RealFusion
360° Reconstruction of Any Object from a Single Image

Luke Melas-Kyriazi Iro Laina Christian Rupprecht Andrea Vedaldi

Visual Geometry Group, Department of Engineering Science, University of Oxford
{lukemk,iro,chrisr,vedaldi}@robots.ox.ac.uk
https://lukemelas.github.io/realfusion

Figure 1. **RealFusion** generates a full 360° reconstruction of any object given a *single image* of it (left column). It does so by leveraging an existing diffusion-based 2D image generator. From the given image, it synthesizes a prompt that causes the diffusion model to "dream up" other views of the object. It then extracts a neural radiance field from the original image and the diffusion model-based prior, thereby reconstructing the object in full. Both appearance and geometry are reconstructed faithfully and extrapolated in a plausible manner.

Abstract

We consider the problem of reconstructing a full 360° photographic model of an object from a single image of it. We do so by fitting a neural radiance field to the image, but find this problem to be severely ill-posed. We thus take an off-the-self conditional image generator based on diffusion and engineer a prompt that encourages it to "dream up" novel views of the object. Using the recent DreamFusion method, we fuse the given input view, the conditional prior, and other regularizers into a final, consistent reconstruction. We demonstrate state-of-the-art reconstruction results on benchmark images when compared to prior methods for monocular 3D reconstruction of objects. Qualitatively, our reconstructions provide a faithful match of the input view and a plausible extrapolation of its appearance and 3D shape, including to the side of the object not visible in the image.

1. Introduction

We consider the problem of obtaining a 360° photographic reconstruction of *any* object given a *single image* of it. The challenge is that a single image *does not* contain sufficient information for 3D reconstruction. Without access to multiple views, an image only provides weak evidence about the 3D shape of the object, and only for one side of it. Even so, there is proof that this task *can* be solved: any skilled 3D artist can take a picture of almost any object and, given sufficient time and effort, create a plausible 3D model of it. The artist can do so by tapping into their vast knowledge of the natural world and the objects it contains, making up for the information missing from the image.

Hence, monocular 3D reconstruction requires combining

visual geometry with a powerful statistical model of the 3D world. Diffusion-based 2D image generators like DALL-E 2 [30], Imagen [35], and Stable Diffusion [33] are able to generate high-quality images from ambiguous inputs such as text, showing that powerful priors for 2D images can be learned. However, extending them to 3D is not easy because, while one can access billions of 2D images for training [36], the same cannot be said for 3D data.

A simpler approach is to extract or *distill* 3D information from an existing 2D generator. A 2D image generator can in fact be used to sample or validate multiple views of a given object, which can then be used to perform 3D reconstruction. This idea was already demonstrated with GAN-based generators for simple data like faces and synthetic objects [2, 6, 8, 24, 25, 47]. Better 2D generators have since resulted in better results, culminating in methods such as DreamFusion [27], which can produce high-quality 3D models from an existing 2D generator and text.

In this paper, we port distillation approaches from text-based generation to monocular 3D reconstruction. This is not a trivial change because conditioning generation on an image provides a much more fine-grained specification of the object than text. This in turn requires the 2D diffusion model to hallucinate new views of a *specific object* instead of *some object* of a given type. The latter is difficult because the coverage of generator models is limited [1], meaning that not every version of an object is captured well by the model. We find empirically that this is a key problem.

We address this issue by introducing RealFusion, a new method for 3D reconstruction from a single image. We express the object's 3D geometry and appearance by means of a neural radiance field. Then, we fit the radiance field to the given input image by minimizing the usual rendering loss. At the same time, we sample random other views of the object, and constrain them with the diffusion prior, using a technique similar to DreamFusion.

We find that, due to the coverage issue, this idea does not work well out of the box, but can be improved via adequately conditioning the 2D diffusion model. The idea is to configure the prior to "dream up" or sample images that may *plausibly constitute other views of the given object*. We do so by automatically engineering the diffusion prompt from random augmentations of the given image. In this manner, the diffusion model provides sufficiently strong constraints to allow meaningful 3D reconstruction.

In addition to setting the prompt correctly, we also add some regularizers: shading the underlying geometry and randomly dropping out texture (also similar to DreamFusion), smoothing the normals of the surface, and fitting the model in a coarse-to-fine fashion, capturing first the overall structure of the object and only then the fine-grained details. We also focus on efficiency and base our model on Instant-NGP [23]. In this manner, we achieve reconstructions in the span of hours instead of days if we were to adopt traditional MLP-based NeRF models.

We assess our approach by using random images captured in the wild as well as existing benchmark datasets. Note that we do *not* train a fully-fledged 2D-to-3D model and we are *not* limited to specific object categories; rather, we perform reconstruction on an image-by-image basis using a pretrained 2D generator as a prior. Nonetheless, we can surpass quantitatively and qualitatively previous single-image reconstructors, including Shelf-Supervised Mesh Prediction [50], which uses supervision tailored specifically for 3D reconstruction.

Qualitatively, we obtain plausible 3D reconstructions that are a good match for the provided input image (Fig. 1). Our reconstructions are not perfect, as the diffusion prior clearly does its best to explain the available image evidence but cannot always match all the details. Even so, we believe that our results convincingly demonstrate the viability of this approach and trace a path for future improvements.

