Convolutional Implicit Neural Representation of pathology whole-slide images

DongEon Lee[1[0000−0002−0189−0231]], Chunsu Park[2[0000−0002−9640−7619]], SeonYeong Lee[1[0009−0006−3682−2454]], SiYeoul Lee[1[0009−0000−1817−6037]], and MinWoo Kim[2,3[0000−0001−7547−2596]⋆]

[1] Department of Information Convergence Engineering, College of Information and Biomedical Convergence Engineering, Pusan National University, Yangsan, Korea
[2] School of Biomedical Convergence Engineering, College of Information and Biomedical Engineering, Pusan National University, Yangsan, Korea
[3] Center for Artificial Intelligence Research, Pusan National University, Busan, Korea
mkim180@pusan.ac.kr

Abstract. This study explored the application of implicit neural representations (INRs) to enhance digital histopathological imaging. Traditional imaging methods rely on discretizing the image space into grids, managed through a pyramid file structure to accommodate the large size of whole slide images (WSIs); however, the continuous mapping capability of INRs, utilizing a multi-layer perceptron (MLP) to encode images directly from coordinates, presents a transformative approach. This method promises to streamline WSI management by eliminating the need for down-sampled versions, allowing instantaneous access to any image region at the desired magnification, thereby optimizing memory usage and reducing data storage requirements. Despite their potential, INRs face challenges in accurately representing high spatial frequency components that are pivotal in histopathology. To address this gap, we introduce a novel INR framework that integrates auxiliary convolutional neural networks (CNN) with a standard MLP model. This dual-network approach not only facilitates pixel-level analysis, but also enhances the representation of local spatial variations, which is crucial for accurately rendering the complex patterns found in WSIs. Our experimental findings indicated a substantial improvement in the fidelity of histopathological image representation, as evidenced by a 3-6 dB increase in the peak signal-to-noise ratio compared to existing methods. This advancement underscores the potential of INRs to revolutionize digital histopathology, offering a pathway towards more efficient diagnostic imaging techniques. Our code is available at https://pnu-amilab.github.io/CINR/

Keywords: Implicit Neural Representation · Pathological Image

1 Introduction

Deep neural networks have considerably advanced complex imaging tasks across various domains of artificial intelligence owing to their robust representational

⋆ Corresponding author: mkim180@pusan.ac.kr

abilities. Implicit neural representations (INRs) have emerged as a popular alternative to traditional explicit frameworks that discretize the image space into rectangular grids. This method employs neural networks as representation functions, parameterizing (encoding) the signal of interest by capitalizing on their capacity for functional approximation. A multi-layer perceptron (MLP) is commonly used for generating pixel (or voxel) values as outputs, using coordinates as inputs [10, 7, 4].

This study was based on the hypothesis that INRs could lead to substantial advancements in digital histopathological imaging. Histopathology examination is essential for a definitive diagnosis based on biopsy samples. The advent of digital scanning technology has enabled the transformation of tissue samples on glass slides into whole-slide images (WSIs), facilitating high-resolution examinations at various magnifications. Owing to the high resolution and substantial size of these images, typically several gigabytes each, a pyramid file structure was adopted for efficient management and processing. This file structure incorporates multiple layers, each of which is a down-sampled version of the full-resolution image. Standard software typically loads a low-resolution layer to enable rapid interactions. Upon zooming in, the system dynamically retrieved the high-resolution tiles from the corresponding layer of the pyramid.

INRs function as continuous mappings within a continuous domain, allowing them to interpolate values between pixels. Consequently, the resolution-agnostic property of INRs has the potential to supersede the traditional pyramidal format of WSIs, which requires the storage of additional down-sampled versions and the intricate management of multiple tiling positions and resolutions. This could reduce the time and memory required to access an image at any magnification level, resolution, or specific local region of interest. Generating an image requires constructing a set of the desired pixel coordinates and inputting them into a network. Moreover, INRs have shown promise for memory-efficient image compression, suggesting a pathway for representing high-resolution pathological images with fewer network parameters, thereby reducing storage and data transfer demands[1, 8, 11, 2].

