Automated Content Editing in NeRFs

Joel Julin[12], Heng Yu[2], László Jeni[2]

Abstract— **Neural rendering for novel view synthesis has been a rising problem within the computer vision community. Among the many proposed techniques, neural radiance fields (NeRF) have proven to be one of the most effective. NeRF represents static scenes as a fully-connected deep network with a 5D coordinate input (x, y, z, θ, ϕ) and an output of color and density (RGBσ). The system introduced by NeRF builds a 3D representation of a scene given a number of 2D images. When applied to dynamic scenes, NeRF's performance significantly declines. Recent strides have been made towards solving this problem with the likes of CoNeRF and Non-Rigid NeRF; both works have shown to be effective in re-rendering and manipulating neural radiance fields despite the presence of dynamic objects. However, this previous research is hindered in both the labor and function domain. CoNeRF requires the tedious task of manually annotating the dynamic component of the input images; whereas Non-Rigid NeRF is unable to generalize to new movements and only works with a single deformable object. We propose a followup method capable of re-rendering and manipulating a dynamic object within a radiance field without the need for manual annotation. With our proposed method, dynamic scenes with the human shape can be more easily rendered and manipulated. In addition to dynamic scenes, our work also brings benefits to static scene manipulation. We hope that this work sheds light on future NeRF manipulation methods.**

Index Terms— **keywords, Computer Vision, Neural Rendering, Computer Graphics**

I. INTRODUCTION

Neural Radiance Field (NeRF) has recently become the standard method for view synthesis. This is certainly not without reason, as NeRF has outstanding performance on both static [1]–[5] and dynamic objects [6], [7]. Despite this performance in both domains, radiance field manipulation and usability remains an open research question. There is a large amount of research dedicated to the application of NeRF to dynamic scenes, but it is often tedious and requires much work.

One such work that aims to control the dynamic movement within neural radiance fields is CoNeRF. CoNeRF yields impressive results, but its scalability is largely limited due to the need for manual annotation of the controllable component. There also exist few methods that utilize automatic segmentation of the dynamic and static components but fail to provide fine-tuned rendering selection, such as Non-Rigid NeRF [6]. In this paper, we propose a method capable of automatically, and accurately, annotating both a dynamic and static human body for use in neural radiance field control and manipulation. For use in dynamic scenes, our method proves to be effective at radiance field control with CoNeRF. Within the static domain, the same method is effective at segmentation and rendering control. This work predominantly showcases efficient usage of image segmentation for dynamic scene control while also revealing an additional use-case for static scenes. In specific, this paper describes a method that offers:

- **Automatic Selection of the Human Shape for Neural Rendering.** Automatically selecting the shape of interest within a scene brings heaps of improvements to both static and dynamic neural rendering, namely the elimination of manual annotation. While currently structured to solely segment human bodies, this work can be extended to a variety of other classes.
- **Dynamic Scene Manipulation.** Many of the NeRF methods that apply to dynamic scenes [8] require a degree of manual annotation or are limited in function [6]. This work applies automatic segmentation methods to this dynamic scene control.
- **Static Scene Manipulation.** Given the selected object(s) the neural radiance field can be rendered without the selected objects, or without the background.

II. RELATED WORKS

Our work is closely related to a number of recent developments made within neural rendering.

A. Neural Rendering for Novel View Synthesis

NeRF has led to cascades of research on neural rendering for novel view synthesis. The original method proposed by Mildenhall et al. [1] is capable of building a high fidelity 3D representation of a scene given a number of 2D images. When initially published, a large constraint of this method was its inability to represent non-rigid or dynamic scenes. Since then, there have been a few works that extend NeRF's exemplary performance on static objects to objects in motion [6]–[8].