To summarize, we make the following **contributions**: (1) We propose RealFusion, a method that can extract from a single image of an object a 360° photographic 3D reconstruction without assumptions on the type of object imaged or 3D supervision of any kind; (2) We do so by leveraging an existing 2D diffusion image generator via a new single-image variant of textual inversion; (3) We also introduce new regularizers and provide an efficient implementation using InstantNGP; (4) We demonstrate state-of-the-art reconstruction results on a number of in-the-wild images and images from existing datasets when compared to alternative approaches.

2. Related work

Radiance fields. The problem of reconstructing photometry and geometry together has been dramatically revitalized by the introduction of neural radiance fields (RFs). NeRF [20] in particular noticed that a coordinate MLP provides a compact and yet expressive representation of 3D fields, and can be used to model RFs with great effectiveness. Many variants of NeRF-like models have since appeared. For instance, some [18, 41, 43] use sign distance functions (SDFs) to recover cleaner geometry. These approaches assume that dozens if not hundreds of views of each scene are available for reconstruction. Here, we use radiance fields for single-image reconstruction by using a diffusion model to "dream up" the missing views.

Few-view reconstruction. Many authors have attempted to improve the statistical efficiency of NeRF-like models by learning or incorporating various kinds of priors. Most of these approaches train deep networks specifically for the goal of inferring NeRFs from a small number of views; examples include IBRNet [44], NeRF-WCE [9], Pixel-

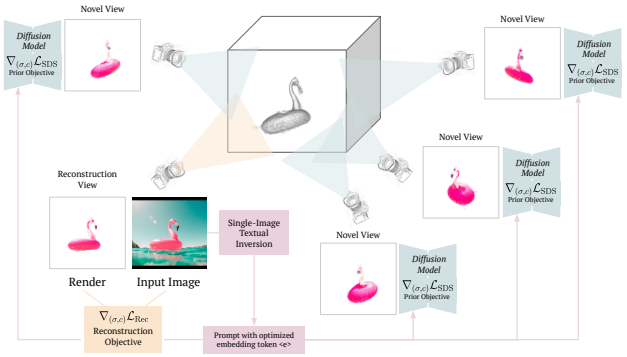

Figure 2. **Method diagram.** Our method optimizes a neural radiance field using two objectives simultaneously: a reconstruction objective and a prior objective. The reconstruction objective ensures that the radiance field resembles the input image from a specific, fixed view. The prior objective uses a large pre-trained diffusion model to ensure that the radiance field looks like the given object from randomly sampled novel viewpoints. The key to making this process work well is to condition the diffusion model on a prompt with a custom token $\langle \mathbf{e} \rangle$, which is generated prior to reconstruction using single-image textual inversion.

NeRF [51], NeRFormer [31], and ViewFormer [16]. These models still generally require more than one input view at test time and multi-view data for training. Closer to our work, NeRF-on-a-Diet [13] reduces the number of images needed for NeRF optimization by generating random views and measuring their "semantic compatibility" via CLIP embeddings [28], but it also still requires several input views.

Single-view reconstruction. Some authors have attempted to recover full 3D representations from single images, but this generally requires multi-view data for training, as well as learning models that are specific to a specific object category. 3D-R2N2 [4], Pix2Vox [48], and LegoFormer [49] learn to reconstruct volumetric representation of simple objects, mainly from synthetic data like ShapeNet [3]. More recently, CodeNeRF [14] predicts a full radiance field and AutoRF [22] learns a similar autoencoder specifically for cars. Shelf-Supervised Mesh Prediction [50] and SS3D [42] learn to reconstruct a broad set of object categories by refining a mesh extracted from a coarse predicted volume. The former learns independent models for each cateogry, whereas the latter unites them into a unified model via distillation.

Extracting 3D models from 2D generators. Several authors have proposed to extract 3D models from 2D image generators, originally using GANs [2, 6, 8, 24, 25, 47]. Also related to our work, CLIP-Mesh [15] and Dream Fields [12] use a pre-trained image-text model [28] to condition 3D generation on textual prompts and do not require 3D supervision. Recently, DreamFusion [27] builds on this idea using a diffusion model as a prior. These models, however, are used as either pure generators or generators conditioned on vague cues such as class identity or textual descriptions. Here, we build on the DreamFusion approach, extending the idea to single-view reconstruction.

