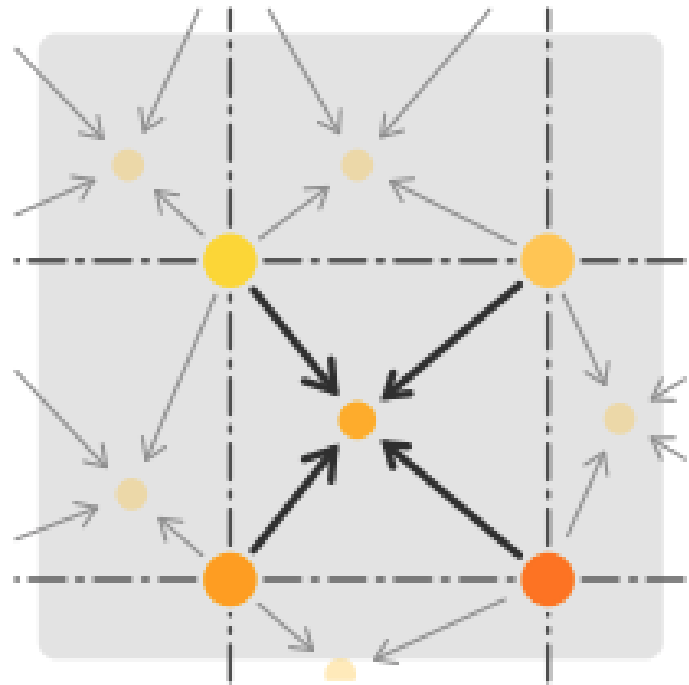


Figure 10: Non-spatial (left) vs. spatial encoding (middle). SAPE fit high frequencies while maintaining a smooth background using a different spatial encoding (right).

