1 Paper Summary

The paper, titled "Common Objects in 3D: Large-Scale Learning and Evaluation of Real-life 3D Category Reconstruction" was published to the 2021 Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), by Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, David Novotny, from the Meta AI research group.

The paper is composed of two parts. The first part being the contribution of a large dataset, Common Objects in 3D (CO3D). CO3D is a dataset comprised of videos of static objects, along with their corresponding 3D ground truths, represented by point clouds approximating the object surface. The second contribution in the paper is a new deep learning architecture, NerFormer. The architecture was developed to improve the state of the art for the task of object category reconstruction. This reconstruction task involves taking in a limited number of 2D views of an instance of an object class, then being able to reconstruct a 3D representation of it.

1.1 CO3D

The dataset created by the authors follows the category separations introduced by Lin et al. in their work Microsoft COCO (MS COCO)[4]. The authors specify that other datasets with 3D ground truth annotations are either limited in the categories they represent, such as that recorded by lim et al.[3], or comprised purely of synthetic examples [1], representing a significant domain shift when applying trained models on real world examples.

To facilitate their large scale data collection, the authors leveraged Amazon Mechanical Turk (AMT), a crowdsourcing platform. Each worker on the platform would choose a category from the MS COCO set. They would then place the object on a solid surface, and using a smartphone camera, film a video of the object while circling around it. For generating the 3D annotations and extracting relative camera poses of the individual frames within a particular video, the authors leveraged the mature structure-from-motion (SfM) software, COLMAP[9]. On passing the videos to COLMAP, a camera pose annotation is produced, along with an approximate per frame depth map. The depth map is then passed to COLMAP's point cloud algorithm, to create a combined 3D pointcloud for the object. Post this automated processing, the author's then manually check the generated 3D ground truths in order to remove erroneous results. Post checks, the dataset comprises of 1.5 million video frames in 18,600 videos from 50 of the MS COCO categories.

1.2 NerFormer

The architecture proposed by the authors can be seen as an improvement over a previous work by Henzler et al. which introduces the idea of Warp Ray Embeddings (WREs)[2]. This work in turn makes heavy use of the seminal work done by Mildenhall et al.[5], which proposed the idea of Neural Radiance Fields (NeRF). A NeRF is a Multi-Layer Perceptron (MLP) that learns to represent a 3D space, through the ability to render new views of that particular space. The NeRF is trained on a dataset of photos taken of the space. The network learns to approximate 2 functions, $\sigma(X)$ and c(X, r), with $\mathbf{X}$ being a 3D coordinate in the

space, and **r** being the direction from which we wish to view the scene. $\sigma(X)$ represents a probability distribution that at point X, it is filled with material opaque enough to stop or reflect a ray of light. Function c learns what colour light, if it hits point **X** from direction **r**, would emit. To render a view of a scene, we project lines, or 'rays', from the location of the view's camera into the scene. We sample 3D points along the ray within the scene and pass them into the learned σ and c. Each ray corresponds with a pixel in the output image. This can be seen in figure 2, wherein pixel u for target view l^{tgt} corresponds with several points along projected ray r_u. By marching over all points along all rays, we can generate an image of the scene, in a process known as ray-marching.

