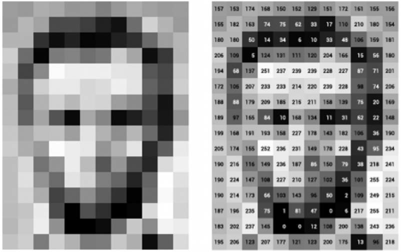
ref.bib

# Implicit neural representations

Amer ESSAKINE

May 2024- August 2024

# Representation continue



We transition from the discrete to the continuous. $\longrightarrow$ $f : \mathbb{R}^2 \rightarrow \mathbb{R}^3$

Given an input $x$, we aim to parameterize a continuous (and often differentiable) function $\phi$, such that there exists a dependency between $x$ (or a representation $a(x)$) with $\phi$ and possibly its derivatives:

$$\mathcal{C}(x, \psi, \nabla\psi, \ldots, \nabla^n\psi) = 0$$

The Implicit Neural Representation (INR) is constructed using an MLP (multilayer perceptron) with alternating linear layers and basic nonlinear activation functions.

# Biais for low frequencies

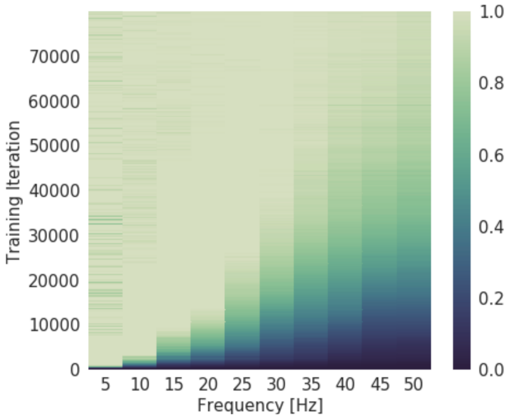



Figure: MLPs with ReLU tend to focus low frequencies

# INR

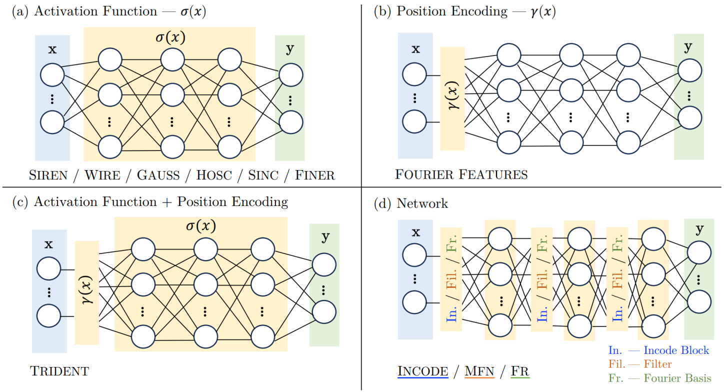



Figure: Various Architectures of Implicit Neural Representations

# Positional encoding

- **Basic** : $\gamma(x) = [\cos(2\pi x), \sin(2\pi x)]^T$ where $\sigma$ is the frequency hyperparameter.
- **Positional encoding** $\gamma(x) = [x, \dots, \cos(2\pi\sigma^{j/m}x), \sin(2\pi\sigma^{j/m}x), \dots]^T$, where $\sigma$ represents the frequency hyperparameter and $m$ the embedding size.
- **Random Fourier Features** $\gamma(x) = [\cos(2\pi Bx), \sin(2\pi Bx)]^T$, with $B$ being a random Gaussian matrix sampled from $\mathcal{N}(0, \sigma^2)$.