Adaptive Grid

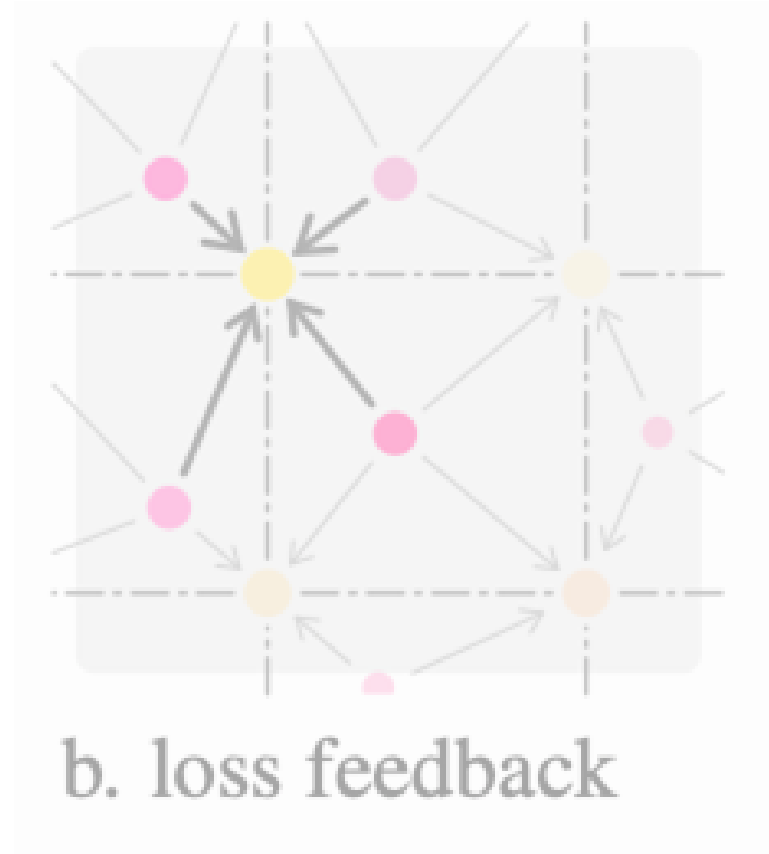


Adaptive Grid



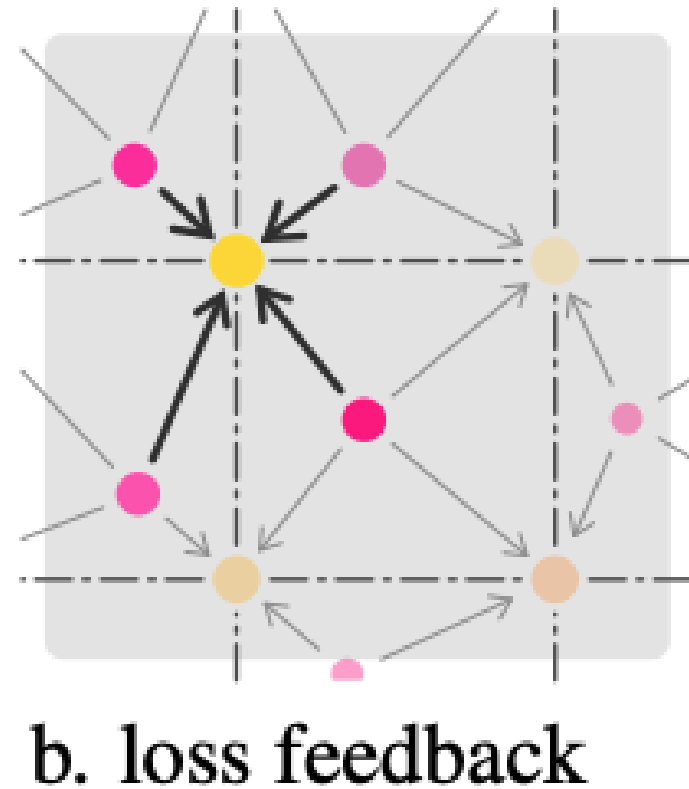
a. forward masking

$$\alpha(t, \mathbf{p}) = \sum_{\mathbf{u} \in \mathcal{N}_G(\mathbf{p})} w_{\mathbf{p}, \mathbf{u}} \alpha(t, \mathbf{u})$$



b. loss feedback

Adaptive Grid

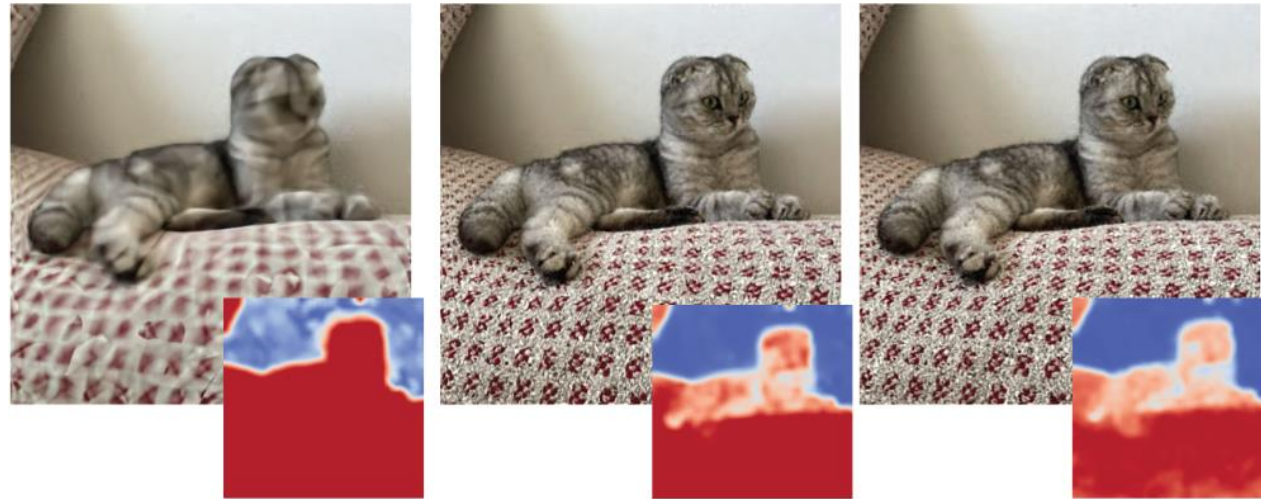


$$\mathcal{L}(t, \mathbf{u}) = \frac{1}{\sum_{\mathbf{p}} w_{\mathbf{p}, \mathbf{u}}} \sum_{\mathbf{p}} w_{\mathbf{p}, \mathbf{u}} \mathcal{L}(t, \mathbf{p}).$$