B. CoNeRF: Controllable Neural Radiance Fields

CoNeRF is one of the most influential NeRF followup works that propose a method for neural radiance field control [8]. The method proposed in this work controls object movement through tedious annotation of the controllable component (i.e. arm moving) and a value assignment. To annotate the controllable component, CoNeRF uses a manual click and drag annotation software known as labelme [9]. At each annotated frame, a value is assigned to represent

[1]Joel Julin is with the School of Computing and Information, University of Pittsburgh, Pittsburgh, PA 15260, USA jmj96@pitt.edu

[2]Joel Julin, Heng Yu, and László Jeni are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15260, USA jjulin, hengyu, laszlojeni@andrew.cmu.edu

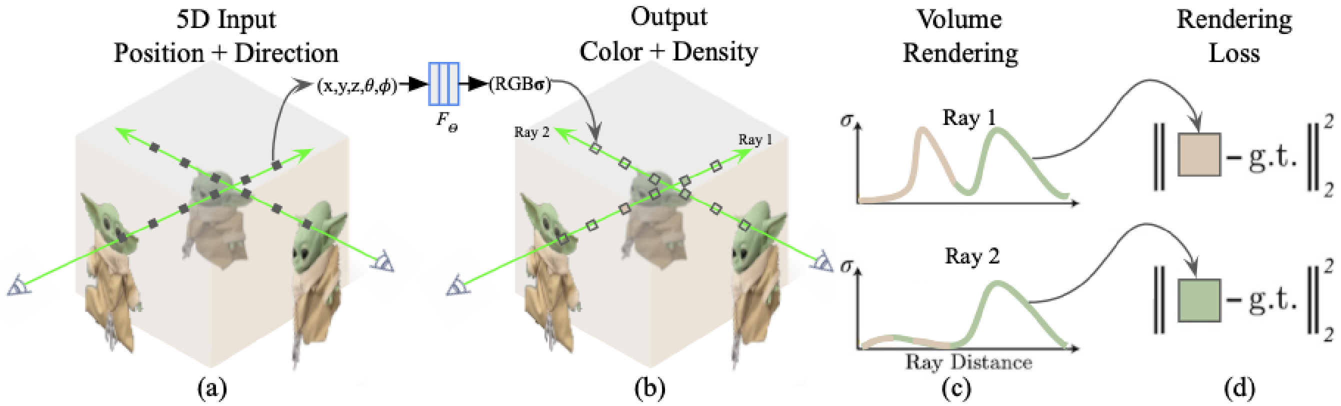


Fig. 1. NeRF architecture

the transition status. For instance, an arm fully located at a person's side would be assigned a value of -1.0, an arm that is full stretched out would be given a value of 1.0, and an arm that is somewhere in the middle would be assigned a value of 0.0. Since our work brings automatic annotation to CoNeRF and improves its usage, it is closely related. However, our work does not overlap with that of CoNeRF's [8] in terms of contribution. We only use this preexisting work as a means to display our application of automatic annotation.

C. Non-Rigid NeRF

Non-Rigid NeRF [6] focuses on the automatic separation and manipulation of a scenes rigid (static) and non-rigid (dynamic) counterparts. Non-Rigid NeRF is largely limited in function, this work is *only* capable of scene manipulation when composed of both static and dynamic objects. Our work significantly extends upon theirs since we enable radiance field manipulation regardless of the scenes composition; meaning that distinct dynamic and static components are not required.

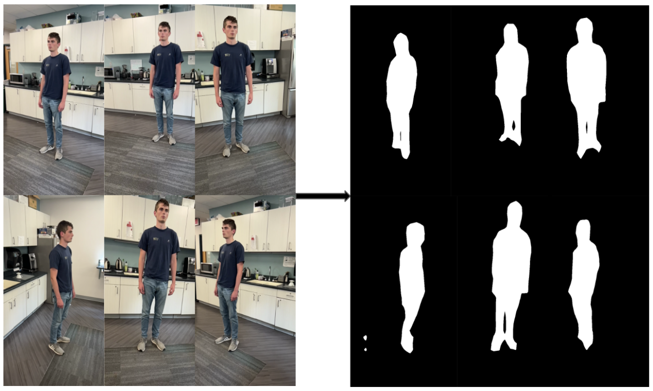
III. Method

Our method consists of four components (i) data collection and preparation, (ii) automatic segmentation, (iii) NeRF architecture for automated content editing, and (iv) automated segmentation for dynamic scenes.

A. Data Collection and Preparation

The data used for this work was captured using an iPhone 13 Pro's 240fps slo-mo camera. For a thorough 3D representation of the scene and a large number of viewing angles, the video was captured in a circular motion with a moving camera. After the video is captured, a sparse set of the captured frames (approximately 300) are passed through COLMAP to obtain the camera poses that are needed to determine where the camera is located during each frame. This is important because without the poses, there is no structural information of where these images were captured in relation to other images, and are needed as inputs to NeRF's fully connected network that will later be discussed in further detail.