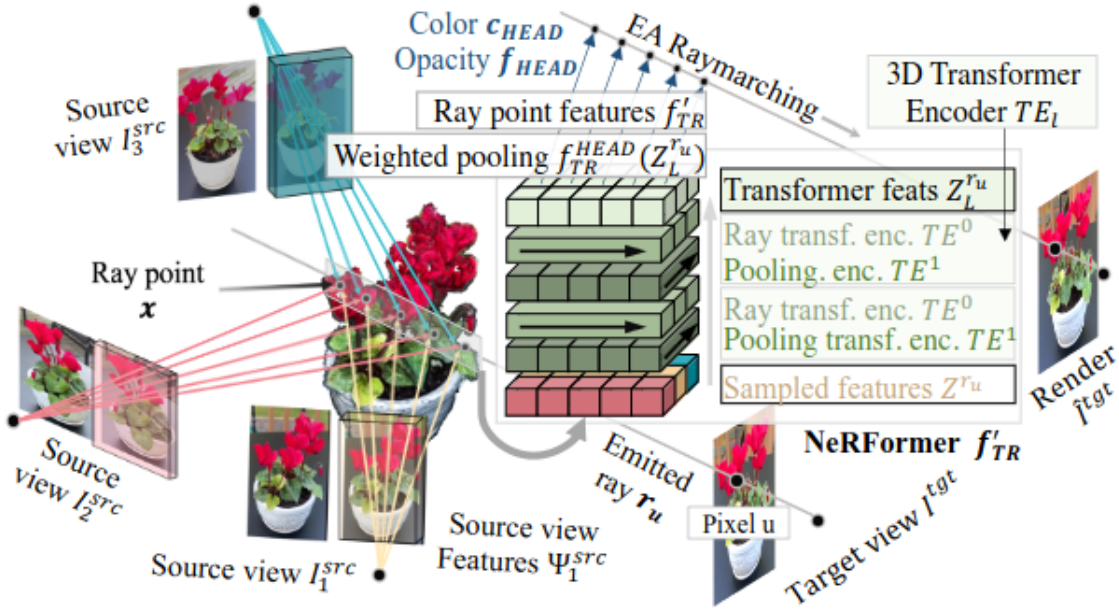

Figure 1: NerFormer Architecture as illustrated by [8]

To extend this idea beyond representing a single space to being able to represent a particular object class, during inference, the model needs some notion of the specific object it is to reconstruct. To this end, WREs are embeddings, learned through the introduction of a Convolutional Network ϕ at the start of a NeRF pipeline. ϕ learns to produce a feature map for each of the source images of the specific object to be reconstructed. As rays are projected into the scene from the origin of the target rendering, from which points **X** are then sampled. A reverse projection can take place identifying the pixels in the source image(s), to which **X** corresponds. As seen in figure 2, point **x** on r_u corresponds with specific pixels in source views l_1^{src}, l_2^{src}, l_3^{src}. The relevant pixels are then passed into corresponding feature mappings to generate embeddings. The embeddings are then combined through a weighted sum, with weight being determined by the cosine similarity between a source image's angle relative to the target rendering. This combined embedding is then passed into the standard NeRF (in addition to the standard inputs **X** and **r**).

The authors of the subject paper argue that this simple weighting, in addition to processing scene coordinates along a ray independently can lead to sub-optimal results, particularly

if there are noisy source input images, for example caused by bright lighting or shadows. NerFormer seeks to address the noise issue by instead modelling the points along $\mathbf{r}$ as a sequence, in addition to learning a aggregation function for the source image WREs. The authors argue, that by giving the model access to this extra context, it is able to detect failures and recover from them via *spatial reasoning*. The input to this new network is a tensor of size N_S x N_{Src} x D_Z with N_S being the ray dimension, N_{Src} being the source image dimension, and D_Z being the latent embedding (WRE) dimension. The authors in-

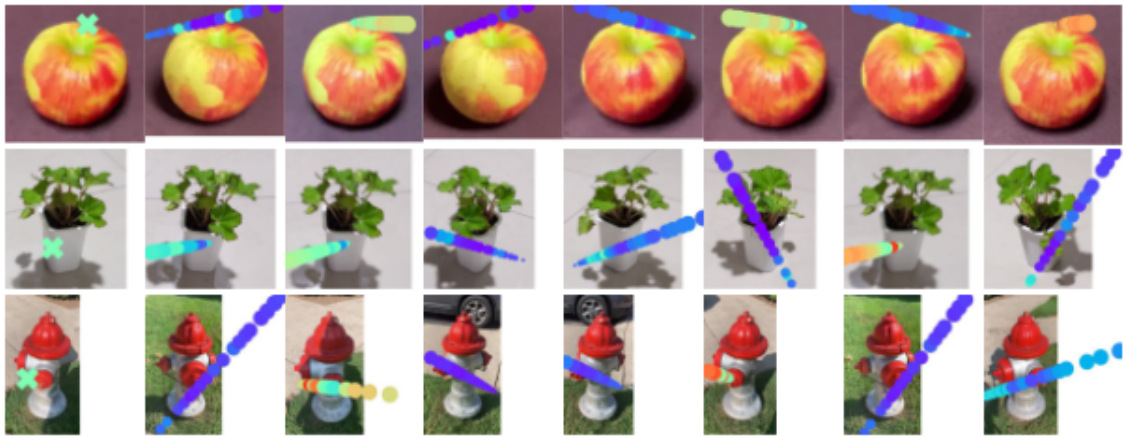

Figure 2: Attention along ray dimension, as illustrated by [8]

tegrate transformer modules[11] to first perform multi-headed self-attention across the N_S dimension, then the N_{Src} dimension. Finally there is a pooling layer which learns to pool across the N_{Src} dimension. Figure 2 illustrates levels of attention from a target pixel in a target view in relation to different source view features. Red is high attention, blue is low attention. This pooled input is then passed to an MLP and the NeRF pipeline continues as before. The authors provide some qualitative and quantitative comparisons between this new architecture and existing work.