However, the principal challenge in applying INRs to pathological imaging is their limited representational capacity for high-frequency spatial components[3, 12, 13]. These components are proportionally more prominent in WSIs than in standard natural images due to the presence of intricate cellular patterns and microorganisms as small as a few micrometers, such as Helicobacter pylori. Numerous studies have attempted to reconstruct high-frequency patterns using publicly available natural-image datasets. Some studies have demonstrated that frequency encoding, which involves mapping Cartesian coordinates to a high-dimensional space using a sinusoidal pattern and feeding them into a network, can facilitate the learning of complex details more effectively. A technique known as parameter encoding has shown remarkable effectiveness by allowing trainable parameters to encode the coordinates[12]. An alternative method involves transforming the standard activation function into a sinusoidal function, leveraging its periodicity to capture fine image details[9]. Other proposed methods include

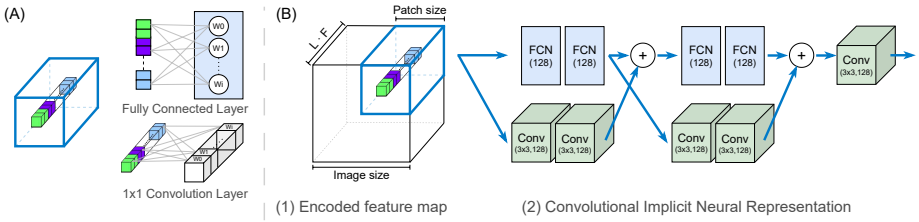

Fig. 1. (A) Fully Connected and Convolutional Layers: These layers are interchangeable if the convolutional layer has a kernel size of 1×1. The kernel values correspond to the weights in the fully connected layer. (B) Our Convolutional Implicit Neural Representation (CINR) Model: The collection of feature vectors from all target pixels is represented as a 3D tensor. This tensor is divided into multiple patches, with each patch processed by the model. The processing can occur in one of two ways: 1) treating all feature vectors within a patch as a batch, or 2) processing the 3D patch directly. In the model, each rectangle box represents a fully connected layer, and each cube box represents a convolutional layer.

decomposing an image into multi-resolution components, with each component represented by a distinct network, and then combining all representations[3, 13]. However, our empirical examinations of WSIs indicate that these methods are still inadequate for restoring complex signals that convey critical diagnostic information.

To address these challenges, we introduce a pioneering INR approach that employs the latest coordinate encoding strategy using hash tables at various resolutions. This model was augmented by integrating an auxiliary CNN with the MLP. This dual-network approach extends contextual analysis beyond individual pixels, allowing for a more accurate determination of pixel values. When a batch containing all pixels from a single image is input into only the MLP flow, the process can be likened to an image undergoing a series of multi-channel convolutions with merely 1 x 1 kernel size. The novel auxiliary flow is composed of a sequence of convolutions with 3 x 3 kernel sizes, allowing for the engagement of not only the target pixel position, but also adjacent positions, thereby adeptly managing local spatial fluctuations. Our experimental results highlight a significant improvement in high-frequency restoration within pathological images.

2 Method

2.1 Position encoding

The universal approximation theorems suggest that neural networks can be used to represent functions. Recent studies demonstrated that a simple MLP can reconstruct a continuous 2D image or 3D volumetric function by mapping arbitrary coordinates to their corresponding values. However, it has been shown that MLPs exhibit a spectral bias when fed raw input coordinates, leading to a

diminished ability to represent high spatial frequencies[12]. To mitigate this, positional encoding has been introduced, enabling networks to effectively capture high-frequency content.

In our implementation of pathological imaging, we leveraged a multi-resolution hash grid-encoding method[6], which is currently considered to be state-of-the-art based on various restoration metric scores. This method establishes L 2D grids across the domain, with each grid representing a level of resolution. At level l, the grid divides the full image domain into $N_l \times N_l$. The progressive N_l between the coarest and fineast resolutions was determined as $N_l = \lfloor N_{min} \cdot b_l \rfloor$ and $b = e^{(\ln N_{max} - \ln N_{min})/(L-1)}$. This method then encodes each grid point using a hash table of size T. Specifically, each grid point $\mathbf{x}^{(l)}$ at level $l \in \{0, 1, \ldots, L-1\}$ is mapped to an index using the corresponding hash table asfollows:

$$h(\mathbf{x}^{(l)}) = (\bigoplus_{i=1}^{2} x_i^{(l)} \pi_i) \bmod T, \tag{1}$$

where $\bigoplus$ denotes the bitwise XOR operator and π_i represents unique and large prime numbers. Each index stores F trainable parameters(features), denoted by a feature vector $\theta_{h(\mathbf{x}^{(l)})} \in \mathbb{R}^F$. For a given queried input coordinate $\bar{\mathbf{x}}$, a feature vector $\mathbf{v}^{(l)} \in \mathbb{R}^F$ is calculated for each level by interpolating nearby grid features. The final feature vector $\mathbf{z} \in \mathbb{R}^{LF}$ is then assembled through concatenation as

