Unified Implicit Neural Stylization

Zhiwen Fan[1*], Yifan Jiang[1*], Peihao Wang[1*], Xinyu Gong[1],
Dejia Xu[1], Zhangyang Wang[1]

[1]The University of Texas at Austin

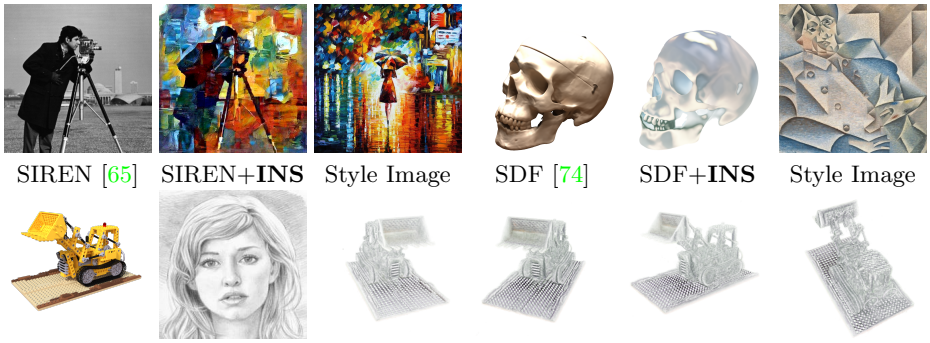

Fig. 1. Representative visual examples generated by the proposed method. We show stylized results on three different types of implicit representation, including 2D coordinate-based mapping function (SIREN [65]), Signed Distance Function(SDF [55]), and Neural Radiance Field (NeRF [50]).

Abstract. Representing visual signals by implicit representation (e.g., a coordinate based deep network) has prevailed among many vision tasks. This work explores a new intriguing direction: training a **stylized** implicit representation, using a generalized approach that can apply to various 2D and 3D scenarios. We conduct a pilot study on a variety of implicit functions, including 2D coordinate-based representation, neural radiance field, and signed distance function. Our solution is a Unified **I**mplicit **N**eural **S**tylization framework, dubbed **INS**. In contrary to vanilla implicit representation, INS decouples the ordinary implicit function into a style implicit module and a content implicit module, in order to separately encode the representations from the style image and input scenes. An amalgamation module is then applied to aggregate these information and synthesize the stylized output. To regularize the geometry in 3D scenes, we propose a novel self-distillation geometry consistency loss which preserves the geometry fidelity of the stylized scenes. Comprehensive experiments are conducted on multiple task settings, including novel view synthesis of complex scenes, stylization for implicit surfaces, and fitting images using MLPs. We further demonstrate that the learned representation is continuous not only spatially but also style-wise, leading to effortlessly interpolating between different styles and generating images with new mixed styles. Please refer to the video on our project page for more view synthesis results: https://zhiwenfan.github.io/INS.

* equal contribution

1 Introduction

Implicit Neural Representation (INS) has gained remarkable popularity in representing concise signal representation in computer vision and computer graphics [65,50,55,46,75]. As an alternative to discrete grid-based signal representation, implicit representation is able to parameterize modern signals as samples of a continuous manifold, using multi-layer perceptions (MLP) to map between coordinates and signal values. Several seminal works [50,65,74] have verified the effectiveness of INS in representing image, video, and audio. Followups further apply INS to more challenging tasks including novel-view synthesis [50,5,6,78], 3D-aware generative model [82,9,8], and inverse problem [13,69].

While implicit neural representation reveals multiple advantages compared to conventional discrete signals, a general question of curiosity might be: *which and how modern visual signal processing approaches/tasks designed for discrete signals can also be applied to continuous representations?* Research pursuing this answer has been conducted on implicit neural representation since its origin. Chen *et al.* [14] apply a local implicit function to image super-resolution and they observe that INS can surpass bilinear and nearest upsampling. Sun *et al.* [69] demonstrates the effectiveness of INS in the context of sparse-view X-ray CT. Dupont *et al.* [19] propose to store the weights of a neural implicit function instead of pixel values, which surprisingly outperforms JPEG compression format. [79] further demonstrate superior video compression using similar ideas.

We investigate a novel setting: to yield visually pleasing stylized examples under various 2D and 3D scenarios, using a generalized approach leveraging implicit neural representations. Note that, training a stylized implicit neural representation still faces many hurdles. On one hand, the aforementioned works mostly have the access to dense measurements or at least sparse clean data, which enables training an implicit neural network **under the supervision of target signal**. In contrast to those tasks/approaches, current image stylization mechanisms are mostly conducted in an unsupervised manner, due to the absence of stylized ground truth data. Consequently, it is still unknown whether coordinate-based MLP can be optimized without accessing corresponding ground truth signals. On the other hand, existing style images are mostly based on 2D scenes, which raises obstacles when being considered as the appearance of 3D implicit representation. Prior art [16] attempted on marrying stylization with one specific type of Implicit Neural Representation, the neural radiance field (NeRF). Nevertheless, they still capture the statistics of style information by a series of pre-trained convolution-based hypernetwork, rather than implicitly encoding stylization. As indicated by recent literature [43], training a robust hypernetwork requires a large amount of training samples, while novel-view synthesis tasks commonly hold no more than hundreds of views, potentially jeopardizing the synthesized visual quality.

To conquer the aforementioned fragility, we propose an Unified **I**mplicit **N**eural **S**tylization framework, coined as **INS**. Different from the vanilla implicit function which is built upon a single MLP network, the proposed framework divides an ordinary implicit neural representation to multiple individual

components. Concretely speaking, we introduce a Style Implicit Module to the ordinary implicit representation, and coin the later one as Content Implicit Module in our framework. During the training process, the stylized information and content scene are encoded as one continuous representation, and then fused by another amalgamation module. To further regularize the geometry of given scenes, we utilize an additional geometry consistency loss for self-distillation loss on top of the rendered density. Eventually, INS is able to render view-consistent stylized scenes from novel views, with visually impressive texture details: a few examples are shown in Figure. 1.

Our contributions are outlined below:

- We propose a Unified Implicit Neural Stylization framework, dubbed INS. INS consists of a style implicit module, a content implicit module, and an amalgamation module, which enables us to synthesize promising stylized scenes under both 2D and 3D scenarios.
- We conduct comprehensive experiments on several popular implicit representation frameworks in this novel stylization setting, including 2D coordinate-based framework (SIREN [65]), Neural Radiance Field (NeRF [50]), and Signed Distance Functions (SDF [55]). The rendering results are found to be more consistent, in both shape and style details, from different views.
- We further demonstrate that INS is able to learn representations that are continuous not only with regard to spatial placements (including views), but also in the style space. This leads to effortlessly interpolating between different styles and generating images rendered by the new mixed styles.