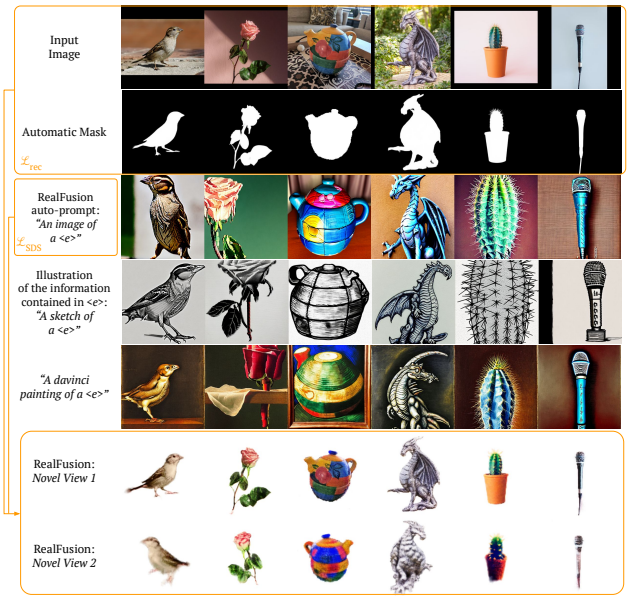

Figure 3. **Examples demonstrating the level of detail of information captured by the optimized embedding $\langle \mathbf{e} \rangle$.** Rows 1-2 show input images and masks. The images are used to optimize $\langle \mathbf{e} \rangle$ via our single-image textual inversion process. Rows 3-5 show examples of 2D images generated using $\langle \mathbf{e} \rangle$ in new prompts, which we hope demonstrate the type of information encoded in $\langle \mathbf{e} \rangle$. Rows 6-7 show RealFusion's output, optimized using the prompt "An image of a $\langle \mathbf{e} \rangle$".

Recently, the authors of [46] have proposed to directly generate multiple 2D views of an object, which can then be reconstructed in 3D using a NeRF-like model. This is also reminiscent of our approach, but their model requires multi-view data for training, is only tested on synthetic data, and requires to explicitly sample multiple views for reconstruction (in our case they remain implicit).

Diffusion models. Diffusion denoising probabilistic models are a class of generative models based on iteratively reversing a Markovian noising process. Early works formulated the problem as learning a variational lower bound [10], or framed it as optimizing a score-based generative model [37, 38] or as the discretization of a continuous stochastic process [39]. Recent improvements include the use of faster and deterministic sampling [10, 19, 45], class-conditional models [5, 38], text-conditional models [26], and modeling in latent space [34].

3. Method

We review the background and notation in Sec. 3.1, and then discuss our RealFusion method in Sec. 3.2.

3.1. Radiance fields, diffusion and distillation

Radiance fields. A *radiance field* (RF) is a pair of functions $(\sigma(\boldsymbol{x}), c(\boldsymbol{x}))$ mapping a 3D point $\boldsymbol{x} \in \mathbb{R}^3$ to an opacity value $\sigma(\boldsymbol{x}) \in \mathbb{R}_+$ and a color value $c(\boldsymbol{x}) \in \mathbb{R}^3$. The RF is called *neural* when these two functions are implemented by a neural network.

The RF represents the shape and appearance of an object. In order to generate an image of it, one *renders* the RF using the emission-absorption model [21]. This can be modeled as a function $I = \mathcal{R}(\sigma, c, \pi) \in \mathbb{R}^{3\times H\times W}$ that extracts a color image I from the radiance field (σ, c) given a viewpoint $\pi \in \mathrm{SE}(3)$. The rendering function $R(\sigma, c, \pi)$ is differentiable, which allows training the model by means of a standard optimizer. Specifically, the RF is fitted to a dataset $\mathcal{D} = \{(I, \pi)\}$ of images I with known camera parameters by minimizing the L_2 image reconstruction error

$$\mathcal{L}_{\text{rec}}(\sigma, c; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(I,\pi)\in\mathcal{D}} \|I - R(\sigma, c, \pi)\|^2. \quad (1)$$

In order to obtain good-quality results, one typically requires a dataset of dozens or hundreds of views.

Here, we consider the case in which we are given *exactly one* input image I_0 corresponding to some (unknown) camera π_0. In this case, we can also assume *any* standard viewpoint π_0 for that single camera. Optimizing Eq. (1) with a single training image leads to severe over-fitting: it is straightforward to find a pair (σ, c) that has zero loss and yet does not capture any sensible 3D model of the object. Below we will leverage a pre-trained 2D image prior to (implicitly) dream up novel views of the object and provide the missing information for 3D reconstruction.

Diffusion models. A *diffusion model* draws a sample from a probability distribution $p(I)$ by inverting a process that gradually adds noise to the image I. The diffusion process is associated with a noising schedule $\{\alpha_t \in (0, 1)\}_{t=1}^T$, which defines how much noise is added at each time step. The noisy version of sample I at time t can then be written $I_t = \alpha_t I + \sigma_t \epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, is a sample from a Gaussian distribution and $\sigma_t^2 = 1 - \alpha_t^2$. One then learns a denoising neural network $\hat{\epsilon} = \Phi(I_t; t)$ that takes as input the noisy image I_t and the noise level t and predicts the noise component ϵ. Iterated applications of the network generate $I = I_0$ from $I_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

The diffusion model is trained on large collections $\mathcal{D}' = \{I\}$ of images by minimizing the loss

$$\mathcal{L}_{\text{diff}}(\Phi; \mathcal{D}') = \tfrac{1}{|\mathcal{D}'|} \sum_{I\in\mathcal{D}'} ||\Phi(\alpha_t I + \sigma_t \epsilon; t) - \epsilon||^2. \quad (2)$$

This model can be easily extended to draw samples from a distribution $p(\boldsymbol{x}|\boldsymbol{e})$ conditioned on a *prompt* $\boldsymbol{e}$. Conditioning on the prompt is obtained by adding $\boldsymbol{e}$ as an additional input of the network Φ, and the strength of conditioning can be controlled via classifier-free guidance [5, 11].