Results

Robustness to the Frequency Range

SAPE+FF



FF



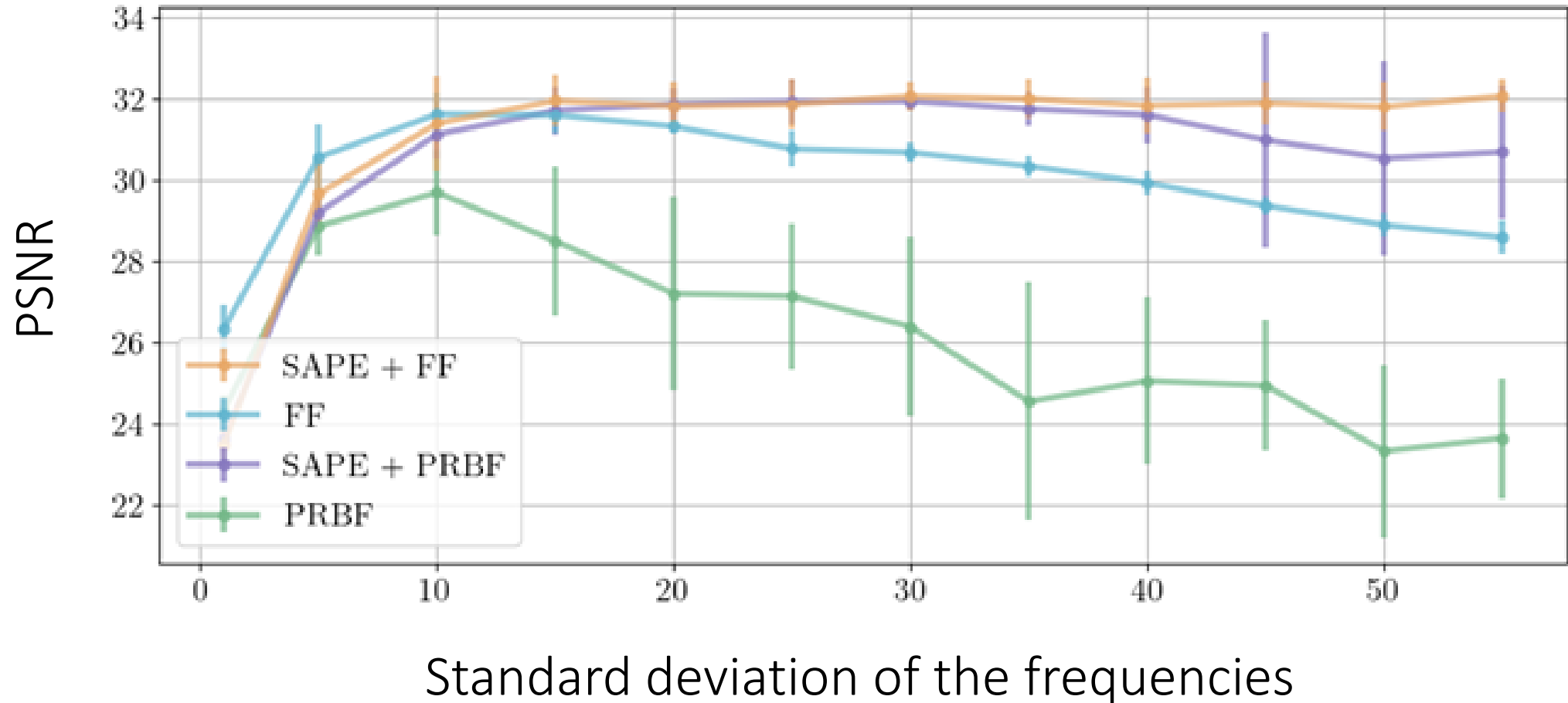
$\sigma = 1$

$\sigma = 20$

$\sigma = 40$



Robustness to the Frequency Range



Robustness to Sample Size

Sampled pixels



10%

20%