2 Related Works

2.1 Implicit Function

Recent research has exhibited the potential of Implicit Neural Representation (INS) to replace traditional discrete signals with continuous functions parameterized by multilayer perceptrons (MLP), in computer vision and graphics [70,66]. The coordinate-based neural representations [15,46,47] have become a popular representation for various tasks such as representing image/video [65,19,79], 3D reconstruction [24,3,54,15,26,46,53,55,58,60], and 3D-aware generative modelling [9,17,27,29,45,52,61,82]. Analogously, as this representation is differentiable, prior works apply coordinate-based MLPs to many inverse problems in computational photography [13,11,67,2,62] and scientific computing [41,28,81].

2.2 Implicit 3D Scene Representation

The recent paradigm of coordinate-based neural representations has prevailed among 3D scene representation tasks, which simply adopt an MLP that maps from any continuous input 3D coordinate to the geometry and appearance of the scene, including signed distance function (SDF) [55,37,4,47,26,68,36], 3D occupancy network [46,15], and so on. In addition to representing shape, implicit representation has also been extended to encode object appearance. Among them,

Neural Radiance Field (NeRF [50]) is one of the most effective coordinate-based neural representations for photo-realistic view synthesis that represents a scene as a field of particles. Draw inspiration from the preliminary success made by NeRF, a lot of following works further improve and extend it to wider application [5,6,7,20,44,76,48,57]. Jonathan *et al.* [5,6] propose to adopt cone based sampling instead of a ray through a pixel, which is able to represent multi-scale novel-view synthesis without aliasing artifacts. Peter *et al.* [30] precompute the synthesized views and store them via the sparse neural radiance grid, which largely reduces the inference time and achieves real-time rendering. Park *et al.* [57] propose HyperNeRF, enabling dynamic scene representation through implicit function. Different from the aforesaid approaches, training a stylized implicit representation can access no ground truth signals, which further amplifies the difficulty of optimizing implicit neural representation.

2.3 Stylization

Traditionally, image stylization is formulated as a painterly rendering process through stroke prediction [80,77]. The first neural style transfer method, proposed by Gatys *et al.* [22], builds an iterative framework to optimize the input image in order to minimize the content and style loss defined by a pre-trained VGG network. Due to the frustratingly large cost of training time, a number of follow-ups further explore how to design a feed-forward deep neural networks [38,71], which obtain real-time performance without sacrificing too much style information. Recently, efforts have been devoted to extending singe-image neural style transfer to more general scenarios, including video stream [59,34,10,12], multi-stereo [25], and 3D environment [51,33,16]. The essential challenge for the aforesaid tasks lies in preventing flicker artifacts brought by the inconsistency between different views. To solve this issue, Ruder *et al.*[59] propose to add a temporal constraint loss to the vanilla iterative-based neural style transfer approach [23]. Huang *et al.*[32] are able to generate temporally coherent stylized videos in real-time, by incorporating temporal consistency into a feed-forward neural networks during the training stage. Similarly, Gong *et al.* [25] propose a dual-path network to share information between multi-views, which accomplishes view-consistent stereoscopic style transfer. Later, Kato *et al.* [39] consider apply for style transfer on mesh, to reconstruct 3D stylized objects. Hollein *et al.* [31] further extend it to the complex indoor-scene using a depth-aware optimization at different screen-space resolutions. Mu *et al.* [51] and Huang *et al.* [33] adopt point cloud based scene representation to stylized single image, and generate view-consistent stylized output from different poses.

The most related work to our method is Chiang *et al.* [16], which adopts a hypernetwork to generate per-style parameters for neural radiance field and renders stylized novel-view synthesis. In comparison, our proposed INS framework can work for more general implicit representations beyond neural radiance field, and can also be extended to encoding multiple styles. Experiments demonstrate that INS outperforms [16] with a large margin.

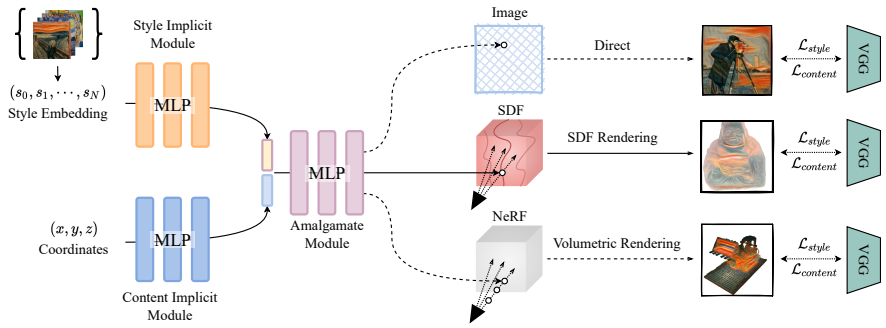

Fig. 2. The main pipeline of unified implicit neural stylization (INS) framework and its components. We took NeRF with the proposed INS, for example, it inputs with implicit coordinates along with ray directions and style embeddings. Style Implicit Module (SIM) and Content Implicit Module (CIM) are used to extract conditional implicit style features and implicit scene features. Amalgamate Module (AM) is applied to fuse features in the two spaces, generating scene density and view-dependent color value. An implicit rendering step is applied on the top of AM (i.e., Volume rendering for NeRF, surface rendering for SDF) to render the pixel intensity. VGG used to generate style supervision in training is omitted in this figure for simplicity.

3 Preliminary

This section introduces the relevant background on several implicit representations and volumetric radiance representations, including image fitting [65], signed distance function [55], and neural radiance field [50]

Implicit Image Fitting: The most prototypical example of neural implicit representation is image regression [65,70]. Consider fitting a function $f : \mathbb{R}^2 \to \mathbb{R}^3$ that encodes the pixel array of a given image $\mathcal{I}$ into a continuous representation. Function $f(\boldsymbol{x})$ takes pixel coordinates $\boldsymbol{x} \in \mathbb{R}^2$ as the inputs, and outputs the corresponding RGB colors $\boldsymbol{c} \in \mathbb{R}^3$. Parameterizing f with a multi-layer perception networks (MLPs), it can be optimized by the mean-squared error (MSE) loss function $\mathcal{L} = \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}(\mathcal{I})}\|f(\boldsymbol{x}) - \boldsymbol{c}\|_2^2$, where $\mathbb{P}(\mathcal{I})$ is a probability measure support in image lattice $\mathcal{I}$.