DreamFusion and Score Distillation Sampling (SDS). Given a 2D diffusion model $p(I|\boldsymbol{e})$ and a prompt $\boldsymbol{e}$, DreamFusion extracts from it a 3D rendition of the corresponding concept, represented by a RF (σ, c). It does so by randomly sampling a camera parameter π, rendering a corresponding view I, assessing the likelihood of the view based on the model $p(I|\boldsymbol{e})$, and updating the RF to increase the likelihood of the generated view based on the model.

In practice, DreamFusion uses the denoising network as a frozen critic and takes a gradient step

$$\nabla_{(\sigma,c)} \mathcal{L}_{\text{SDS}}(\sigma, c; \pi, \boldsymbol{e}, t) = E_{t,\epsilon}\Big[w(t)(\Phi(\alpha_t I + \sigma_t \epsilon;\, t, \boldsymbol{e}) - \epsilon) \cdot \frac{\partial I}{\partial(\sigma, c)}\Big], \quad (3)$$

where $I = R(\sigma, c, \pi)$ is the image rendered from a randomly-sampled viewpoint π and $w(t)$ is a timestep-dependent weighting function. This process is called *Score Distillation Sampling* (SDS).

One final aspect of DreamFusion is essential for understanding our contribution in the following section: DreamFusion finds that it is necessary to use classifier-free guidance [5] with a very high guidance weight of 100, much larger than one would use for image sampling, in order to obtain good 3D shapes. As a result, the generations tend to have limited diversity; they produce only the most likely objects for a given prompt, which is incompatible with our goal of reconstructing any given object.

3.2. RealFusion

Our goal is to reconstruct a 3D model of the object contained in a single image I_0, utilizing the prior captured in the diffusion model Φ to make up for the missing information. We do this by optimizing a radiance field using two simultaneous objectives: (1) a reconstruction objective (Eq. (1)) from a fixed viewpoint, and (2) a SDS-based prior objective (Eq. (3)) on novel views randomly sampled at each iteration. Figure 2 provides a diagram of the method.

Single-image textual inversion as a substitute for alternative views. The most important component of our method is the use of single-image textual inversion as a substitute for alternative views. Ideally, we would like to condition our reconstruction process on multi-view images of the object in I_0, *i.e.*, on samples from $p(I|I_0)$.

Since these images are not available, our idea is to engineer a text prompt $\boldsymbol{e}^{(I_0)}$ specifically for image I_0 as a proxy for this multi-view information, *i.e.*, as an approximation of $p(I|I_0)$. We do so by generating random augmentations $h(I_0)$, $h \in H$ of the input image, which serve as pseudo-alternative-views. We use these augmentations as a mini-dataset $\mathcal{D}' = \{h(I_0)\}_{h\in H}$ and optimize the diffusion loss

Figure 4. **Qualitative results.** RealFusion reconstructions from a single input view. Each pair of columns shows the textured object and the underlying 3D shape, as a shaded surface. Different pairs of columns show different viewpoints.

$\mathcal{L}_{\text{diff}}(\Phi(\cdot; \boldsymbol{e}^{(I_0)}))$ of Eq. (2) with respect to the prompt $\boldsymbol{e}^{(I_0)}$, while freezing all other text embeddings and model parameters.

In practice, our prompt is derived automatically from the template "an image of a $\langle$**e**$\rangle$", where "$\langle$**e**$\rangle$" ($= \boldsymbol{e}^{(I_0)}$) is a new optimizable token introduced to the vocabulary of the text encoder of our diffusion model (see the supplementary material for details). A caption or description of I_0 is not needed. Our optimization procedure mirrors and generalizes the recently-proposed textual-inversion method of [7]. Differently from [7], we work in the single-image setting and utilize image augmentations for training rather than multiple views.

To help convey the intuition behind $\langle$**e**$\rangle$, consider an attempt at reconstructing an image of a fish using the generic text prompt "An image of a fish" with losses (2) and (3). In our experience, this often produces a reconstruction which looks like the input fish from the input viewpoint, but looks like some *different, more-generic* fish from the backside. By contrast, using the prompt "An image of a $\langle$**e**$\rangle$", the reconstruction resembles the input fish from all angles. An example of exactly this case is shown in Fig. 7, and Fig. 3 demonstrates the amount of detail captured in the embedding $\langle$**e**$\rangle$.

Coarse-to-fine training. We use the diffusion prior described above to optimize an RF in a coarse-to-fine manner. Before describing our coarse-to-fine training methodology, we first briefly introduce the underlying RF model, *i.e.*, InstantNGP [23]. InstantNGP is a grid-based model which stores features at the vertices of a set of feature grids $\{G_i\}_{i=1}^{L}$ at multiple resolutions. The resolution is chosen to be a geometric progression between the coarsest and finest resolutions, and feature grids are trained simultaneously.