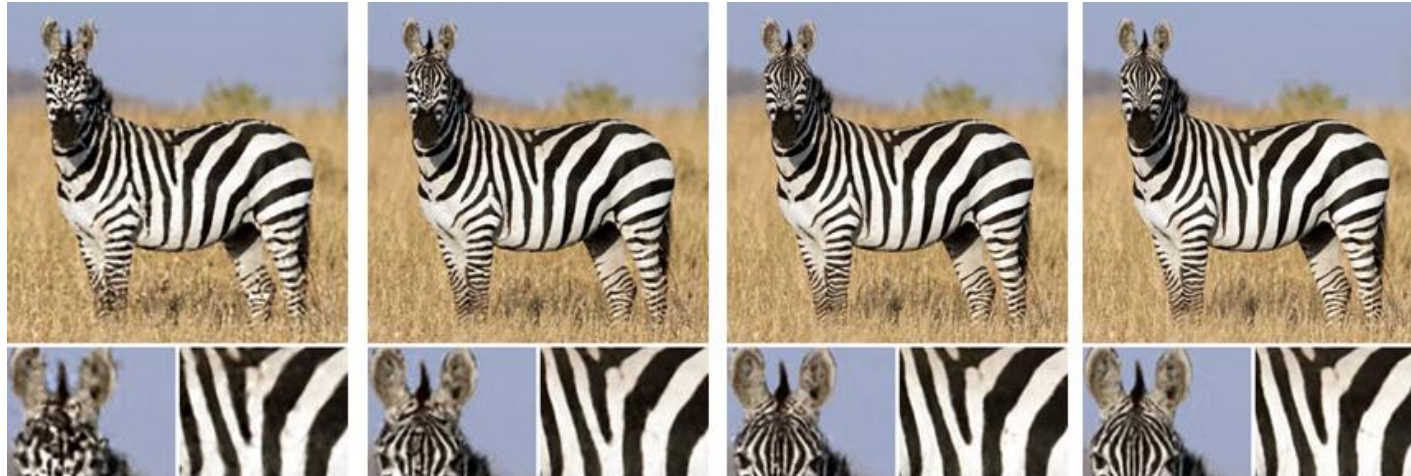
30%

40%

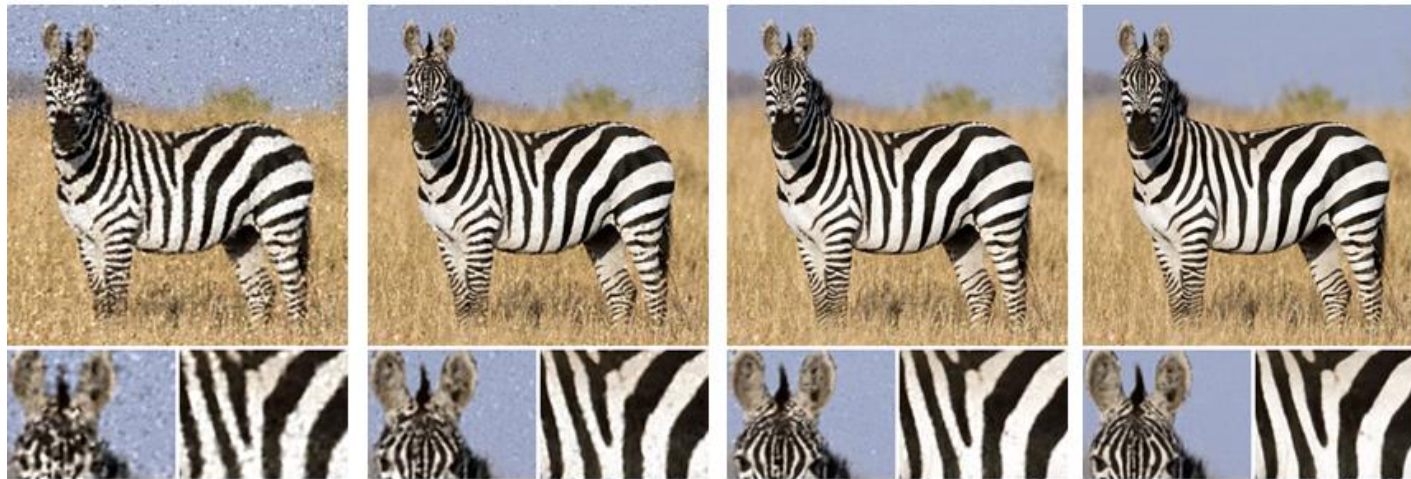
Sample size

Robustness to Sample Rate

SAPE+FF



FF



10%

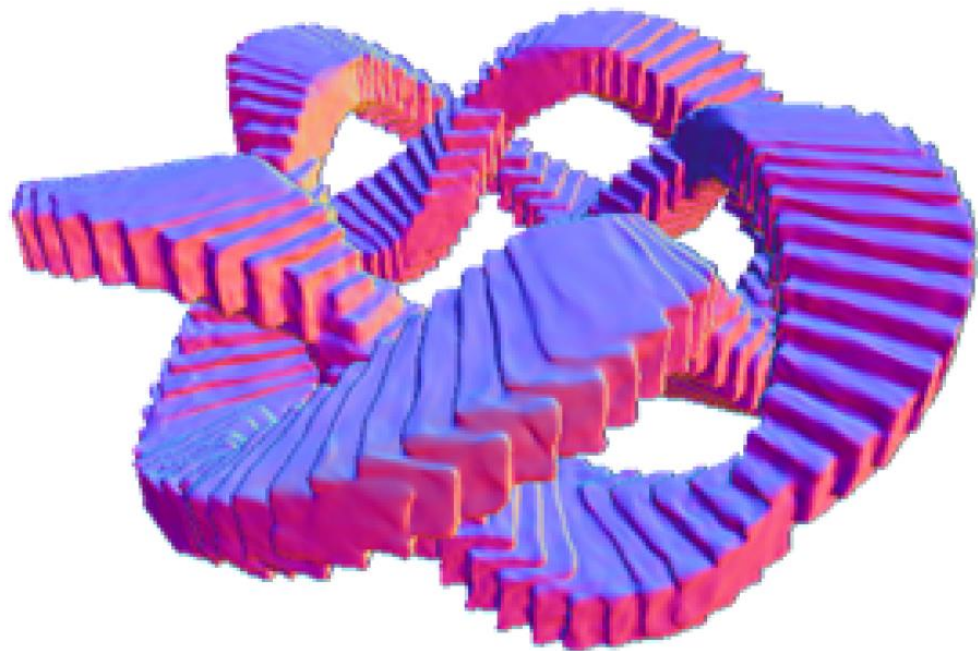
20%

30%

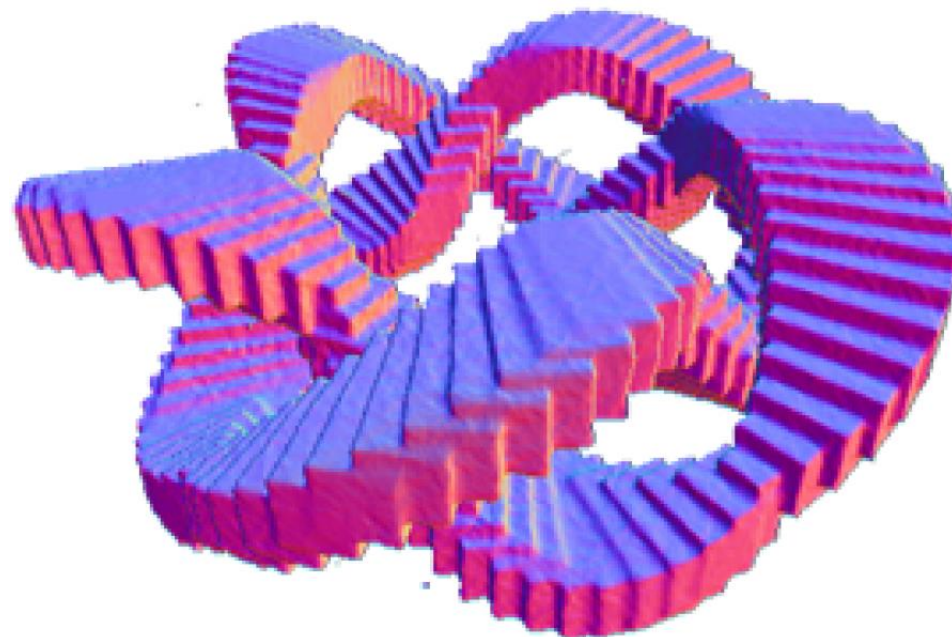
40%