Neural Radiance Field: In contrast to point-wisely regression of implicit fields, NeRF proposes to reconstruct a radiance field by inversing a differentiable rendering equation from captured images. NeRF learns an MLP $f : (\boldsymbol{x}, \boldsymbol{\theta}) \mapsto (\boldsymbol{c}, \sigma)$ with parameters $\boldsymbol{\Theta}$, where $\boldsymbol{x}$ is the spatial coordinate in 3D space and $\boldsymbol{\theta}$ represents the view directions $\in [-\pi, \pi]^2$. The output $\boldsymbol{c} \in \mathbb{R}^3$ indicates the predicted color of the sampled point, $\sigma \in \mathbb{R}_+$ signifies its density value. The pixel color intensity can be obtained using volume rendering [18] by ray tracing and integrating the predicted color and density along the ray. Specifically, to render a pixel on the image plane, we cast a ray $\boldsymbol{r} = (\boldsymbol{o}, \boldsymbol{d}, \boldsymbol{\theta})$ through the

pixel and accumulate the color and density of K point samples along the view direction in the 3D space. The pixel color intensity can be estimated:

$$\boldsymbol{C}(\boldsymbol{r}|\boldsymbol{\Theta}) = \sum_{k=1}^{K} T_k(1 - \exp(-\sigma_k \Delta t_k))\boldsymbol{c}_k, \tag{1}$$

where $(\boldsymbol{c}_k, \sigma_k) = f(\boldsymbol{x}_k, \boldsymbol{\theta})$, $\boldsymbol{x}_k = \boldsymbol{o} + t_k\boldsymbol{d}$, t_k are the marching distance of sampled points, and $T_k = \exp(-\sum_{l=1}^{k-1} \sigma_l \Delta t_l)$ is known as the transmittance to model occlusion. $\Delta t_k = t_{k+1} - t_k$ indicates the distance of sampled point in 3D space. With this approximated rendering pipeline, the model weights are optimized by minimizing the L_2 distance between rendered ray colors $\boldsymbol{C}(\boldsymbol{r})$ and captured pixel colors $\widehat{\boldsymbol{C}}(\boldsymbol{r})$ as follows:

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} \mathbb{E}_{\boldsymbol{r}\sim\mathbb{P}(\mathcal{R})} \left\| \boldsymbol{C}(\boldsymbol{r}|\boldsymbol{\Theta}) - \widehat{\boldsymbol{C}}(\boldsymbol{r}) \right\|_2^2, \tag{2}$$

where $\mathcal{R}$ is a collection of rays cast from all pixels in the training set, and $\mathbb{P}(\mathcal{R})$ defines a distribution over it.

Implicit Surface Representation: Signed Distance Function (SDF) $f : \mathbb{R}^3 \to \mathbb{R}$ is an implicit representation of 3D geometries. SDF specifies each spatial point with the signed distance to the implicit iso-surface, where the sign indicates whether the point is inside or outside the object. Recent works of [56,64] propose to employ MLPs to represent this continuous field via direct supervision using point clouds. To optimize a textured SDF from multi-view images like NeRF [48], Yariv *et al.*[74] proposes a neural rendering pipeline, named IDR, which enables rendering images from an SDF. With this framework, one can indirectly supervise SDF using its multi-view projections. Suppose given a camera pose, we can cast rays $\boldsymbol{r} = (\boldsymbol{o}, \boldsymbol{v})$ through each pixel to trace an intersected point with the surface:

$$\hat{\boldsymbol{x}} = \boldsymbol{o} + t_0\boldsymbol{v} - \frac{\boldsymbol{v}}{\nabla f_0 \cdot \boldsymbol{v}_0} f(\boldsymbol{o} + t_0\boldsymbol{v}), \tag{3}$$

where t_0, $\boldsymbol{v}_0$ and f_0 are initial states when performing ray tracing (see [74]). After obtaining the intersection $\hat{\boldsymbol{x}}$ of ray and surface, IDR also lets the SDF network f output an appearance embedding $\hat{\boldsymbol{\gamma}}$, and computes the normal $\hat{\boldsymbol{n}} = \nabla f(\hat{\boldsymbol{x}})$. Then the ray color can be rendered by another rendering MLP r conditioned on both point coordinate $\hat{\boldsymbol{x}}$ and normal $\hat{\boldsymbol{n}}$:

$$\boldsymbol{C}(\boldsymbol{r}|\boldsymbol{\Theta}) = r(\hat{\boldsymbol{x}}, \hat{\boldsymbol{n}}, \boldsymbol{d}, \hat{\boldsymbol{\gamma}}). \tag{4}$$

Similar to NeRF [48], f and r are simultaneously optimized by photometric loss between captured image pixels and rendered rays (see Equation 2).

4 Method

We next illustrate the main pipeline of Unified **I**mplicit **N**eural **S**tylization (INS). INS consists of a Style Implicit Module (SIM) to transform the input style embedding into implicit style representations, a Content Implicit Module (CIM) to

map the input coordinates into implicit scene representations, and an Amalgamate Module (AM) which amalgamates the two representation to predict RGB intensity. To preserve the geometry fidelity while generating the stylized texture of rendered views, a self-distilled geometry consistency regularization is applied upon INS framework.

4.1 Implicit Style and Content Representation

Generating the stylized images $\boldsymbol{Y}$ can be formulated as an energy minimization problem [22]. It consists of a *content loss* and a *style loss*, defined under a pre-trained VGG network. We build upon the prior work and thus propose our implicit stylization framework for SIREN, SDF and NeRF.

Content Representation The content loss in 2D images stylization pipeline L_{content} is defined as:

$$\mathcal{L}_{\text{content}}(\boldsymbol{C}, \boldsymbol{Y}) = \frac{1}{C_{i,j} W_{i,j} H_{i,j}} \left\| F_{i,j}(\boldsymbol{C}) - F_{i,j}(\boldsymbol{Y}) \right\|_F^2, \tag{5}$$

where $\boldsymbol{C}$ denotes the content ground truth image and $\boldsymbol{Y}$ denotes synthesized output, $F_{i,j}$ denotes the feature map extracted from a VGG-16 model pre-trained on ImageNet, i represents its i-th max pooling, and j represents its j-th convolutional layer after i-th max pooling layer. $C_{i,j}$, $W_{i,j}$ and $H_{i,j}$ are the dimensions of the extracted feature maps. We adapt the content loss on the output of INS pipeline to preserve the content of the predicted color image patch, we choose $i = 2$, $j = 2$ by default.

Style Representation To extract representation of the stylized information, [63] introduces a different feature space to capture texture information [21]. Similar to the content loss, the feature space is built upon the filter response in multiple layers of a pre-trained VGG network. By capturing the correlations of the filter responses expressed by the Gram matrix $\boldsymbol{G}_{i,j} \in \mathbb{R}^{C_{i,j} \times C_{i,j}}$ between the style image $\boldsymbol{S}$ and the synthesized image $\boldsymbol{Y}$, multi-scale representations can be obtained to capture the texture information from the style image and endow such texture on the stylized image. Here, we define our style loss $\mathcal{L}_{\text{style}}$ using the same layers of VGG-16 with [22] on the top of the prediction of implicit neural representations and the given style image:

$$\mathcal{L}_{\text{style}}(\boldsymbol{S}, \boldsymbol{Y}) = \sum_{(i,j) \in \mathcal{J}} \left\| \boldsymbol{G}_{i,j}(\boldsymbol{S}) - \boldsymbol{G}_{i,j}(\boldsymbol{Y}) \right\|_F^2, \tag{6}$$

$$\text{where } [\boldsymbol{G}_{i,j}]_{c,c'}(\boldsymbol{Y}) = \frac{1}{C_{i,j} W_{i,j} H_{i,j}} \sum_{k=1}^{H \times W} F_{i,j}(\boldsymbol{Y})_{c,k} F_{i,j}(\boldsymbol{Y})_{c',k}, \tag{7}$$

where $\mathcal{J}$ are the indices of selected feature maps. In practice, we choose $\mathcal{J} = \{(1,2), (2,2), (3,3), (4,3)\}$ in our experiments.