B. Automatic Segmentation



a) input

b) masked output

Fig. 2. Body Pix 2.0 generates near perfect masks. These binary masks are used to determine the object of interest. Areas marked as white are editable and areas marked as black are unaffected. It is important to note that these masks can be inverted. When this happens, the areas marked as white become black and the areas that were once marked as black become white, effectively switching the areas of interest.

BodyPix 2.0 is an effective segmentation software that is directly trained to recognize, and segment, the human shape. As shown in Fig. 2, the masks that BodyPix 2.0 creates are quite accurate. The masks generated by this software are used to determine which object in our scene we wish to manipulate.

C. NeRF Architecture for Automated Content Editing

At the heart of this work is the standard NeRF method. [1]. This method works by representing a static scene as a fully-connected deep network with a 5D coordinate (x, y, z, θ, ϕ) and an output of color and density (RGBσ). As shown in Fig. 1, given an image from a given viewing direction or camera position (θ, ϕ) a ray is passed through each pixel. As that ray is sent through the pixel at location (x,y), a sampled point z is sent through a fully connected deep network and is outputted a color (RGB) and density (σ) where density is a value that denotes whether or not an object is present.

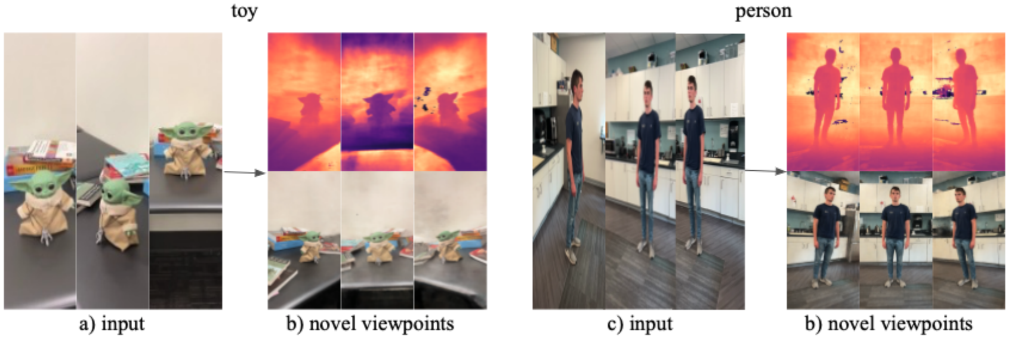


Fig. 3. The standard NeRF method yields impressive results when applied to our own image sequences. The inputs to both of the above experiments contain approximately 150 images from a variety of different viewing directions.

This process of sampling a point along the ray and passing it through the deep network to receive an output of color and density is repeated for every sample along the ray as shown in part b of the figure. Whenever this process is completed, all of the sampled points are combined using a classical volume rendering technique [10] to receive the final prediction of the pixel's color as shown in part c. The final step (d) within the NeRF architecture is to compute the loss between the rendered color and the ground truth, then take that loss to reduce the rendering error in future iterations.

With the standard NeRF method now being outlined, we will now present our simple, yet successful, modification that allows for automated content editing. Since NeRF renders the color of a pixel from a sampled ray, removing the entire ray effectively prevents portions of a scene from being rendered. When this ray removal is applied at a larger scale, by utilizing selective binary masking, entire objects can be removed from the scene. Partnering binary object masking with ray removal is the extent of our method that allows for static scene manipulation.

D. Automated Segmentation for Dynamic Scenes

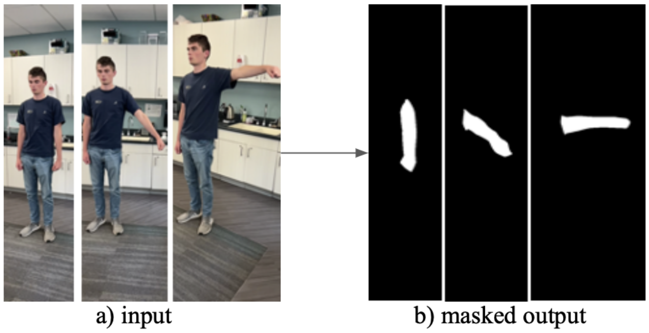


Fig. 4. Body Pix 2.0 is capable of automatically generating masks for parts of the body, such as an arm.