We choose InstantNGP over a conventional MLP due to its computational efficiency. However, InstantNGP tends to produce small irregularities on the surface of the object. We find that training in a coarse-to-fine manner helps to alleviate these issues: for the first half of training we only optimize the lower-resolution feature grids $\{G_i\}_{i=1}^{L/2}$, and then in the second half of training we optimize all feature grids $\{G_i\}_{i=1}^{L}$. Using this strategy, we obtain the benefits of both efficient training and high-quality results.

Figure 5. **Qualitative comparison with prior work.** We show the results of our method and the category-level method of [50] on real-world images from the CO3D dataset [31]. Each pair of rows show two novel views produced by [50] and our method. For [50], we use category-specific models for each CO3D category (in this case, motorcycles, cups, and backpacks). Despite not requiring any category-specific information, our method is able to reconstruct objects at a higher level of detail than [50].

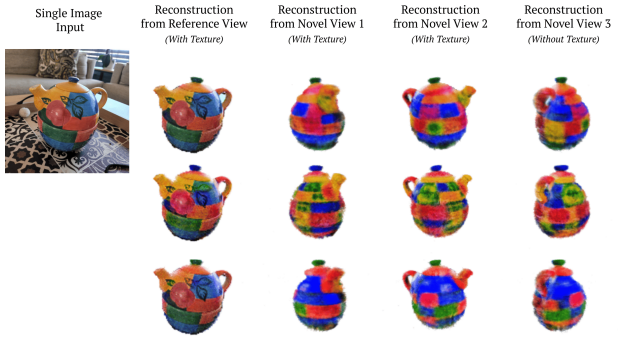

Figure 6. **Diversity of 3D reconstructions from a single image.** Above, we see our method's ability to generate a diverse set of object reconstructions given the same input image. In particular, the method produces different textures on the backsides of the generated objects, despite all objects matching the input image from the reference view.

Normal vector regularization. We introduce a regularization term to encourage our geometry to have smoothly varying normal vectors. Notably, we perform this regularization in *2D* rather than in 3D because we found that it reduces the variance of the regularizer and improves results.

At each iteration, in addition to computing RGB and opacity values, we also compute normals for each point along the ray and aggregate these via the ray marching equation to obtain normals[1] $N \in \mathcal{R}^{3\times H\times W}$. Our loss is:

$$\mathcal{L}_{\text{normals}} = \|N - \text{stopgrad}(\text{blur}(N, k))\|^2 \qquad (4)$$

[1]Normals may be computed either by taking the gradient of the density field or by using finite differences. We found that using finite differences works well in practice.

	F-score		*CLIP-similarity*	
	[50]	Ours	[50]	Ours
Backpack	7.58	**12.22**	0.72	**0.74**
Chair	8.26	**10.23**	0.65	**0.76**
Motorcycle	8.66	**8.72**	0.69	**0.70**
Orange	6.27	**10.16**	0.71	**0.74**
Skateboard	**7.74**	5.89	**0.74**	**0.74**
Teddybear	**12.89**	10.08	0.73	**0.82**
Vase	6.30	**9.72**	0.69	**0.71**
Mean	8.24	**9.58**	0.70	**0.74**

Table 1. **Quantitative comparison.** We compare our method with Shelf-Supervised [50] on seven object categories. The F-score and CLIP-similarity metrics are designed to measure the quality of reconstruction shape and appearance, respectively. For both metrics, higher is better. Metrics are averaged over three images per category. Our method outperforms [50] in aggregate, despite the fact that [50] uses a *different category-specific model* for each category.

where stopgrad is a stop-gradient operation and $\text{blur}(\cdot, k)$ is a Gaussian blur with kernel size k (we use $k = 9$).

Mask loss. In addition to the input image, our model also utilizes a mask of the object that one wishes to reconstruct. In practice, we use an off-the-shelf image matting model to obtain this mask for all images. The loss is simply the L^2 norm of the difference between the rendered opacities from the fixed reference viewpoint $\hat{M} = \mathcal{R}(\sigma, \pi_0) \in \mathcal{R}^{H\times W}$ and the object mask M: $\mathcal{L}_{\text{mask}} = \|\hat{M} - M\|^2$

Our final objective then consists of four terms:

$$\nabla_{\sigma,c}\mathcal{L} = \nabla\mathcal{L}_{\text{SDS}} + \lambda_{\text{normals}} \cdot \nabla\mathcal{L}_{\text{normals}} + \lambda_{\text{image}} \cdot \nabla\mathcal{L}_{\text{image}} + \lambda_{\text{mask}} \cdot \nabla\mathcal{L}_{\text{mask}} \qquad (5)$$

where the first two terms are regularizers and the remaining ones are data terms.

4. Experiments

4.1. Implementation details

Regarding hyperparameters, we use nearly the same set of hyper-parameters for all experiments—there is no per-scene hyper-parameter adjustment.[2]. For our diffusion model prior, we employ the open-source *Stable Diffusion* model [34] trained on the LAION [36] dataset. At each optimization step, we first render from the reconstruction camera and compute our reconstruction losses $\mathcal{L}_{\text{image}}$ and $\mathcal{L}_{\text{mask}}$. We then render from a randomly sampled camera to obtain a novel view and use this view for $\mathcal{L}_{\text{SDS}}$ and $\mathcal{L}_{\text{normals}}$. We use $\lambda_{\text{image}} = 5.0$, $\lambda_{\text{mask}} = 0.5$, and $\lambda_{\text{normals}} = 0.5$.