Conditional INS Stylization Conditional encoding has been widely applied in convolutional networks [35,73]. Similarly, we propose the conditional implicit representation by adding style condition embeddings with a style implicit module to extract style-dependent features and the fusion using the subsequent amalgamation module, which is shown in Figure 2. We transform the input style embedding to scene feature maps via SIM and concatenate it with the intermediate feature map from CIM. In training, we prepare n style images with an n dimensional one-hot vector and make a mini-batch with the combinations of one content training patch and all the style training images. The one-hot conditional vectors are fed into SIM to output w-dimensional intermediate features, they are concatenated with the implicit representations from CIM. The following layers of AM take the two features, aggregate them to predict the pixel intensity along with volume rendering. VGG [63] is appended on the top of the INS pipeline to produce style and content constraints in training. Conditional scene stylization can be achieved via the scene embeddings and the target style images. During the inference stage, we remove the VGG network and the INS framework becomes a pure MLPs-based network.

4.2 Geometry Consistency for Neural Radiance Field

NeRF [50] represents scenes as colorized volume densities and integrates radiance along with rays via alpha blending [42]. It learns reasonable 3D geometry inherently due to the particular design of implicit radiance field and the supervision from multiple views. However, the INS framework is expected to integrate style statistic from 2D image into 3D radiance field, where no multi-view style images accessible during training process. To specialize INS for neural radiance fields, we propose to regularize INS with proper geometry constraint to produce faithful shape and appearance.

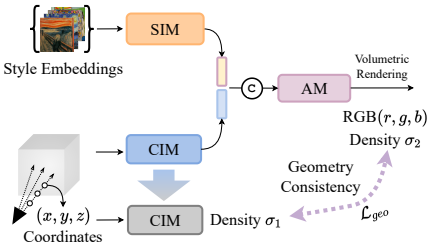

Fig. 3. The proposed self-distilled geometry consistency.

As the ground truth of target geometry is unavailable in most novel view synthesis benchmarks, we seek help from the self-distillation framework [72]. Concretely speaking, we first train the content implicit module (CIM) only to obtain a clean geometry σ_1, following vanilla NeRF training pipeline. After that, we save the trained weight of CIM as a checkpoint (as shown in the grey block in Figure 3) and continue to optimize the whole INS framework. In the meanwhile, we adopt a self-distilled geometry constraint between the original geometry σ_1 produced by the checkpoint weight and the final geometry σ_2 reconstructed by the implicit neural stylization framework. The objective of self-distilled geometry consistency loss is formulated as $\mathcal{L}_{\text{geo}} = |\sigma_1 - \sigma_2|$, where we adopt the mean-absolute error for the densities of each sampled point. As a result, the whole pipeline is illustrated in Figure 3.

Sampling-Stride Ray Sampling Neural Radiance Field casts a number of rays (typically not adjacent) from camera origin, intersecting the pixel, into the volume and accumulating the color based on density along the ray.

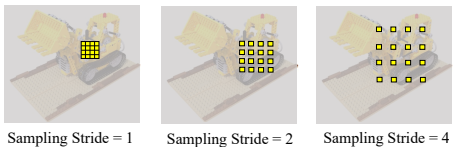

Fig. 4. Illustration of the Sampling Stride (SS) strategy in the ray sampling stage. With a given sampling stride larger than 1, we can approach a larger receptive field without sacrificing additional computational cost.

While our model input with rays intersected with an image patch of size $\mathcal{P} \in K \times K$, predicting the stylized patch $\mathcal{P}' \in K \times K$ with its texture closed to the given style images. 2D style transfer methods [10,34] typically crop the patch larger than 256×256. However, it is too expensive for the neural radiance field as it queries the MLPs more than $256 \times 256 \times N$ times of the MLP model in each iteration [50], where N indicates sampled points number alone each ray.

Similar to [61,45], we adopt a *Sampling-Stride Ray Sampling* strategy to enlarge the receptive field of the patch to capture a more global context. The details of the ray sampling can be seen in Figure 4, where a sampling stride larger than 1 can result in a large receptive field while keeping computational cost fixed.

4.3 Optimization

Let $\boldsymbol{Y}(\boldsymbol{T})$ denote an INS model parameterized by $\boldsymbol{\Theta}$, which synthesizes an image (patch) from the view specified by the camera pose $\boldsymbol{T} \in \mathbb{R}^{4\times4}$ by marching and rendering the rays for all the pixels on the image (patch). Given multi-view images and the corresponding camera parameters $\mathcal{T} = \{\boldsymbol{C}_i, \boldsymbol{T}_i\}_{i=1}^{N}$, as well as a set of style images $\mathcal{S} = \{\boldsymbol{S}_i\}_{i=1}^{M}$, we train the INS $\boldsymbol{Y}(\boldsymbol{T})$ using a combination of losses including reconstruction loss $\mathcal{L}_{\text{recon}}$, geometry consistency loss $\mathcal{L}_{\text{geo}}$, content loss $\mathcal{L}_{\text{content}}$ and style loss $\mathcal{L}_{\text{style}}$:

$$\begin{aligned}\mathcal{L}_{\text{total}}(\boldsymbol{\Theta}|\mathcal{T},\mathcal{S},\boldsymbol{\Theta}_{\text{vgg}}) &= \mathbb{E}_{(\boldsymbol{C},\boldsymbol{T})\sim\mathbb{P}(\mathcal{T})}\left[\mathcal{L}_{\text{recon}}(\boldsymbol{Y}(\boldsymbol{T}),\boldsymbol{C}) + \lambda_1\mathcal{L}_{\text{geo}}(\boldsymbol{Y}(\boldsymbol{T}))\right] \\ &+ \mathbb{E}_{\boldsymbol{S}\sim\mathbb{P}(\mathcal{S})}\,\mathbb{E}_{(\boldsymbol{C},\boldsymbol{T})\sim\mathbb{P}(\mathcal{T})}\left[\lambda_2\mathcal{L}_{\text{content}}(\boldsymbol{Y}(\boldsymbol{T}),\boldsymbol{C}|\boldsymbol{\Theta}_{\text{vgg}}) + \lambda_3\mathcal{L}_{\text{style}}(\boldsymbol{Y}(\boldsymbol{T}),\boldsymbol{S}|\boldsymbol{\Theta}_{\text{vgg}})\right],\end{aligned} \tag{8}$$

where λ_1, λ_2, λ_3 control the strength of each loss term, $\boldsymbol{\Theta}_{\text{vgg}}$ denotes the parameters of the VGG network, and $\mathbb{P}(\cdot)$ defines a distribution over a support.