# Multiplicative Fourier features

**MFN**

The output of each layer is

$$y_i = (W_i y_{i-1} + b_i) \circ g(x; \theta^{i+1})$$

Where

$$g(x; \theta^i) = \exp(-\gamma^i \|x - \mu^i\|_2^2) \sin(\omega^i x + \phi^i)$$

is the Gabor wavelet

# Activation function

- **SIREN** The periodic activity function $\sigma(x) = sin(\omega_0 x)$
- **Gauss** The sinus cardinal $\sigma(x) = \exp(-\sigma x^2)$
- **WIRE** The Gabor wavelet : $\sigma(x) = \exp(-\sigma x^2 + i\omega_0 x)$
- **SINC** The sinus cardinal $\sigma(x) = sinc(\omega_0 x)$
- **FINER** The activation function $\sigma(x) = sin(\omega_0 |x+1| x)$
- **INCODE** The activation function $\sigma(x) = asin(b\omega_0 x + c) + d$ where $a$,$b$,$c$ and $d$ are leanrt by an MLP

# Other methods

**Reparametrized learning**

The output of each layer is :

$$y_i = \sigma(\Lambda_i B_i y_{i-1} + b_i)$$

here $\Lambda_i$ is learnable and

$$B_i^{k,l} = cos(w_k z_l + \phi_k)$$

# Wide range of applications

**1D applications**

- Audio reconstruction
  Metrics : L2 loss

**2D Applications**

- Image denoising
- CT reconstruction
  Metrics : PSNR and SSIM

**3D applications**

- 3D Occupancy Reconstruction
  Metrics : PSNR, IoU

# Audio reconstruction

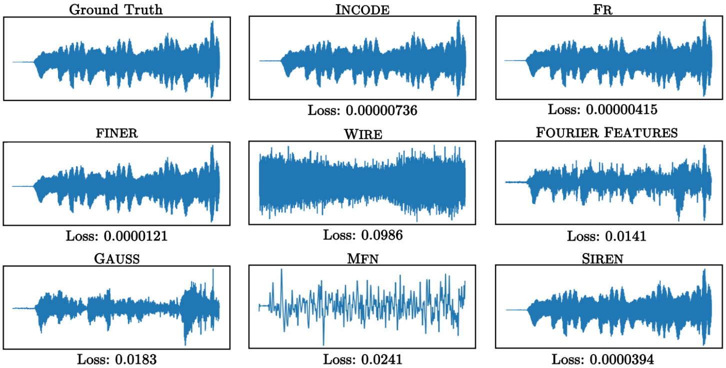



Figure: Reconstruction of an audio

# Radon Transform

The Radon transform of $f$ is defined by:

$$R_f(\theta, t) = \int_{-\infty}^{\infty} f(x \cos\theta + y \sin\theta, -x \sin\theta + y \cos\theta)\, dt$$

It represents the projection of this density onto a line

$$L(\theta, t) = \{x \in \mathbb{R}^2 \mid x_1 \cos(\theta) + x_2 \sin(\theta) = t\}$$

.

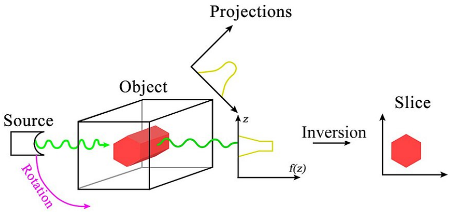
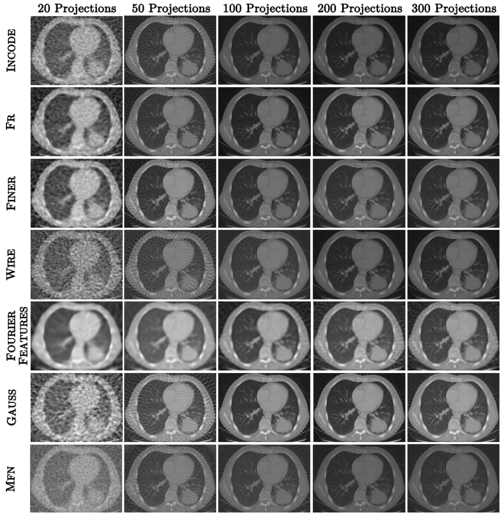