Our method of utilizing automatic segmentation for static NeRFs, also brings benefits to NeRFs representing dynamic scenes. Instead of removing components of a scene, we applied this method to more easily control a scene. CoNeRF: Controllable Neural Radiance Fields [8] acted as our standard dynamic method to which we made modifications to. This method uses manual annotation to signify which component within a scene is in motion, which is oftentimes a tedious task. The automatic segmentation software, Body Pix 2.0, entirely removes the need for manual annotation of the controllable segment of the scene as shown in Fig. 4. The primary modification made to CoNeRF was to directly accept binary images as labels instead of a json file containing the mask coordinates generated by labelme (a manual annotation software). Other than this modification, the original CoNeRF code was used for our experiments.

IV. EXPERIMENTS

The experiments that were conducted for this work include standard NeRF without modifications acting as our baseline, static NeRF modifications with both manual and automatic annotations, and dynamic NeRF modifications with automatic annotation. In the following sections we will explain each of these experiments in complete detail.

A. Baseline: Standard NeRF

This first experiments conducted for this project used the standard NeRF code without any modifications. The NeRF architecture, as shown in Fig. 1, sends rays through every pixel from a given viewing direction, samples along each of the rays to obtain (RGBσ), and applies a volumetric rendering technique [10] to accumulate the sampled points and render each pixel's predicted color. The loss between the ground-truth and this prediction is used to reduce the rendering error in future iterations. We applied this method to two of our own image sequences, a toy and a person (Fig. 3), with each sequence containing approximately 150 images.

B. Static NeRF Modifications

The two experiments for static NeRF modifications concerned object and background removal. For both of these experiments, the standard NeRF model was modified such that certain rays falling within a masked region are not rendered.

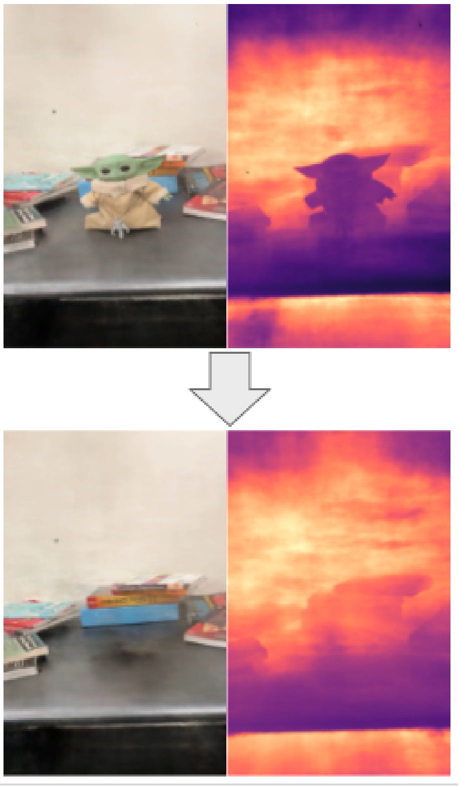


Fig. 5. Our method is capable of fully removing an object from a neural radiance field (bottom). This successful removal becomes even more apparent when compared to the standard NeRF output and depth map generated from the same images (top).

1) Object Removal: Our method for object removal was first tested on a manually annotated image sequence as shown in Fig. 5. As expected, manual annotating a large amount of images is tedious. Nevertheless, this experiment demonstrated the effectiveness of our method before introducing automatic annotations from Body Pix 2.0.

The success of this method continues to hold when using Body Pix 2.0's automatic annotation to generate binary masks. As shown in Fig. 6, the person is removed from the neural radiance field. By using a much more efficient annotation method, these experiments become much more feasible.

2) Background Removal: Similar to the object removal experiments, we first observed the result of our background removal method using a manually annotated scene before using an automatically annotated one. Fig. 7 showcases a viewpoint taken from this experiment. Since annotations for our project are binary (i.e. selected component is marked as white and unaffected as black), it took little modification to our object removal method to remove the background. Using the automatic annotations from Body Pix 2.0 we can more easily remove the background from NeRFs, as shown in Fig. 8.