5 Experiments

In this section, we conduct experiments on several applications of Implicit Neural Representations (INR) to assess our method. We first provide qualitative results of our proposed INS to INR for fitting an image [65], SDF [74] and novel view synthesis [50] to demonstrate INS [50] transfers the style more faithfully with view-consistency. Then, we show INS can interpolate between multiple style images. Finally, we analyze our proposed modules in the ablation study.

5.1 Stylization on SIREN MLPs

Fig. 5. Visual examples generated by applying INS to SIREN [65]. Given an input image used for fitting and style images, SIREN+INS can express the style statistics via an implicit manner(i.e., MLPs).

As one representative example, we apply INS on fitting an image via SIREN [66] MLPs. We reuse the original SIREN framework [65] as Content Implicit Module (CIM) and follow its training recipes to fit images of 512×512 pixels. Besides that, we also incorporate the Stylization Implicit and Amalgamation Modules on SIREN. VGG network is appended on the output to provide style supervision during training. As seen in Figure 5, the proposed framework is successful in representing the images with the given style statistics in an implicit way.

5.2 Novel View Synthesis with NeRF

NeRF [50] renders the novel view of a scene with MLPs which take in coordinates on 3D space along with view directions. It is supervised with RGB colors of multi-view images.

Experimental Settings We train our INS framework on NeRF-Synthetic [50] dataset and Local Light Field Fusion(LLFF) [49] dataset. NeRF-Synthetic consists of complex scenes with 360-degree views, where each scene has a central object with 100 inward-facing cameras distributed randomly on the upper hemisphere. Both rendered images and ground truth meshes are provided in NeRF-Synthetic dataset. LLFF dataset consists of forward-facing scenes, with fewer images. We implement INS on the same architecture and training strategy with the original NeRF [50]. λ_1, λ_2 and λ_3 are set as zero in the first 150k iterations and then set as 1e6, 1 and 1e6 in the following 50k iterations. The self-distilled density supervision depicted in Figure 3 is generated from the CIM with 150k iterations. Adam optimizer is adopted with learning rates of 0.0005. Hyper-parameters are carefully tuned via grid searches and the best configuration is

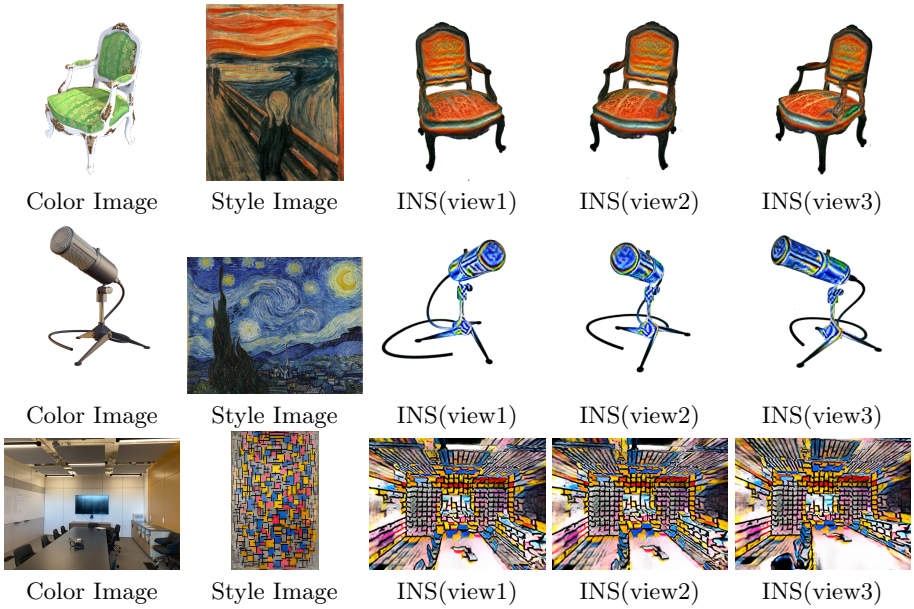

Fig. 6. Multi-view examples rendered by applying INS to the neural radiance field. The rendered scenes and objects show consistent stylized texture under different views. Please refer to the supplementary videos.

applied to all experiments. All experiments are trained on one NVIDIA RTX A6000 GPU. We retrain Style3D [16] in NeRF-synthetic and LLFF datasets using their provided code and setting. We train all methods using the same number of style images for fair comparisons.

Results In Figure 6, we show INS generates faithful and view-consistent results for new viewpoints, with rich textures across scenes and styles. We further compare INS with three state-of-the art methods, including 3D neural stylization [16], image-based stylization methods [34,38]. As is shown in Figure 7, We can see that stylizations from image-based methods produce noisy and inconsistent stylization as they transfer styles based on a single image. Method in [16] generates blur results as it still relies on convolution networks (a.k.a. hypernetwork) to generate the MLP weights for the subsequent volume rendering. Our proposed implicit neural stylization method is trained to preserve correct scene geometry as well as capture global context, generating view-consistent stylizations.

5.3 Stylization on Signed Distance Function

The original Signed Distance Functions (SDFs) only learns the 3D geometry from given inputs. Later works [74] extend it to reconstruct both 3D surface and

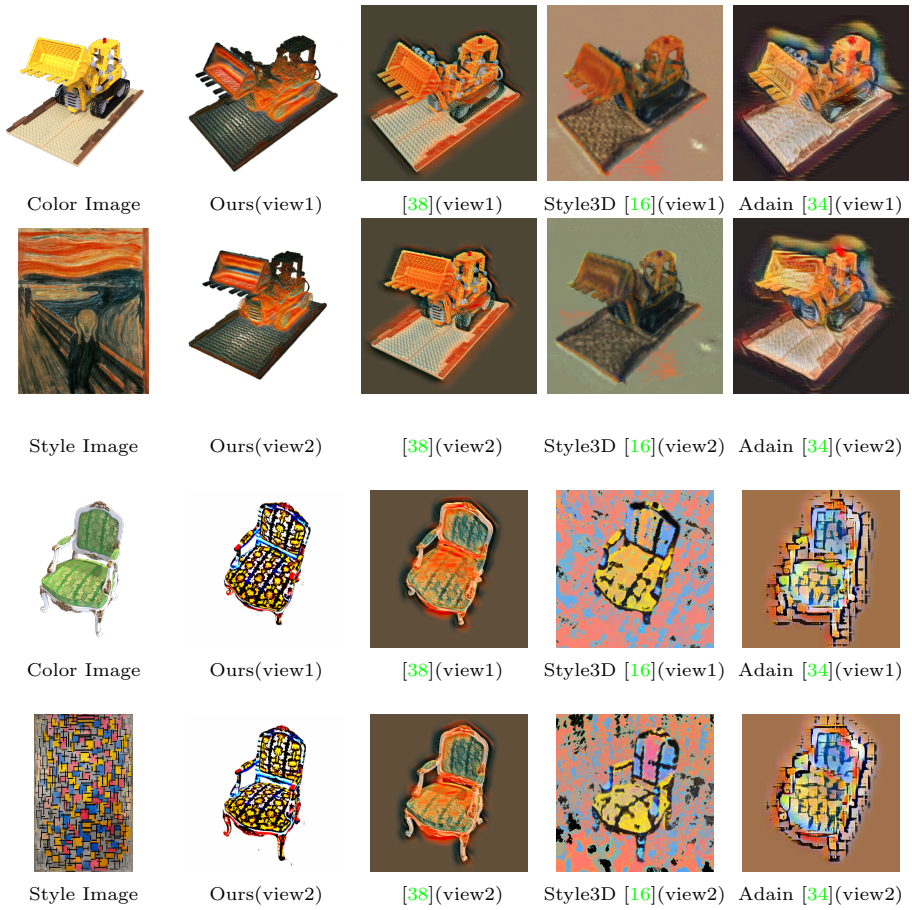

Fig. 7. Qualitative comparisons. We compare INS with several methods on novel view synthesis datasets. Three scenes with different style images are demonstrated, we can see our proposed INS method preserves the learned geometry better (e.g. with clean background) while achieving the desired style. Please refer to the supplementary videos.