Results: 3D Occupancy



Siren

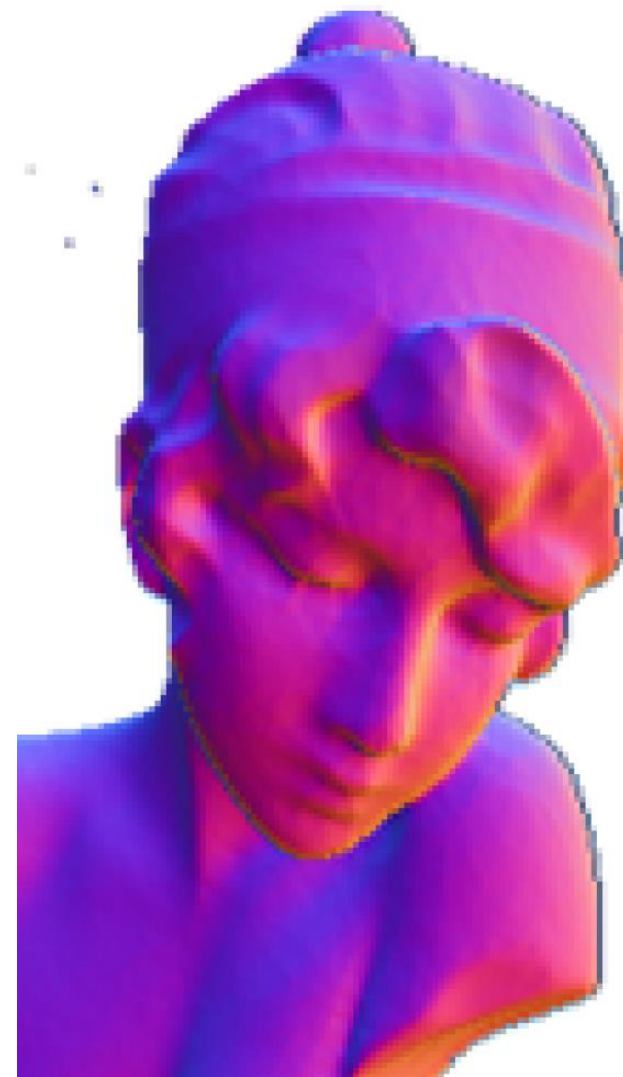


SAPE + FF

Results: 3D Occupancy

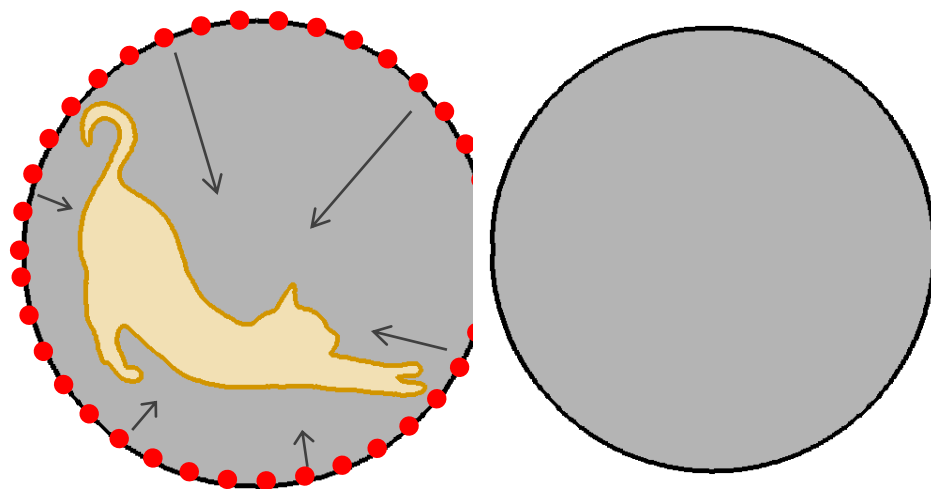


Siren

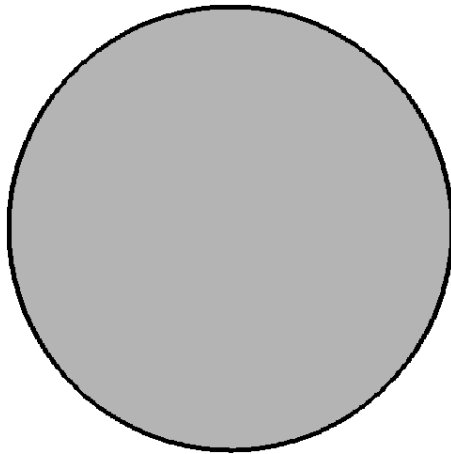


SAPE + FF

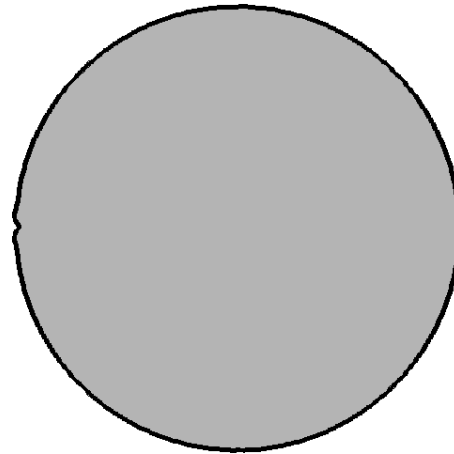