$$\mathbf{z} = [\mathbf{v}^{(1)}; \mathbf{v}^{(2)}; \ldots; \mathbf{v}^{(L)}], \ \mathbf{v}^{(l)} = \Psi(\{\theta_{h(\mathbf{x}^{(l)})} | \text{nearby corners } \mathbf{x}^{(l)} \text{ of } \bar{\mathbf{x}}\}), \tag{2}$$

where $\Psi(\cdot)$ denotes a linear interpolation operator. The entire encoding process is represented succinctly by $\mathbf{z} = \text{enc}(\bar{\mathbf{x}}; \Theta)$, where $\Theta = \{\theta_0, \ldots, \theta_{T-1}\}$. The feature vector $\mathbf{z}$ served as the network input. The optical hyperparmater values empirically found for pathological images are summarized in the supplementary.

2.2 Convolutional implicit neural representation

An INR maps each encoded vector $\mathbf{z} = \text{enc}(\bar{\mathbf{x}}; \Theta)$ to the vector $\mathbf{y} = \Theta(\mathbf{z}) \in \mathbb{R}^3$ denoting the pixel color values. Typically, a simple MLP, which is a fully connected network, is used to map the model $\Theta(\cdot)$. Assuming that the MLP consists of N layers and each layer has no bias, for an arbitrary pixel position $\bar{\mathbf{x}} = [\bar{x}_1; \bar{x}_2]$, the computation in the MLP can be expressed as

$$\boldsymbol{h}^{(n+1)} = \varphi(\boldsymbol{W}^{(n)} \boldsymbol{h}^{(n)}), \tag{3}$$

where $\boldsymbol{h}^{(n)}$ denotes the output of neurons in the lth layer or the input of neurons in the $n+1$th layer. $\boldsymbol{W}^{(n)}$ and $\varphi(\cdot)$ denote the weight and activation functions, respectively. $\boldsymbol{h}^{(n+1)}$ denote the output of the neurons in the $n+1$th layer. The initial input is $\boldsymbol{h}^{(0)} = \mathbf{z}$ and the final output is $\boldsymbol{h}^{(N)} = \mathbf{y}$. If a batch contains encoded vectors for all pixel positions from a single image $\mathbf{I}$, the computations for all positions are conducted in parallel.

As illustrated in Fig. 1 (A), these processes are analogous to those in CNNs. Assuming that the CNN consists of only N convolution layers and is fed by the image $\mathbf{I}$, the computation in the CNN can be expressed as

$$\boldsymbol{D}_j^{(n+1)} = \varphi(\sum_{k=1}^{K^{(n)}} \boldsymbol{D}_k^{(n)} * \boldsymbol{P}_{jk}^{(n)}), \tag{4}$$

where the operation $*$ denotes convolution, $K^{(n)}$ denotes the number of feature maps (channels) in the nth layer, $\boldsymbol{D}_k^{(n)}$ denotes the kth feature map (channel) in the nth layer, $\boldsymbol{p}_{jk}^{(n)}$ denotes the kth filter (patch or kernel) in the nth layer generating the jth feature map $\boldsymbol{D}_j^{(n+1)}$ in the next layer, and $\varphi(\cdot)$ denotes the activation function. Let each filter be of size 1×1 and $\boldsymbol{P}_{jk}^{(n)} = w_{jk}$. The channel-wise vector at pixel position $(\bar{x}_1, \bar{x}_2)$ in feature map $\boldsymbol{D}_k^{(n)}$ can be expressed as

$$\mathbf{d}^{(n)} = [\boldsymbol{D}_1^{(n)}[\bar{x}_1, \bar{x}_2], \boldsymbol{D}_2^{(n)}[\bar{x}_1, \bar{x}_2], \ldots, \boldsymbol{D}_{K^{(n)}}^{(n)}[\bar{x}_1, \bar{x}_2]]^T. \tag{5}$$

If $\mathbf{d}^{(n)}$ is set to $\mathbf{h}^{(n)}$, the form of the CNN computation (Eq. 4) is identical to that of MLP (Eq. 3).