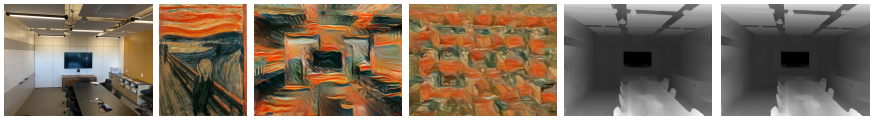

Fig. 8. Qualitative evaluation on the effectiveness of Sampling Stride (SS). We can see INS with a larger receptive field can produce better stylized images.

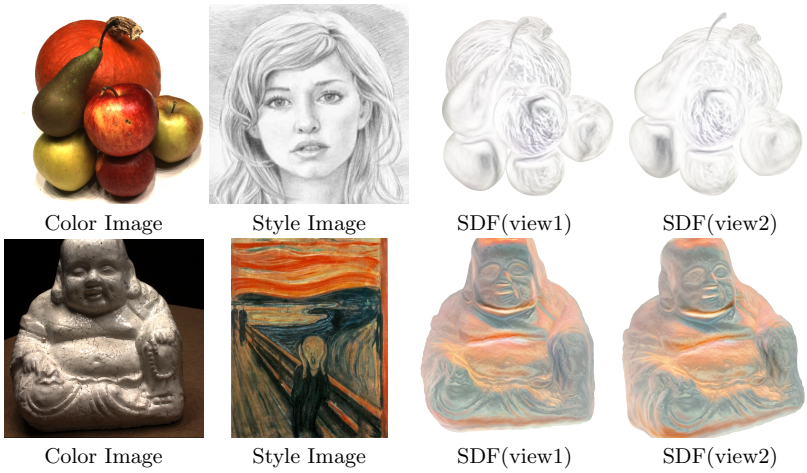

Fig. 9. Visualization results of applying INS to the Signed Distance Function. Given multi-view color images and style images, IDR [74] can learn the style statistics for the disentangled geometry and appearance.

appearance. We follow [74] to implement the implicit neural stylization framework. To encode style statistic onto SDFs, we project the learned textured SDF into multi-view images and implement our style loss on the rendered results. Similar to [74], we add another rendering network to the SDF network and a rendering network for reconstructing appearance and geometry representation, respectively. See the details in Section 3. In the experiments, we picked 2 scenes from the DTU dataset [1], where each scene consists of 50 to 100 images and object masks captured from different angles. Similar to NeRF, we pre-train the IDR model for chosen scenes by minimizing the loss between the ground truth image and the rendered result. Then both the SDF network and rendering network are jointly optimized the proposed framework from projected views. Note that due to IDR's architectural design, we are no longer able to impose self-distilled geometry consistency loss. Instead, we employ content/style loss in the masked region, similar to [74]. Besides, we observe that the SDF representation is more sensitive to parameter variations. To maintain intact geometries, we adjust the learning rate for the SDF network to 10^{-11} times smaller than the rendering network. As are shown in Figure 9, the visualizations of two-view SDF representation demonstrate that both the learned appearance and geometry have deformed to fit the given style statistics.

5.4 Conditional Style Interpolation

Input with style embedding with multiple style images, we can interpolate between style images to mix multiple styles with arbitrary weights. Specifically, we train INS on NeRF with two style images along with a two-dimensional one-hot vector as conditional code. After training, we can mix the style statistics

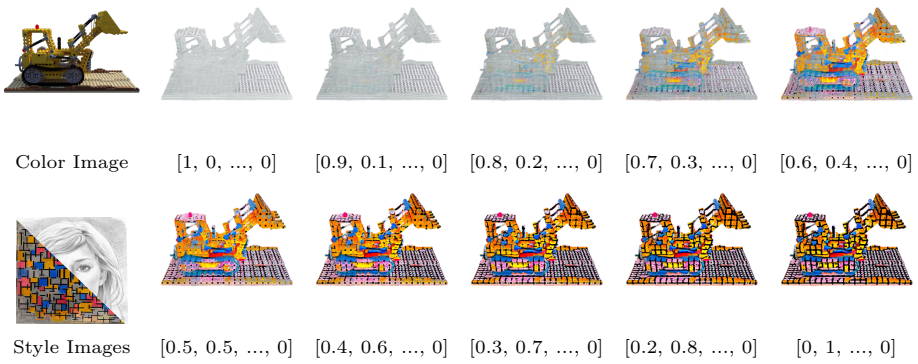

Fig. 10. Visualization of the mixture of two styles using different mixture weights. We can see the style smoothly transfers from one style to another. Please refer to the supplementary video for more detailed and smooth results.

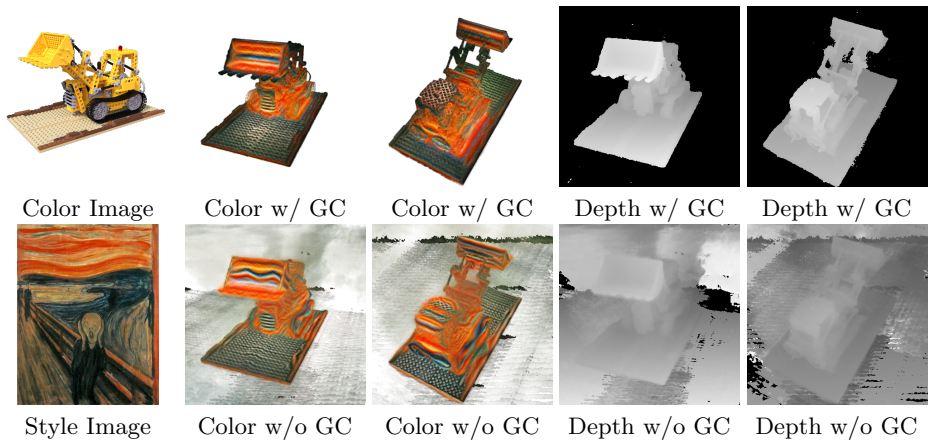

Fig. 11. Qualitative results of the self-distilled geometry consistency. "GC" denotes self-distilled Geometry Consistency loss. Both the rendered images and depth maps are shown to validate the effectiveness.

by using a weighted two-dimensional vector in the inference stage. As is shown in Figure 10, the synthesized results can smoothly transfer from the first style to the second style when we linearly mix the two style embeddings at inference time. The visualization of conditional style interpolation with more styles on 200 test views can be seen in the supplementary video.