[2]There is one small exception to this rule, which is that for a small number of images where the camera angle was clearly at an angle higher than 15° above the horizontal plane, we used a camera angle of 30 or 40°.

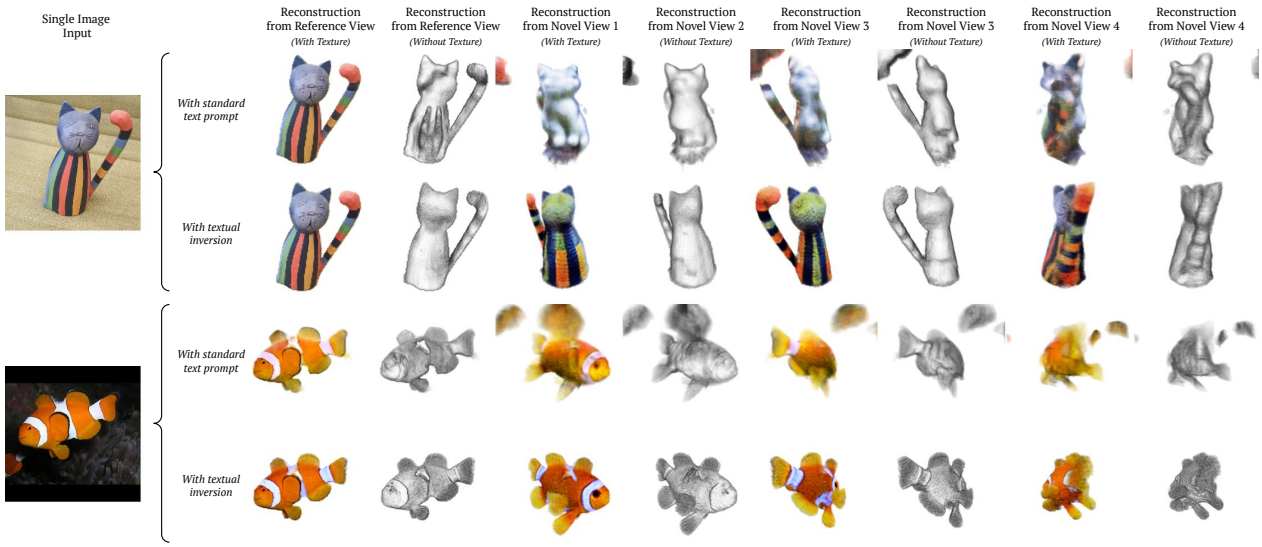

Figure 7. **A visualization of the effect of single-image textual inversion on reconstruction quality.** In each pair of rows, the top row shows the result of utilizing a standard text prompt for our diffusion-model-based loss (*e.g.*, "An image of a statue of a cat"). The bottom row shows the result of utilizing a text prompt optimized for the input image in a fully-automatic manner; this textual inversion process dramatically improves object reconstruction.

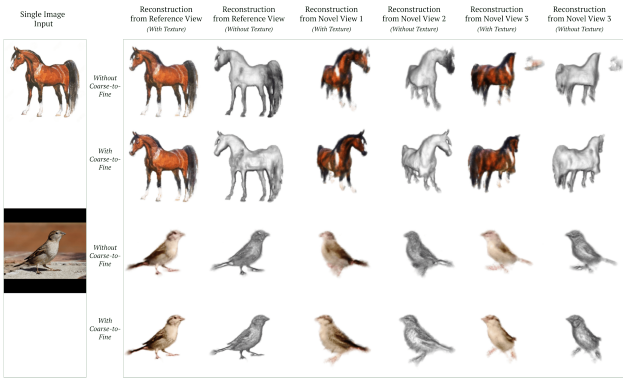

Figure 8. **Effect of coarse-to-fine training.** The top row of each pair is generated by optimizing all levels of a multi-resolution 3D feature grid from the first optimization step, whereas he bottom row is optimized in a coarse-to-fine manner.

Figure 9. **Effect of normal smoothness on reconstruction quality.** Each pair of rows show the reconstruction without and with the normal smoothness regularization term (4). The regularizer improves the visual appearance of surfaces and reduces the number of irregularities on the surface of reconstructed objects. In most cases, we also find that it helps to improve the overall realism of the reconstructed shape.

Regarding camera sampling, lighting, and shading, we keep nearly all parameters the same as [27]. We describe our setup in detail in the supplementary material.

4.2. Quantitative results

There are only few methods that attempt to reconstruct arbitrary objects in 3D. The most recent and best-performing of these is Shelf-Supervised Mesh Prediction [50], which we compare to here. They provide 50 pre-trained category-level models for 50 different categories in OpenImages [17]. Since we aim to compute metrics using 3D or multi-view ground truth, we evaluate on seven categories from the CO3D dataset [32] with corresponding OpenImages categories. For each of these seven categories, we select three images at random and run both RealFusion and Shelf-Supervised to obtain reconstructions.