In this context, to obtain the target position values $\mathbf{I}(\bar{\mathbf{x}}^*)$, the MLP, which functions as a CNN with 1×1 kernels, can access the feature vector $\mathbf{z}$ only of the target position $\bar{\mathbf{x}}^*$. Our central idea involves expanding the receptive field within the network, which enables the inclusion of features from adjacent pixels to determine the target values. Extending the single level $[v^{(n)}]$ to multiple levels $\mathbf{z} = [v^{(n)}; \ldots; v^{(n)}]$ contributes to the restoration of high-frequency components, as indicated in previous research[6]. Building on this, our further extension from the multilevel feature $\mathbf{z}$ to a set of multilevel features $\{\mathbf{z} = \mathrm{enc}(\mathbf{p}) \mid \text{adjacent pixels } \mathbf{p} \text{ of the target pixel } \mathbf{p}^*\}$ is expected to enhance restoration even further. This can be achieved using a CNN architecture that employs larger kernels.

Our CINR model is depicted in Fig. 1 (B). A full-size image was segmented into overlapping patches (each 128×128 pixels), and the tensor (a collection of encoded vectors) derived from each patch area was fed into the model for training purposes. The tensor is processed using two parallel-network flows in the first stage. The first flow, a standard MLP with two layers, focuses on the features of the target position, whereas the second flow, a CNN with two layers and 3×3 kernels, encompasses the features from the surrounding areas of the target position. In the second stage, the first flow processes the concatenated maps, which are the resultant feature maps from both flows at the first stage, through another standard MLP with two layers. Concurrently, a CNN with two layers and 3×3 kernels processes the resultant maps from the first flow of the first stage. The outputs from these two flows are merged and passed through an additional CNN layer to integrate the features, culminating in the determination of the target values.

3 Results

3.1 Experiments

To assess the representational capabilities of our model, CINR, we utilized a publicly available digital pathology image dataset from The Cancer Genome Atlas (TCGA). The dimensions of each image are $(10,000, 10,000, 3)$. The images were segmented into patches of $(128, 128, 3)$, with each patch overlapping its neighbors by 20%. During training, each batch comprised 100 patches. For inference, the images were divided into non-overlapping patches of the same size to evaluate performance of each model. We benchmark our model against two conventional INR models: 1) NGP[6], which employs an MLP with four layers and 128 nodes per layer (totaling 57,344 trainable parameters), and 2) ENGP, which is an enhanced version with an MLP of eight layers and 256 nodes per layer (totaling 475,136 trainable parameters). All models utilized the same hash encoding technique (totaling 257,270,128 trainable parameters) and were configured with identical hyperparameters. The mean squared error was used as the loss function. These implementations were based on the PyTorch library, incorporating elements from Instant NGP[6] and tiny-cuda-nn[5] and were executed on an NVIDIA A5000 GPU.

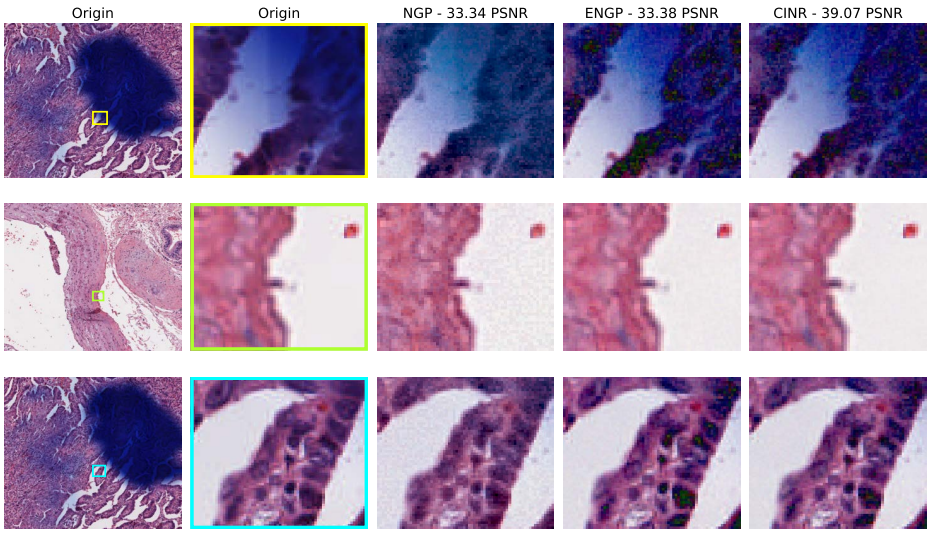

Fig. 2. Original and reconstructed images by INR models. The first column presents the original images from three different samples. The second column shows zoomed-in views of specific local areas highlighted by rectangular boxes in each original image. The subsequent columns showcase the INR models' reconstructions of these local areas.