Figure: CT Reconstruction - Wikipedia

# CT reconstruction



# CT reconstruction

| Method | 20 projections | | 50 projections | | 100 projections | |
|---|---|---|---|---|---|---|
| | PSNR↑ | SSIM↑ | PSNR↑ | SSIM↑ | PSNR↑ | SSIM↑ |
| INCODE (Kazerouni et al., 2024) | 24.38 | 0.651 | 27.33 | 0.762 | 31.48 | 0.890 |
| FR (Shi et al., 2024) | 25.58 | 0.738 | 29.14 | 0.852 | 31.18 | 0.917 |
| FINER (Liu et al., 2024) | 25.53 | 0.731 | 28.20 | 0.841 | 30.92 | 0.888 |
| WIRE (Saragadam et al., 2023) | 20.73 | 0.417 | 25.01 | 0.651 | 28.83 | 0.826 |
| FOURIER FEATURES (Tancik et al., 2020) | 25.64 | 0.780 | 26.44 | 0.803 | 26.74 | 0.802 |
| GAUSS (Ramasinghe & Lucey, 2022) | 22.21 | 0.548 | 26.44 | 0.752 | 27.80 | 0.764 |
| MFN (Fathony et al., 2020) | 21.10 | 0.402 | 22.70 | 0.487 | 25.30 | 0.643 |
| SIREN (Sitzmann et al., 2020) | 20.85 | 0.421 | 24.42 | 0.607 | 29.61 | 0.842 |

| Method | 150 projections | | 200 projections | | 300 projections | |
|---|---|---|---|---|---|---|
| | PSNR↑ | SSIM↑ | PSNR↑ | SSIM↑ | PSNR↑ | SSIM↑ |
| INCODE (Kazerouni et al., 2024) | 33.94 | 0.939 | 34.29 | 0.946 | 34.76 | 0.953 |
| FR (Shi et al., 2024) | 30.25 | 0.907 | 30.49 | 0.910 | 30.40 | 0.910 |
| FINER (Liu et al., 2024) | 31.84 | 0.914 | 32.13 | 0.922 | 32.24 | 0.927 |
| WIRE (Saragadam et al., 2023) | 30.54 | 0.891 | 31.88 | 0.916 | 30.95 | 0.902 |
| FOURIER FEATURES (Tancik et al., 2020) | 26.98 | 0.804 | 26.94 | 0.803 | 26.94 | 0.794 |
| GAUSS (Ramasinghe & Lucey, 2022) | 27.85 | 0.766 | 27.89 | 0.773 | 27.90 | 0.848 |
| MFN (Fathony et al., 2020) | 28.29 | 0.793 | 30.50 | 0.868 | 33.63 | 0.935 |
| SIREN (Sitzmann et al., 2020) | 31.58 | 0.907 | 32.02 | 0.918 | 32.71 | 0.928 |

Figure: Quantitative results

# Image denoising

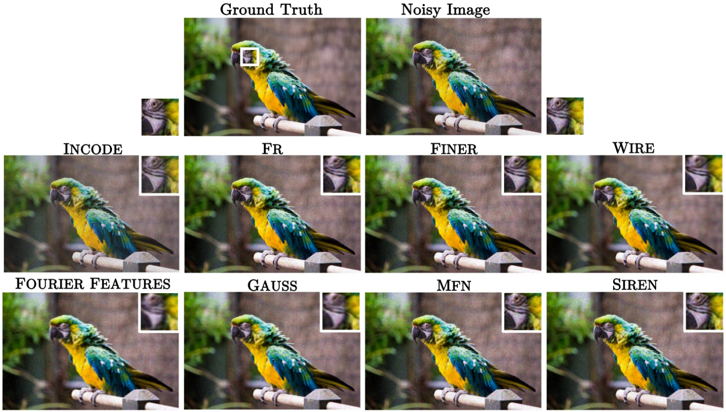



Figure: Image denoising

# Image denoising

| Method | Incode | Fr | Finer | Wire | Fourier Features | Gauss | Mfn | Siren |
|---|---|---|---|---|---|---|---|---|
| **PSNR** ↑ | 29.63 | 29.47 | 29.05 | 28.74 | 27.16 | 28.09 | 28.22 | 28.60 |
| **Time(s)** ↓ | 2370 | 752 | 643 | 1485 | 403 | 709 | 1280 | 482 |