We first test the quality of the recovered 3D shape in Fig. 5. [50] directly predicts a mesh, while we extract a mesh from our predicted radiance field using marching cubes. CO3D comes with sparse point-cloud reconstructions of objects obtained using multi-view geometry. For evaluation, we sample points from the reconstructed meshes and align them optimally with the ground truth point cloud by first estimating a scaling factor and then using Iterated Closest Point (ICP). Finally, we compute the F-score [40] with a threshold of 0.05 to measure the distance between the predicted and ground truth point clouds. Results are

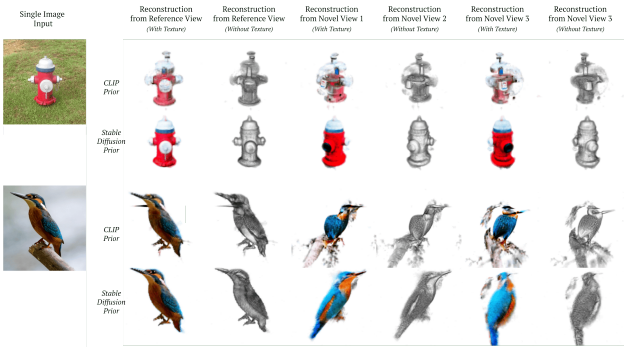

Figure 10. **Comparing Stable Diffusion and CLIP priors.** Results from two different priors: Stable Diffusion [34] and CLIP [29]. Stable Diffusion yields much higher-quality reconstructions, capturing more plausible object shapes.

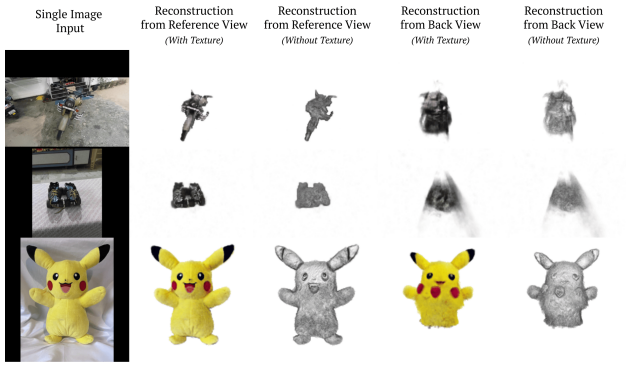

Figure 11. **Failure cases.** In the first two examples, the model simply fails to properly reconstruct the object geometry, and produces a semi-transparent scene which lacks a well-defined geometry. The third case is different in that the geometry is highly realistic, but the texture paints two Pikachu faces, one on each side of the object; this problem is sometimes called the Janus problem, after the two-faced Roman god.

shown in Tab. 1.

In order to evaluate the quality of the reproduced appearance, we also compare novel-view renderings from our and their method. Ideally, these renderings should produce views that are visually close to the real views. In order to test this hypothesis, we check whether the generated views are close or not to the other views given in CO3D. We report (Tab. 1) the CLIP embedding similarity of the generated images with respect to the closest CO3D view available (*i.e.*, the view with maximum similarity).

4.3. Qualitative results

Fig. 4 shows additional qualitative results from multiple viewpoints. Fig. 6 explores the ability of RealFusion to sample the space of possible solutions by repeating the reconstruction several times, starting from the same input image. There is little variance in the reconstructions of the front side of the object, but significant variance for its backside.

Fig. 11 shows two typical failure modes of RealFusion: in some cases, the model fails to converge, and in others, it copies the front view to the back of the object, even if this is not semantically correct.

4.4. Analysis and Ablations

One of the key components of RealFusion is our use of single-image textual inversion, which allows the model to correctly imagine novel views of a specific object. Fig. 7 shows that this component plays indeed a critical role in the quality and consistency of the reconstructions. Without textual inversion, the model often reconstructs the backside of the object in the form of a generic instance from the object category. For example, the backside of the cat statue in the top row of Fig. 7 is essentially a different statue of a more generic-looking cat, whereas the model trained with textual inversion resembles the true object from all angles.

Other components of the model are also significant. Fig. 8 shows that coarse-to-fine optimization reduces the presence of low-level artifacts and results in smoother, visually pleasing surfaces. Fig. 9 shows that the normal smoothness regularizer of Eq. (4) further results in smoother, more realistic meshes. Fig. 10 shows that using Stable Diffusion works significantly better than relying on an alternative such as CLIP.

5. Conclusions

We have introduced RealFusion, a new approach to obtain full 360° photographic reconstructions of any object given a single image of it. To achieve this, we leverage an off-the-shelf diffusion model trained using only pairs of 2D images and text (but no 3D supervision) and optimize a token embedding to represent the object depicted in the image. We use this conditional prior to learn an efficient, multi-scale radiance field representation of the reconstructed object, incorporating an additional regularizer to smooth out the reconstructed surface. The resulting method can generate plausible 3D reconstructions of objects captured in the wild which are faithful to the input image. Future improvements could be obtained by specializing the diffusion model for the task of new-view synthesis.

Ethics. We use the CO3D dataset in a manner compatible with their terms; it does not contain personal information. For further details on ethics, data protection, and copyright please see https://www.robots.ox.ac.uk/~vedaldi/research/union/ethics.html.