5.5 Ablation Study

Effect of Self-distilled Geometry Consistency As mentioned earlier, geometry consistency has been applied to INS for NeRF to preserve density fidelity. To evaluate the effectiveness of the proposed regularizer, we visualize the front and

Table 1. Quantitative results for short-range and long-range consistency score. The consistency scores (the lower the better) are computed between stylized images at different viewpoints.

Metrics	Methods			
	Ours	Style3D [16]	Adain [34]	Perceptual [38]
Short-range Consistency	1.433	1.518	1.741	1.772
Long-range Consistency	2.159	2.290	2.679	2.495

back viewpoints of the synthesized color images and depth maps. As is shown in Figure 11, the proposed self-distilled geometry consistency learns a good tradeoff between stylization and clean geometry.

Should INS Learned with Larger Receptive Field? To investigate the effect by using the Sampling Stride (SS) Ray Sampling strategy, we compare with a ray sampling without sampling stride on neural radiance field stylization. For a fair comparison, we adopt the ray number of 64×64 in both settings. INS with sampling stride covers the content resolution of (64×s) × (64×s) where s indicates sampling strides depicted in Section 4.2 and here we set s=4. Figure 8 shows that INS with the strided sampling achieves significantly better visual results, as it results in a higher receptive field in perceiving content statistics.

Consistency Score We further adopt the video consistency metric [16,40] to measure the consistency between different views. Specifically, we compute the short-range and long-term consistency scores which compare adjacent frames and two far-away views. As shown in Table 1, INS produces the best view-consistency score among all methods. While Style3D [16] generates blurred and inconsistent results at insufficient training data [43], others [34,38] are unable to maintain view-consistency as they are single-view-based method.

6 Conclusions

In this work, we present the Unified Implicit Neural Stylization framework (INS) to stylize complex 2D/3D scenes using implicit function. We conduct a pilot study on different types of implicit representations, including 2D coordinate-based mapping function, Signed Distance Function, and Neural Radiance Field. Comprehensive experiments demonstrate that the proposed method yields photo-realistic images/videos with visually consistent stylized textures. One limitation of our work lies in the training efficiency issue, similar to most implicit representation, rendering a style scene requires several hours of training, precluding on-device training. Addressing this issue could become a future direction.

References

1. Aanæs, H., Jensen, R.R., Vogiatzis, G., Tola, E., Dahl, A.B.: Large-scale data for multiple-view stereopsis. International Journal of Computer Vision **120**(2), 153–168 (2016)
2. Attal, B., Laidlaw, E., Gokaslan, A., Kim, C., Richardt, C., Tompkin, J., O'Toole, M.: Törf: Time-of-flight radiance fields for dynamic scene view synthesis. Advances in neural information processing systems **34** (2021)
3. Atzmon, M., Haim, N., Yariv, L., Israelov, O., Maron, H., Lipman, Y.: Controlling neural level sets. Advances in Neural Information Processing Systems **32** (2019)
4. Atzmon, M., Lipman, Y.: Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2565–2574 (2020)
5. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5855–5864 (2021)
6. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. arXiv preprint arXiv:2111.12077 (2021)
7. Bergman, A., Kellnhofer, P., Wetzstein, G.: Fast training of neural lumigraph representations using meta learning. Advances in Neural Information Processing Systems **34** (2021)
8. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L., Tremblay, J., Khamis, S., et al.: Efficient geometry-aware 3d generative adversarial networks. arXiv preprint arXiv:2112.07945 (2021)
9. Chan, E.R., Monteiro, M., Kellnhofer, P., Wu, J., Wetzstein, G.: pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5799–5809 (2021)
10. Chen, D., Liao, J., Yuan, L., Yu, N., Hua, G.: Coherent online video style transfer. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1105–1114 (2017)
11. Chen, H., He, B., Wang, H., Ren, Y., Lim, S.N., Shrivastava, A.: Nerv: Neural representations for videos. Advances in Neural Information Processing Systems **34** (2021)
12. Chen, X., Zhang, Y., Wang, Y., Shu, H., Xu, C., Xu, C.: Optical flow distillation: Towards efficient and stable video style transfer. In: European Conference on Computer Vision. pp. 614–630. Springer (2020)
13. Chen, Y., Liu, S., Wang, X.: Learning continuous image representation with local implicit image function. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8628–8638 (2021)
14. Chen, Y., Liu, S., Wang, X.: Learning continuous image representation with local implicit image function. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8628–8638 (2021)
15. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948 (2019)
16. Chiang, P.Z., Tsai, M.S., Tseng, H.Y., Lai, W.S., Chiu, W.C.: Stylizing 3d scene via implicit representation and hypernetwork. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1475–1484 (2022)

17. DeVries, T., Bautista, M.A., Srivastava, N., Taylor, G.W., Susskind, J.M.: Unconstrained scene generation with locally conditioned radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14304–14313 (2021)
18. Drebin, R.A., Carpenter, L., Hanrahan, P.: Volume rendering. ACM Siggraph Computer Graphics **22**(4), 65–74 (1988)
19. Dupont, E., Goliński, A., Alizadeh, M., Teh, Y.W., Doucet, A.: Coin: Compression with implicit neural representations. arXiv preprint arXiv:2103.03123 (2021)
20. Gao, C., Shih, Y., Lai, W.S., Liang, C.K., Huang, J.B.: Portrait neural radiance fields from a single image. arXiv preprint arXiv:2012.05903 (2020)
21. Gatys, L., Ecker, A.S., Bethge, M.: Texture synthesis using convolutional neural networks. Advances in neural information processing systems **28** (2015)
22. Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)
23. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2414–2423 (2016)
24. Genova, K., Cole, F., Vlasic, D., Sarna, A., Freeman, W.T., Funkhouser, T.: Learning shape templates with structured implicit functions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7154–7164 (2019)
25. Gong, X., Huang, H., Ma, L., Shen, F., Liu, W., Zhang, T.: Neural stereoscopic image style transfer. In: Proceedings of the European Conference on Computer Vision (ECCV) (September 2018)
26. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. arXiv preprint arXiv:2002.10099 (2020)
27. Gu, J., Liu, L., Wang, P., Theobalt, C.: Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. arXiv preprint arXiv:2110.08985 (2021)
28. Han, J., Jentzen, A., Weinan, E.: Solving high-dimensional partial differential equations using deep learning. Proceedings of the National Academy of Sciences **115**(34), 8505–8510 (2018)
29. Hao, Z., Mallya, A., Belongie, S., Liu, M.Y.: Gancraft: Unsupervised 3d neural rendering of minecraft worlds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14072–14082 (2021)
30. Hedman, P., Srinivasan, P.P., Mildenhall, B., Barron, J.T., Debevec, P.: Baking neural radiance fields for real-time view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5875–5884 (2021)
31. Höllein, L., Johnson, J., Nießner, M.: Stylemesh: Style transfer for indoor 3d scene reconstructions. arXiv preprint arXiv:2112.01530 (2021)
32. Huang, H., Wang, H., Luo, W., Ma, L., Jiang, W., Zhu, X., Li, Z., Liu, W.: Real-time neural style transfer for videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 783–791 (2017)
33. Huang, H.P., Tseng, H.Y., Saini, S., Singh, M., Yang, M.H.: Learning to stylize novel views. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13869–13878 (2021)
34. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE international conference on computer vision. pp. 1501–1510 (2017)
35. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. ACM Transactions on Graphics (ToG) **35**(4), 1–11 (2016)