1.3 Benchmarks and Evaluation

The authors state that training the architecture on all 50 collected categories of the dataset was too computationally expensive, with training on 1 category taking more than 7 days, so a subset of 10 was chosen for evaluation. The authors perform qualitative comparisons of NerFormer results against pre-NeRF methods, such as the Scene Representation Network[10], or Differentiable Volumetric Rendering (DVR)[6]. The results can be seen in fig 3.

For quantitative comparisons, the authors show values for a number of metrics, averaged across results on an unseen test subset of the training data. The results are summarized in figure 4.

PSNR: Computes the Peak Signal to Noise Ratio (in decibels) between two images, the original, and the compressed (or in this case rendered). The higher the value, the better quality the reconstruction. For this metric, the NerFormer performs best, with a value of

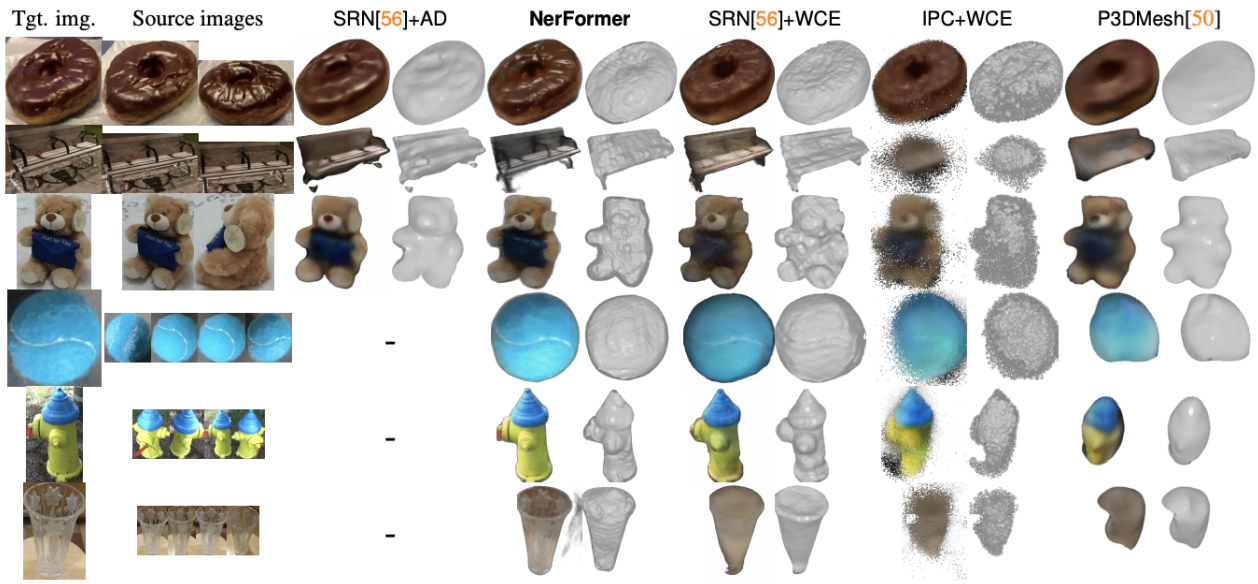

Figure 3: Qualitative comparisons taken from [8]

method	PSNR	LPIPS	ℓ_1^{depth}	IoU	method	PSNR	LPIPS	ℓ_1^{depth}	IoU
NerFormer	23.3	0.17	0.40	**0.96**	SRN[56]	20.4	0.21	0.60	0.93
NeRF+WCE[26]	21.0	0.19	0.74	0.91	SRN+WCE+γ	16.9	0.30	0.60	0.75
NeRF[43]	**23.6**	0.17	**0.38**	0.95	SRN+WCE	15.8	0.26	0.64	0.80
NV[41]	22.2	0.20	0.91	0.91	SRN+γ	16.9	0.30	0.59	0.75
NV+WCE	18.7	0.25	0.85	0.90	DVR[45]	15.0	0.33	0.89	0.68
IDR[74]	18.5	**0.15**	0.81	0.92	DVR+γ	15.8	0.38	0.74	0.65
IPC+WCE	13.9	0.25	1.59	0.83	P3DMesh[50]	17.3	0.22	0.69	0.91
IPC	13.8	0.25	1.58	0.83					