Figure: Image denoising : Quantitative results

# Image super resolution

| Method | 2× PSNR↑ | 2× SSIM↑ | 2× LPIPS ↓ | 4× PSNR↑ | 4× SSIM↑ | 4× LPIPS ↓ |
|---|---|---|---|---|---|---|
| INCODE (Kazerouni et al., 2024) | 29.56 | 0.896 | 0.176 | 27.43 | 0.816 | 0.422 |
| FR (Shi et al., 2024) | 29.10 | 0.879 | 0.243 | 27.49 | 0.822 | 0.371 |
| FINER (Liu et al., 2024) | 29.50 | 0.892 | 0.191 | 27.44 | 0.818 | 0.395 |
| WIRE (Saragadam et al., 2023) | 28.91 | 0.874 | 0.252 | 25.93 | 0.754 | 0.447 |
| FOURIER FEATURES (Tancik et al., 2020) | 26.31 | 0.767 | 0.428 | 25.73 | 0.733 | 0.473 |
| GAUSS (Ramasinghe & Lucey, 2022) | 28.08 | 0.851 | 0.324 | 24.10 | 0.681 | 0.619 |
| MFN (Fathony et al., 2020) | 29.28 | 0.890 | 0.203 | 24.99 | 0.716 | 0.610 |
| SIREN (Sitzmann et al., 2020) | 29.00 | 0.877 | 0.241 | 27.27 | 0.811 | 0.409 |

| Method | 8× PSNR↑ | 8× SSIM↑ | 8× LPIPS ↓ | 16× PSNR↑ | 16× SSIM↑ | 16× LPIPS ↓ |
|---|---|---|---|---|---|---|
| INCODE (Kazerouni et al., 2024) | 25.43 | 0.731 | 0.597 | 22.91 | 0.638 | 0.715 |
| FR (Shi et al., 2024) | 23.75 | 0.662 | 0.643 | 23.40 | 0.682 | 0.637 |
| FINER (Liu et al., 2024) | 25.69 | 0.743 | 0.544 | 23.39 | 0.667 | 0.665 |
| WIRE (Saragadam et al., 2023) | 21.72 | 0.558 | 0.703 | 18.06 | 0.422 | 0.773 |
| FOURIER FEATURES (Tancik et al., 2020) | 22.31 | 0.549 | 0.473 | 20.65 | 0.518 | 0.702 |
| GAUSS (Ramasinghe & Lucey, 2022) | 20.06 | 0.464 | 0.872 | 17.14 | 0.349 | 0.868 |
| MFN (Fathony et al., 2020) | 19.49 | 0.411 | 0.750 | 16.61 | 0.290 | 0.934 |
| SIREN (Sitzmann et al., 2020) | 24.66 | 0.703 | 0.583 | 19.09 | 0.509 | 0.780 |

Figure: Caption

# Image super resolution

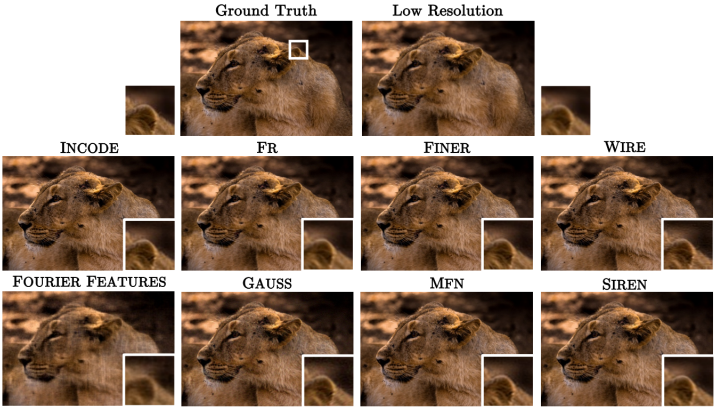



Figure: Super resolution 2x

# 3D shape representation

We attempt to learn the signed distance function of a surface $S$ defined on $\mathbb{R}^3$ with values in $\mathbb{R}$ by:

$$\begin{cases} f(x) = -d(x, \delta S) & \text{si } x \in S, \\ f(x) = d(x, \delta S) & \text{sinon} \end{cases}$$