3.2 Reconstruction Results

Fig. 2 presents the representative results of the selected test images for each model. Compared to the original (ground-truth) images, our CINR model demonstrated superior reconstruction of complex cellular and pathological patterns, including some grain-like spots, without any smoothing effects, surpassing the performance of other models.

These outcomes are supported by the quantitative evaluation results shown in Fig. 3. The graph displays metrics such as the average PSNR and Structural Similarity Index Measure (SSIM) to assess the similarity between the ground truth and reconstructed images. CINR achieved significantly higher scores than both NGP and ENGP. Notably, ENGP did not consistently experience an improvement in reconstruction quality, despite its larger network size. This indicates that the enhanced performance of CINR is not merely due to an increase in the number of trainable parameters.

Fig. 4 visually emphasizes the discrepancies between the original and reconstructed images. Generally, all models struggled with the restoration of stained parts because of their complexity. Our model exhibits fewer gaps in the structural patterns and outlines. In the images illustrating the differences in the spatial frequency domain, it is evident that the lower-frequency components were more effectively restored in all models. However, our model demonstrated superior restoration of high-frequency components compared with the others.

Fig. 5 shows the PSNR of the reconstructed images during the learning period. NGP and ENGP exhibited faster convergence to a plateau. In contrast, CINR demonstrates relatively slower convergence; however, after a certain period (150 min), it surpasses the other models and progressively improves its performance.

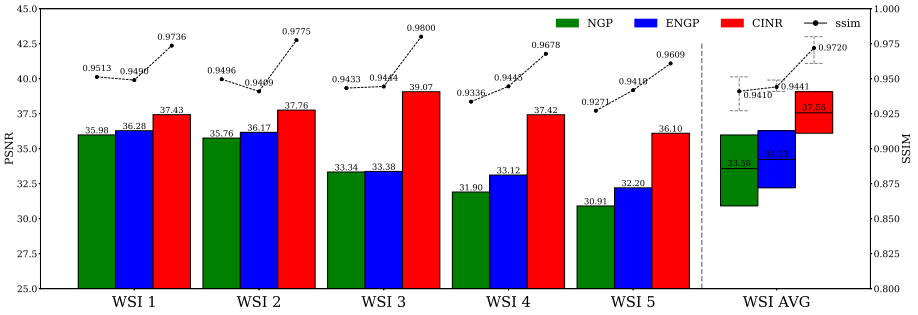

Fig. 3. Quantitative measures of similarity between original and reconstructed images. Bars represent Peak Signal-to-Noise Ratio (PSNR) values, and circle dots placed on the dotted lines represent Structural Similarity Index Measure (SSIM) scores.

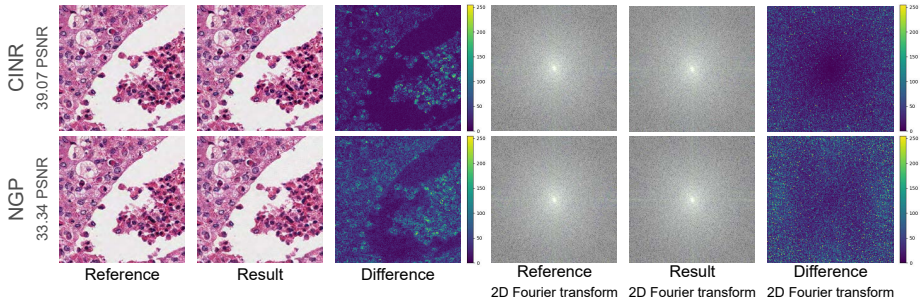

Fig. 4. The first column presents original images as a reference, while the second column shows the reconstruction results from CINR and NGP. The third column highlights the differences between the original and reconstructed images. The fourth and fifth columns display frequency components derived from applying a 2D Fourier transform to both the reference and reconstructed images, respectively. The sixth column illustrates the differences between the original and reconstructed images in the spatial frequency domain.

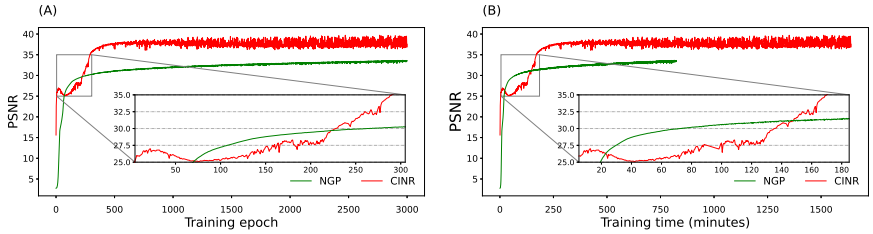

Fig. 5. (A) PSNR over learning epoch. (B) PSNR over learning time (minutes).