Results: 2D Silhouettes Deformation



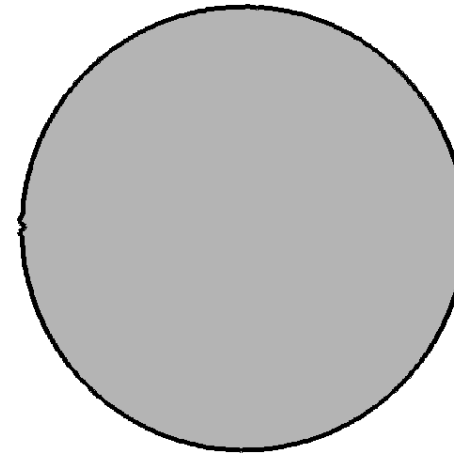
Results: 2D Silhouettes Deformation



FF $\sigma = 1$

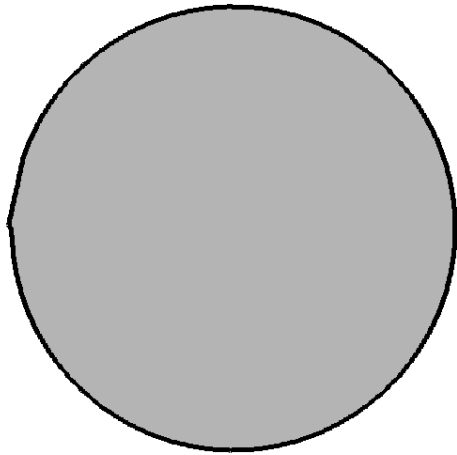


FF $\sigma = 3$

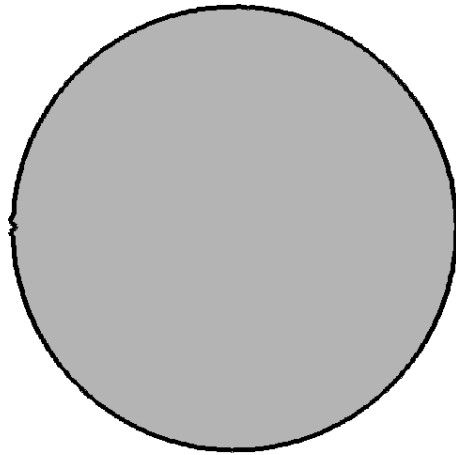