C. Dynamic NeRF Modifications using CoNeRF

In order to apply our usage of automatic segmentation to CoNeRF, a NeRF variant that aims to control dynamic

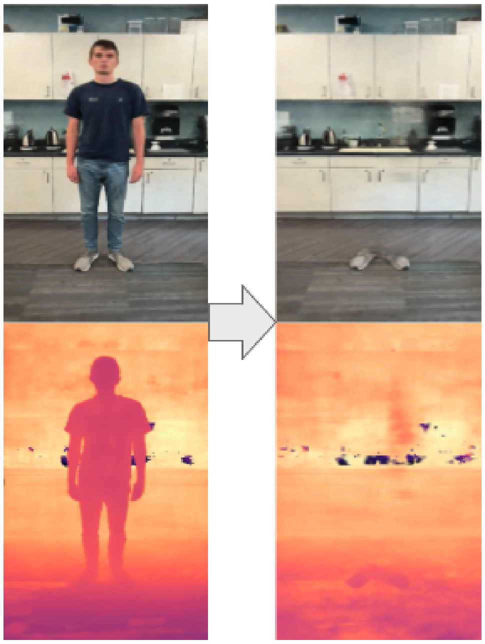


Fig. 6. When applied to scenes in which the object is automatically annotated by Body Pix 2.0, we see similar performance. The standard NeRF rendering (left) compared to the modified rendering (right) show that the person is almost entirely removed from the scene. The shoes of the person partially remain. This can be explained by imperfect annotations.

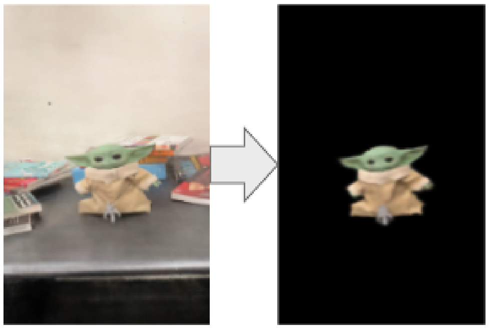


Fig. 7. By inverting the manually annotated binary mask, we can select and remove the background of the scene as opposed to the toy.

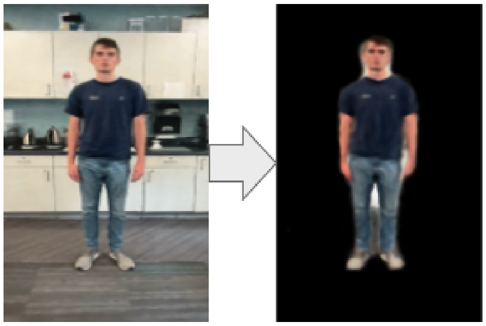


Fig. 8. Inverting the automatically annotated binary mask lets us select and remove the background of the scene as opposed to the person.