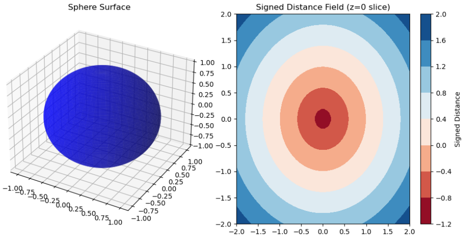
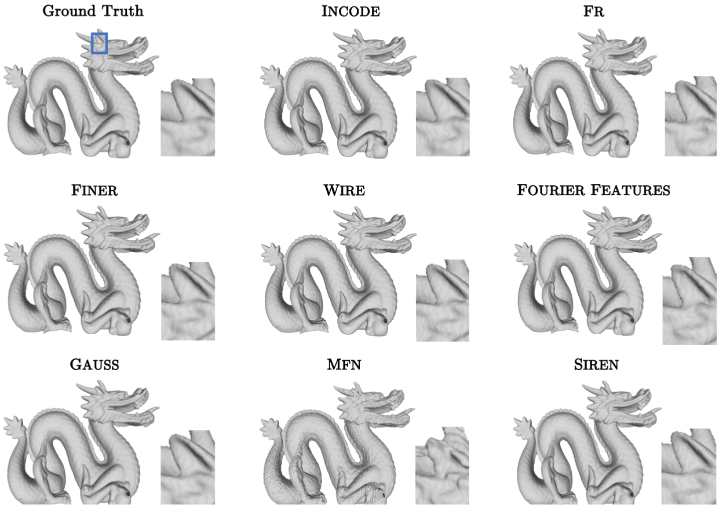



Figure: SDF of a sphere in $R^3$

# 3D shapre representation



# 3D shape representation

| Method | Incode | Fr | Finer | Wire | Fourier Features | Gauss | Mfn | Siren |
|---|---|---|---|---|---|---|---|---|
| **IoU** ↑ | 0.99564 | 0.99136 | 0.99628 | 0.99454 | 0.99424 | 0.99510 | 0.97540 | 0.99552 |

Figure: 3D occupancy metrics

# Sampling theory

A continuous band-limited signal $x(t)$ can be reconstructed from its samples using the interpolation formula:

$$x(t) = \sum_{n \in \mathbb{Z}} x(nT) \cdot \operatorname{sinc}\left(\frac{x}{T} - k\right), \tag{1}$$

# Sampling problem is harder than INRs

**Theorem**

*(Hemanth Saratchandran et AL[1]) If a signal can be reconstructed using the sampling formula with an arbitrary function $\phi$, then a two-layer neural network can approximate the signal as closely as desired.*

# SIREN and sampling

For any trigonometrical polynomial$\phi$, $V(\phi) = \{\phi(x-k) | k \in \mathbb{Z}\}$ is a vectorial space of finite dimesnsion. As such Siren cannot express all functions

# SIREN and sampling

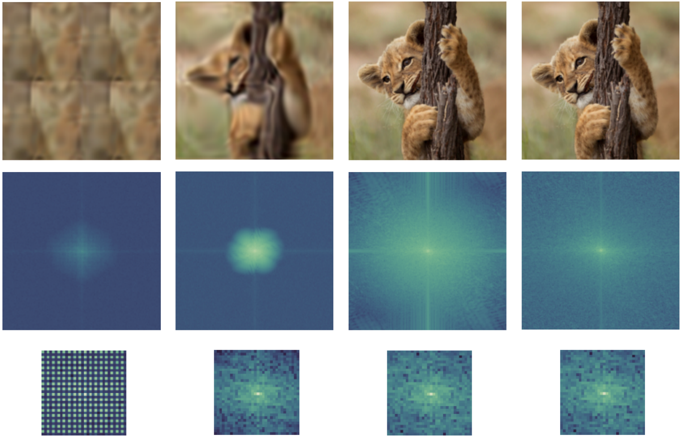


Figure: Reconstruction of an image with SIREN, with the spectrum of each image in