Acknowledgments. L. M. K. is supported by the Rhodes Trust. A. V., I. L. and C. R. are supported by ERC-UNION-CoG-101001212. I. L. and C. R. are also supported by VisualAI EP/T028572/1.

References

[1] David Bau, Jun-Yan Zhu, Jonas Wulff, William S. Peebles, Bolei Zhou, Hendrik Strobelt, and Antonio Torralba. Seeing what a GAN cannot generate. In *Proc. ICCV*, 2019. 2

[2] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *Proc. CVPR*, 2022. 2, 3

[3] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet an information-rich 3d model repository. *arXiv.cs*, abs/1512.03012, 2015. 3

[4] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proc. ECCV*, 2016. 3

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pages 8780–8794, 2021. 3, 4

[6] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3D shape induction from 2D views of multiple objects. In *arXiv*, 2016. 2, 3

[7] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 5

[8] Philipp Henzler, Niloy J. Mitra, and Tobias Ritschel. Escaping plato's cave using adversarial training: 3D shape from unstructured 2D image collections. In *Proc. ICCV*, 2019. 2, 3

[9] Philipp Henzler, Jeremy Reizenstein, Patrick Labatut, Roman Shapovalov, Tobias Ritschel, Andrea Vedaldi, and David Novotny. Unsupervised learning of 3d object categories from videos in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proc. NeurIPS*, 2020. 3

[11] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4

[12] Ajay Jain, Ben Mildenhall, Jonathan T. Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proc. CVPR*, 2022. 3

[13] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proc. ICCV*, 2021. 3

[14] Wonbong Jang and Lourdes Agapito. CodeNeRF: Disentangled neural radiance fields for object categories. In *Proc. ICCV*, 2021. 3

[15] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Pop Tiberiu. CLIP-Mesh: Generating textured meshes from text using pretrained image-text models. *SIGGRAPH Asia*, 2022. 3

[16] Jonáš Kulhánek, Erik Derner, Torsten Sattler, and Robert Babuška. ViewFormer: NeRF-free neural rendering from few images using transformers. In *Proc. ECCV*, 2022. 3

[17] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 7

[18] Chen-Hsuan Lin, Chaoyang Wang, and Simon Lucey. SDF-SRN: learning signed distance 3d object reconstruction from static images. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Proc. NeurIPS*, 2020. 2

[19] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *International Conference on Learning Representations*, 2021. 3

[20] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proc. ECCV*, 2020. 2

[21] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 4

[22] Norman Müller, Andrea Simonelli, Lorenzo Porzi, Samuel Rota Bulò, Matthias Nießner, and Peter Kontschieder. Autorf: Learning 3d object radiance fields from single view observations. *CoRR*, abs/2204.03593, arXiv.cs. 3

[23] Thomas Muller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 2, 5

[24] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. HoloGAN: Unsupervised learning of 3D representations from natural images. *arXiv.cs*, abs/1904.01326, 2019. 2, 3

[25] Thu Nguyen-Phuoc, Christian Richardt, Long Mai, Yong-Liang Yang, and Niloy J. Mitra. Blockgan: Learning 3d object-aware scene representations from unlabelled images. In *Proc. NeurIPS*, 2020. 2, 3

[26] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[27] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv.cs*, abs/2209.14988, 2022. 2, 3, 7

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. ICML*, volume 139, pages 8748–8763, 2021. 3

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen

Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 8

[30] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021. 2

[31] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *International Conference on Computer Vision*, 2021. 3, 6

[32] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common Objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction. In *Proc. CVPR*, 2021. 7

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2

[34] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proc. CVPR*, 2022. 3, 6, 8

[35] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 2

[36] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *Proc. NeurIPS*, 2022. 2, 6

[37] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Proc. NeurIPS*, 2019. 3

[38] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Proc. NeurIPS*, 33:12438–12448, 2020. 3

[39] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Proc. ICLR*, 2021. 3

[40] Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn? In *CVPR*, 2019. 7

[41] Itsuki Ueda, Yoshihiro Fukuhara, Hirokatsu Kataoka, Hiroaki Aizawa, Hidehiko Shishido, and Itaru Kitahara. Neural Density-Distance fields. In *Proc. ECCV*, 2022. 2

[42] Kalyan Alwala Vasudev, Abhinav Gupta, and Shubham Tulsiani. Pre-train, self-train, distill: A simple recipe for supersizing 3d reconstruction. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[43] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv.cs*, abs/2106.10689, 2021. 2

[44] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P. Srinivasan, Howard Zhou, Jonathan T. Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas A. Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proc. CVPR*, 2021. 2

[45] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2021. 3

[46] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv.cs*, abs/2210.04628, 2022. 3

[47] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *Proc. NeurIPS*, 2016. 2, 3

[48] Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *Int. J. Comput. Vis.*, 128(12), 2020. 3

[49] Farid Yagubbayli, Alessio Tonioni, and Federico Tombari. Legoformer: Transformers for block-by-block multi-view 3d reconstruction. *arXiv.cs*, abs/2106.12102, 2021. 3

[50] Yufei Ye, Shubham Tulsiani, and Abhinav Gupta. Shelf-supervised mesh prediction in the wild. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3, 6, 7

[51] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proc. CVPR*, pages 4578–4587, 2021. 3