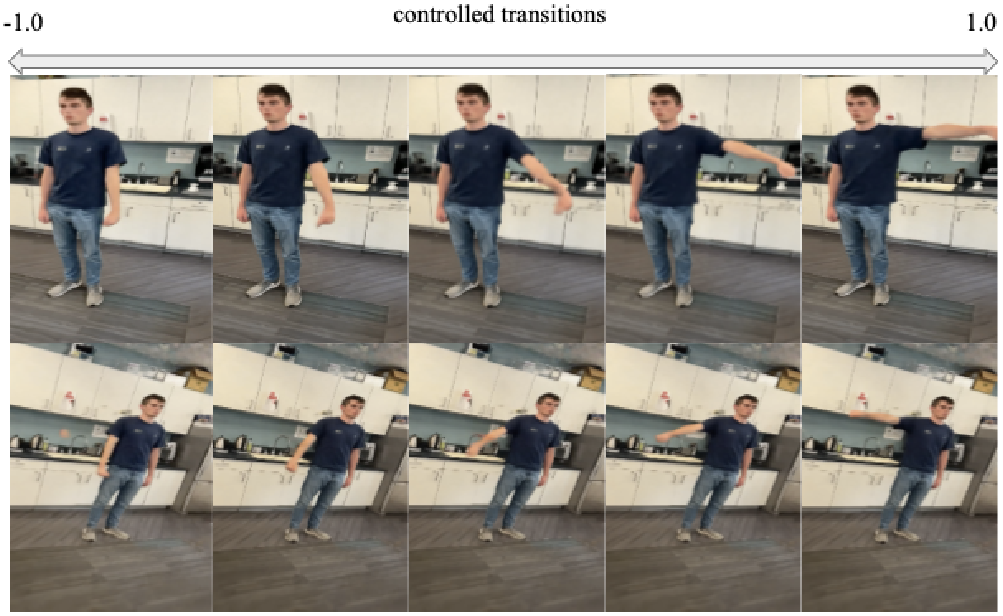


Fig. 9. Controlled output obtained from CoNeRF using automatic annotation for the arm from transition states -1.0-1.0 where -1.0 is the start state and 1.0 is the final state.

scenes, a minor modification was needed. The original CoNeRF code only accepts manual annotations generated from LabelMe in the form of a json file and *then* converts those coordinates to a binary mask. To make this code suitable for our automatically generated masks, we simply removed the json conversion and directly used our binary masks of the dynamic component. Now with CoNeRF accepting our binary masks, we now supplied transition values, from -1 - 1, to a sparse set of the captured frames. The final controllable output for this experiment of both right arm and left arm movement can be found in Fig. 9.

V. CONCLUSIONS

This work proposed a method for automated content editing in NeRFs that allows for simple manipulation of both static and dynamic neural radiance fields. Using the automatically generated masks from Body Pix 2.0 and our ray removal method, static scenes can be rendered without a person present and the background intact or without the background and the person unaffected. When applying our use of automatic annotation to preexisting dynamic NeRF methods, such as CoNeRF, we can remove the need for manual annotation of the controllable component. Both intended applications of this method proves to be successful.

While the experiments presented within this paper were a success, there still remains numerous directions for future work. A few of the most promising steps are to improve the masking coverage to more fully capture the object of interest, apply different segmentation methods to automatically annotate objects of different classes, extend this method to other dynamic NeRF variants that rely on manual annotation, and increase the editing possibilities.

ACKNOWLEDGMENT

This project was funded by the National Science Foundation (NSF) under Grant No. 1659774. I want to say thank you to the Robotics Institute Summer Scholar program organizers Dr. John M. Dolan and Rachel Burcin, my faculty mentor Dr. László A. Jeni, and Dr. Simon Lucey for the many helpful conversations. I will forever be grateful for this engaging academic and social experience.

REFERENCES

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *CoRR*, vol. abs/2003.08934, 2020. [Online]. Available: https://arxiv.org/abs/2003.08934

[2] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," 2022. [Online]. Available: https://arxiv.org/abs/2203.09517

[3] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5501–5510.

[4] Q. Xu, Z. Xu, J. Philip, S. Bi, Z. Shu, K. Sunkavalli, and U. Neumann, "Point-nerf: Point-based neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5438–5448.

[5] K. Zhang, G. Riegler, N. Snavely, and V. Koltun, "Nerf++: Analyzing and improving neural radiance fields," *arXiv preprint arXiv:2010.07492*, 2020.

[6] E. Tretschk, A. Tewari, V. Golyanik, M. Zollhöfer, C. Lassner, and C. Theobalt, "Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a deforming scene from monocular video," *CoRR*, vol. abs/2012.12247, 2020. [Online]. Available: https://arxiv.org/abs/2012.12247

[7] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 10 313–10 322.

[8] K. Kania, K. M. Yi, M. Kowalski, T. Trzcinski, and A. Tagliasacchi, "Conerf: Controllable neural radiance fields," *CoRR*, vol. abs/2112.01983, 2021. [Online]. Available: https://arxiv.org/abs/2112.01983

[9] K. Wada, "labelme: Image polygonal annotation with python," https://github.com/wkentaro/labelme, 2018.

[10] J. T. Kajiya and B. P. Von Herzen, "Ray tracing volume densities," *ACM SIGGRAPH computer graphics*, vol. 18, no. 3, pp. 165–174, 1984.

[11] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Deformable neural radiance fields," *CoRR*, vol. abs/2011.12948, 2020. [Online]. Available: https://arxiv.org/abs/2011.12948

[12] S. Liu, X. Zhang, Z. Zhang, R. Zhang, J.-Y. Zhu, and B. Russell, "Editing conditional radiance fields," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[13] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv preprint arXiv:2106.13228*, 2021.