4 Conclusion

Reconstructing images in high spatial frequency domains presents a significant challenge in various scientific disciplines. CINR markedly enhances the restoration of high-resolution images with high spatial frequencies by leveraging features from adjacent locations encoded through a convolutional network. This approach ensures a notable improvement in the performance of complex pathological images.

Structurally, the potential for recovering more high-frequency components and achieving faster learning speeds remains unexplored. Expanding upon the simple CNN framework by incorporating residual or attention mechanisms is anticipated to further improve performance. However, a crucial bottleneck of CINR lies in the extensive parameters required for multigrid hash encoding, which hamper GPU memory allocation during training and subsequently slow learning speeds. Furthermore, the bit count of the encoding and network parameters approaches those of the original images, posing challenges in replacing traditional image storage methods with INR-based approaches in tissue pathology.

Future developments should focus on network compression, parameter sharing, and innovative encoding strategies to overcome these challenges.

In conclusion, CINR holds substantial potential to revolutionize digital pathological imaging systems. Advancements in memory reduction could lead to new data compression protocols that surpass the conventional JPEG standard for long-term storage. Additionally, it offers the possibility of simplifying complex pyramid file structures by rapidly constructing images at any requested resolution.

Acknowledgments. This research was supported by a fund(SMF-AI-EJ003-2023) by Seegene Medical Foundation. This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) under the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2024-RS-2023-00254177) grant funded by the Korea government(MSIT). This study was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2094778).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Dupont, E., Loya, H., Alizadeh, M., Goliński, A., Teh, Y.W., Doucet, A.: COIN++: Neural Compression Across Modalities (Dec 2022), http://arxiv.org/abs/2201.12904, arXiv:2201.12904 [cs, eess, stat]
2. Li, L., Shen, Z., Wang, Z., Shen, L., Bo, L.: Compressing Volumetric Radiance Fields to 1 MB (Nov 2022), http://arxiv.org/abs/2211.16386, arXiv:2211.16386 [cs]
3. Lindell, D.B., Van Veen, D., Park, J.J., Wetzstein, G.: Bacon: Band-limited Coordinate Networks for Multiscale Scene Representation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16231–16241. IEEE, New Orleans, LA, USA (Jun 2022). https://doi.org/10.1109/CVPR52688.2022.01577, https://ieeexplore.ieee.org/document/9880123/
4. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. arXiv:2003.08934 [cs] (Aug 2020), http://arxiv.org/abs/2003.08934, 1554 citations (Semantic Scholar/arXiv) [2022-10-19] arXiv: 2003.08934
5. Müller, T.: tiny-cuda-nn (4 2021), https://github.com/NVlabs/tiny-cuda-nn
6. Müller, T., Evans, A., Schied, C., Keller, A.: Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. arXiv:2201.05989 [cs] (Jan 2022), http://arxiv.org/abs/2201.05989, 159 citations (Semantic Scholar/arXiv) [2022-10-19] arXiv: 2201.05989
7. Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 427–436 (2015)
8. Pham, T., Yang, Y., Mandt, S.: Autoencoding Implicit Neural Representations for Image Compression

9. Sitzmann, V., Martel, J.N.P., Bergman, A.W., Lindell, D.B., Wetzstein, G.: Implicit Neural Representations with Periodic Activation Functions (Jun 2020), http://arxiv.org/abs/2006.09661, arXiv:2006.09661 [cs, eess]
10. Stanley, K.O.: Compositional pattern producing networks: A novel abstraction of development. Genetic Programming and Evolvable Machines **8**(2), 131–162 (Jun 2007). https://doi.org/10.1007/s10710-007-9028-8, http://link.springer.com/10.1007/s10710-007-9028-8
11. Strümpler, Y., Postels, J., Yang, R., van Gool, L., Tombari, F.: Implicit Neural Representations for Image Compression (Aug 2022), http://arxiv.org/abs/2112.04267, arXiv:2112.04267 [cs, eess]
12. Tancik, M., Srinivasan, P.P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J.T., Ng, R.: Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains (Jun 2020). https://doi.org/10.48550/arXiv.2006.10739, http://arxiv.org/abs/2006.10739, arXiv:2006.10739 [cs]
13. Wu, Z., Jin, Y., Yi, K.M.: Neural fourier filter bank. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14153–14163 (2023)