FF $\sigma = 5$

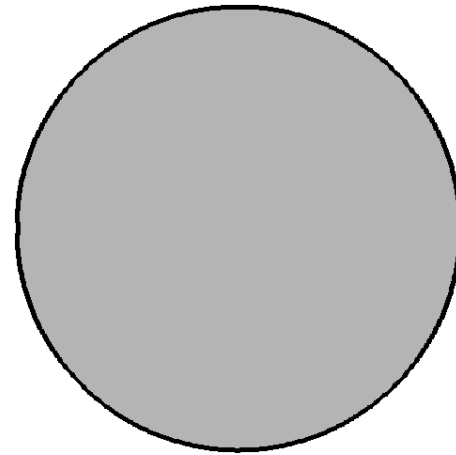
Results: 2D Silhouettes Deformation



No Encoding

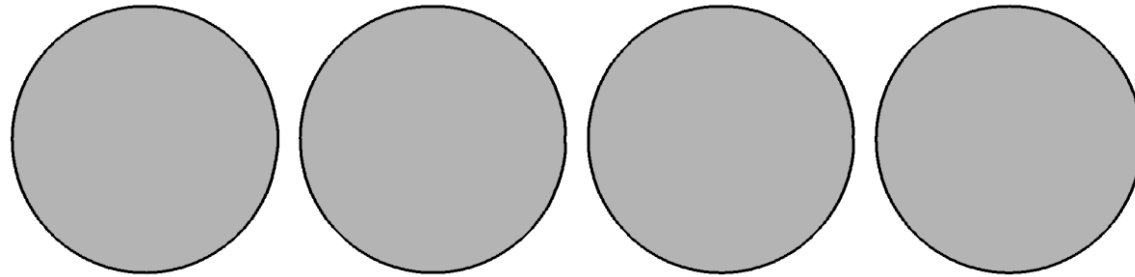


Fourier Features



SAPE

Thank you! 😊



Visit our project page: amirhertz.github.io/sape