36. Jiang, C., Sud, A., Makadia, A., Huang, J., Nießner, M., Funkhouser, T., et al.: Local implicit grid representations for 3d scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6001–6010 (2020)
37. Jiang, Y., Ji, D., Han, Z., Zwicker, M.: Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1251–1261 (2020)
38. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
39. Kato, H., Ushiku, Y., Harada, T.: Neural 3d mesh renderer. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3907–3916 (2018)
40. Lai, W.S., Huang, J.B., Wang, O., Shechtman, E., Yumer, E., Yang, M.H.: Learning blind video temporal consistency. In: Proceedings of the European conference on computer vision (ECCV). pp. 170–185 (2018)
41. Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A.: Fourier neural operator for parametric partial differential equations. arXiv preprint arXiv:2010.08895 (2020)
42. Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y.: Neural volumes: Learning dynamic renderable volumes from images. arXiv preprint arXiv:1906.07751 (2019)
43. Lorraine, J., Duvenaud, D.: Stochastic hyperparameter optimization through hypernetworks. arXiv preprint arXiv:1802.09419 (2018)
44. Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., Duckworth, D.: Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7210–7219 (2021)
45. Meng, Q., Chen, A., Luo, H., Wu, M., Su, H., Xu, L., He, X., Yu, J.: Gnerf: Gan-based neural radiance field without posed camera. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6351–6361 (2021)
46. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4460–4470 (2019)
47. Michalkiewicz, M., Pontes, J.K., Jack, D., Baktashmotlagh, M., Eriksson, A.: Implicit surface representations as layers in neural networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4743–4752 (2019)
48. Mildenhall, B., Hedman, P., Martin-Brualla, R., Srinivasan, P., Barron, J.T.: Nerf in the dark: High dynamic range view synthesis from noisy raw images. arXiv preprint arXiv:2111.13679 (2021)
49. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG) **38**(4), 1–14 (2019)
50. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)
51. Mu, F., Wang, J., Wu, Y., Li, Y.: 3d photo stylization: Learning to generate stylized novel views from a single image. arXiv preprint arXiv:2112.00169 (2021)

52. Niemeyer, M., Geiger, A.: Giraffe: Representing scenes as compositional generative neural feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11453–11464 (2021)
53. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Occupancy flow: 4d reconstruction by learning particle dynamics. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5379–5389 (2019)
54. Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., Geiger, A.: Texture fields: Learning texture representations in function space. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4531–4540 (2019)
55. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174 (2019)
56. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174 (2019)
57. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Martin-Brualla, R., Seitz, S.M.: Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. arXiv preprint arXiv:2106.13228 (2021)
58. Peng, S., Niemeyer, M., Mescheder, L., Pollefeys, M., Geiger, A.: Convolutional occupancy networks. In: European Conference on Computer Vision. pp. 523–540. Springer (2020)
59. Ruder, M., Dosovitskiy, A., Brox, T.: Artistic style transfer for videos. In: German conference on pattern recognition. pp. 26–36. Springer (2016)
60. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2304–2314 (2019)
61. Schwarz, K., Liao, Y., Niemeyer, M., Geiger, A.: Graf: Generative radiance fields for 3d-aware image synthesis. Advances in Neural Information Processing Systems **33**, 20154–20166 (2020)
62. Shen, S., Wang, Z., Liu, P., Pan, Z., Li, R., Gao, T., Li, S., Yu, J.: Non-line-of-sight imaging via neural transient fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
63. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
64. Sitzmann, V., Chan, E., Tucker, R., Snavely, N., Wetzstein, G.: Metasdf: Meta-learning signed distance functions. Advances in Neural Information Processing Systems **33**, 10136–10147 (2020)
65. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. Advances in Neural Information Processing Systems **33**, 7462–7473 (2020)
66. Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.: Implicit neural representations with periodic activation functions. Advances in Neural Information Processing Systems **33**, 7462–7473 (2020)
67. Sitzmann, V., Rezchikov, S., Freeman, W.T., Tenenbaum, J.B., Durand, F.: Light field networks: Neural scene representations with single-evaluation rendering. arXiv preprint arXiv:2106.02634 (2021)

68. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. Advances in Neural Information Processing Systems **32** (2019)
69. Sun, Y., Liu, J., Xie, M., Wohlberg, B., Kamilov, U.S.: Coil: Coordinate-based internal learning for imaging inverse problems. arXiv preprint arXiv:2102.05181 (2021)
70. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. Advances in Neural Information Processing Systems **33**, 7537–7547 (2020)
71. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.S.: Texture networks: Feed-forward synthesis of textures and stylized images. In: ICML. vol. 1, p. 4 (2016)
72. Xu, Y., Qiu, X., Zhou, L., Huang, X.: Improving bert fine-tuning via self-ensemble and self-distillation. arXiv preprint arXiv:2002.10345 (2020)
73. Yanai, K., Tanno, R.: Conditional fast style transfer network. In: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval. pp. 434–437 (2017)
74. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. Advances in Neural Information Processing Systems **33**, 2492–2502 (2020)
75. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. Advances in Neural Information Processing Systems **33**, 2492–2502 (2020)
76. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587 (2021)
77. Zeng, K., Zhao, M., Xiong, C., Zhu, S.C.: From image parsing to painterly rendering. ACM Trans. Graph. **29**(1), 2–1 (2009)
78. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
79. Zhang, Y., van Rozendaal, T., Brehmer, J., Nagel, M., Cohen, T.: Implicit neural video compression. arXiv preprint arXiv:2112.11312 (2021)
80. Zhao, M., Zhu, S.C.: Customizing painterly rendering styles using stroke processes. In: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on non-photorealistic animation and rendering. pp. 137–146 (2011)
81. Zhong, E.D., Bepler, T., Berger, B., Davis, J.H.: Cryodrgn: reconstruction of heterogeneous cryo-em structures using neural networks. Nature Methods **18**(2), 176–185 (2021)
82. Zhou, P., Xie, L., Ni, B., Tian, Q.: Cips-3d: A 3d-aware generator of gans based on conditionally-independent pixel synthesis. arXiv preprint arXiv:2110.09788 (2021)