Figure 4: Quantitative comparisons taken from [8]. Formatting indicates **best result**, Second best result

17.9 in the test set against the other methods (SRN, with WREs, is the nearest performer with a value of 14.6).

LPIPS: Learned Perceptual Image Patch Similarity, is a learned perceptual similarity metric, trained by Zhang et al.[12] that is supposedly reflective of how humans perceive image similarity. A lower value means more similarity. For this metric, the NerFormer came joint second on the test set, with a value of 0.27. The leading method, P3D Mesh[7] scored 0.26.

L1 Depth: The L1 Depth, is the Manhattan distance between points of 2 point clouds, the ground truth and the rendered cloud. For this metric NerFormer achieved a score of 0.91 in the test set, 3rd place behind two forms of the SRN with various input encodings (scoring 0.40 and 0.36).

IoU: The Intersection over Union score is a metric used to evaluate the accuracy of image segmentation, which for this task means comparing the space an object occupies in the rendered image versus that of the ground truth. For this metric, NerFormer gave a value of 0.81, second to SRN which gave a value of 0.82.

The authors perform further studies using the PSNR metric, wherein they compare the effectiveness of the model as the number of source images are varied. In these comparisons, the NerFormer is the most effective.

1.4 Discussion

During the seminar, a number of issues surrounding the paper were raised. The original justification as to why NerFormer was needed was that a NerF + WREs may produce noisy renderings. However, the paper does not provide any examples of such failure cases, and surprisingly such an architecture is not included in the qualitative comparison. Similarly, the author's stipulation that NerFormer is able to recover from rendering failures through *spatial reasoning* is unsupported, as nowhere do they provide any evidence of it being able to handle noisy inputs particularly well. In addition, although some quantitative comparisons are done, the results are not analysed, nor are they presented in a readily comparable way. The new architecture performs well against comparable techniques, however, state of the art results are only achieved for one metric. The authors give no indication as to the varying importance of different metrics, or how these translate tangibly to qualitative improvements, thus there is no indication if the new architecture provides a 'real-world' improvement over existing methods. In a similar vein, although initial quantitative comparisons are done with a number of different metrics, for subsequent studies the effectiveness of the architecture with varying numbers of source images, the authors use only PSNR, the one metric for which their architecture is the most performant.

Although the architecture is relatively complex, the authors' architecture diagram could be broken down into separate figures to better illustrate the proposed training flow. Similarly, as the paper was presented at a computer vision conference, the preceding work, which stems from advances in field of computer graphics, could be explained in more depth, to account for the audience's unfamiliarity with graphics work.

In addition, the issue around suitability for deployment of NerFormer in medical, or safety critical settings was raised. Although qualitatively the results appear reasonable, NerFormer's output (along with competing architectures) is still relatively imprecise in relation to the 3D ground-truth, as shown by the L1 Depth scores, thus perhaps making the approach unsuitable for applications where exact surface boundaries need to be known.

As NerFs are a relatively novel architecture, there has been very little practical use of them, or derived architectures, in the medical field. There are possibilities however, which may include leveraging 2D slice data to build 3D representations of patient anatomy. Such 3D renderings may have use to help with surgical planning, or for providing more realistic training tools for medical students.

2 Acknowledgements

The student was supported by UK Research and Innovation [UKRI Centre for Doctoral Training in AI for Healthcare grant number EP/S023283/1].

References

[1] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[2] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4709, 2021.

[3] Joseph J Lim, Hamed Pirsiavash, and Antonio Torralba. Parsing ikea objects: Fine pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2992–2999, 2013.

[4] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[5] 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. *Communications of the ACM*, 65(1):99–106, 2021.

[6] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020.

[7] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv preprint arXiv:2007.08501*, 2020.

[8] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10901–10911, 2021.

[9] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[10] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[12] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.