# Two Criterias

- the set of function$(x \to \phi(x-k) | k \in \mathbb{Z})$ should be a Reisz sequence :

$$\forall (c(k))_k \in l_2, \quad A \cdot \|(c(k))_k\|_{l_2}^2 \leq \left\| \sum_{k \in \mathbb{Z}} c(k)\phi_k \right\|^2 \leq B \cdot \|(c(k))_k\|_{l_2}^2$$

- Partition of unity:

$$\forall x \in E, \sum_{k \in \mathbb{Z}} \phi(x-k) = 1$$

# Walsh functions

For $k \in \mathbb{N}$, the $k$-th Rademacher function $r_k : [0,1] \to \{-1, 1\}$ is defined as:

$$r_k(t) = \text{sgn}(\sin(2^{k+1}\pi t)),$$

where $\text{sgn}(x)$ is the sign function.
Given $n \in \mathbb{N}$, let $n$ be expressed in binary as $n = \sum_{j=0}^{+\infty} k_j 2^j$, where $k_j \in \{0, 1\}$ are the binary coefficients of $n$. The $n$-th Walsh function $W_n(t)$ is then defined as:

$$wal_n(t) = \prod_{j=0}^{+\infty} r_j(t)^{k_j},$$

# Walsh Functions

The inverse Fourier transform of the $n$-th Walsh function is:

$$W_n(\omega) = (-j)^{\alpha} \left[ \prod_{k=1}^{m} \cos\left( \frac{\omega}{2^{k+1}} - \frac{g_k}{2} \right) \right] \text{sinc}\left( \frac{\omega}{2^M} \right)$$

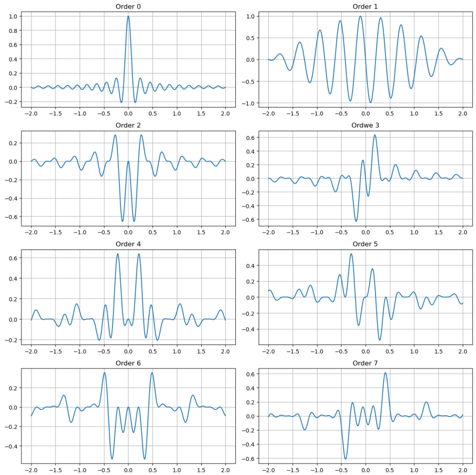



Figure: Graph for the first 8 functions

# Walsh Functions

**Theorem**

*It can be proven that if $n$ is even, then $W_n$ satisfies the two sampling conditions.*

# WIREN

We define the Walsh4 (Walsh Implicit Representation Network) as a Multi-Layer Perceptron (MLP) that uses the fourth-order Walsh function as its activation function, given by:

$$\sigma(x) = -\cos\left(\frac{\omega_0 x}{2}\right)\sin\left(\frac{\omega_0 x}{4}\right)\sin\left(\frac{\omega_0 x}{8}\right)\operatorname{sinc}\left(\frac{\sigma_0 x}{8}\right)$$

We also consider Walsh6, where we chose as an activation function, the Walsh function of order 6 given by :

$$\sigma(x) = -\sin\left(\frac{\omega_0 x}{2}\right)\cos\left(\frac{\omega_0 x}{4}\right)\sin\left(\frac{\omega_0 x}{8}\right)\operatorname{sinc}\left(\frac{\sigma_0 x}{8}\right)$$

Here, $\sigma_0$ and $\omega_0$ are hyperparameters that can be tuned for each experiment, with default values of $\omega_0 = 30$ and $\sigma_0 = 10$.

# CT reconstruction : 100 projection

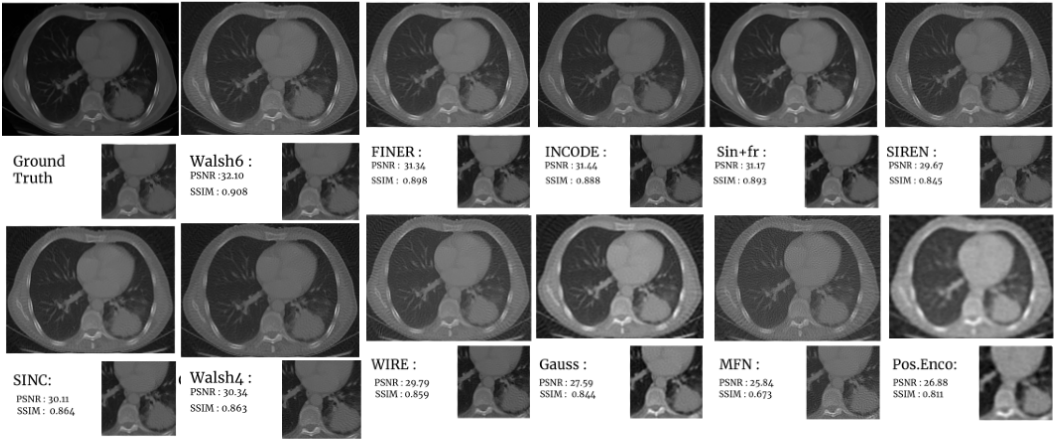



Figure: CT reconstruction task

# Rate of convergence

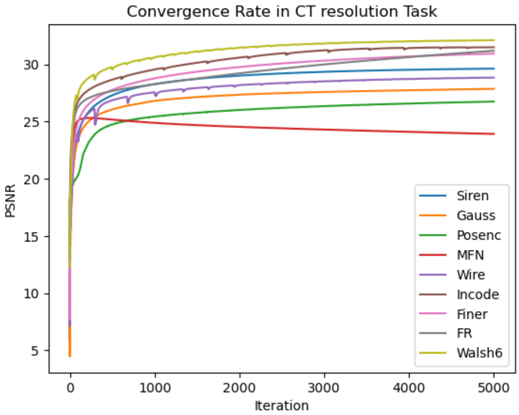



Figure: Rate of convergence for CT measurement

# Image denoising

# Image denoising

| Method | Incode | Re+sin | FINER | Walsh6 | Walsh4 | MFN |
|---|---|---|---|---|---|---|
| PSNR ↑ | 29.63 | 29.47 | 29.05 | 28.79 | 28.56 | 28.22 |

Table: Image denoising metrics for Incode to MFN methods

| Method | SIREN | Gauss | WIRE | PosEnc |
|---|---|---|---|---|
| PSNR ↑ | 28.60 | 28.09 | 28.74 | 27.16 |

Table: Image denoising metrics for SIREN to PosEnc methods

# Audio reconstruction

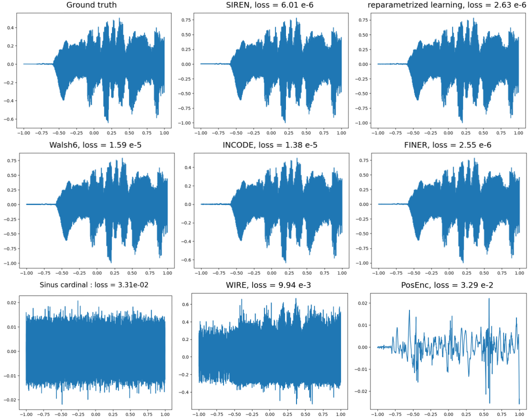



Figure: Audio reconstruction

# Audio reconstruction

| Method | Re+sin | FINER | SIREN | INCODE |
|---|---|---|---|---|
| PSNR ↑ | 2.63e-6 | 2.55e-6 | 6.06e-6 | 1.38e-5 |

| Method | Walsh6 | WIRE | PosEnc | Gauss | Sinc |
|---|---|---|---|---|---|
| PSNR ↑ | 1.59e-5 | 9.94e-3 | 3.29e-2 | 2.88e-2 | 3.31e-2 |

Table: Audio reconstruction metrics

# Conclusion

Conclusion