SRMCV Project: Extending the Diffeomorphic Neural Reconstruction Approach

Vasiliki Papadouli, Alexander Fuchs, Zehranaz Canfes

Practical Course: Shape Reconstruction and Matching in Computer Vision

TUM School of Computation, Information and Technology

Technical University of Munich

14.08.2023

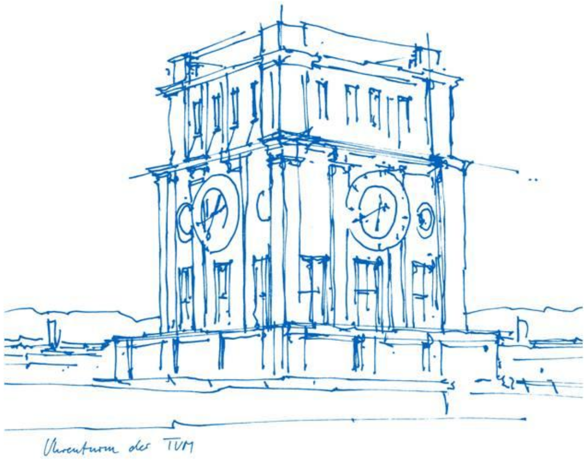
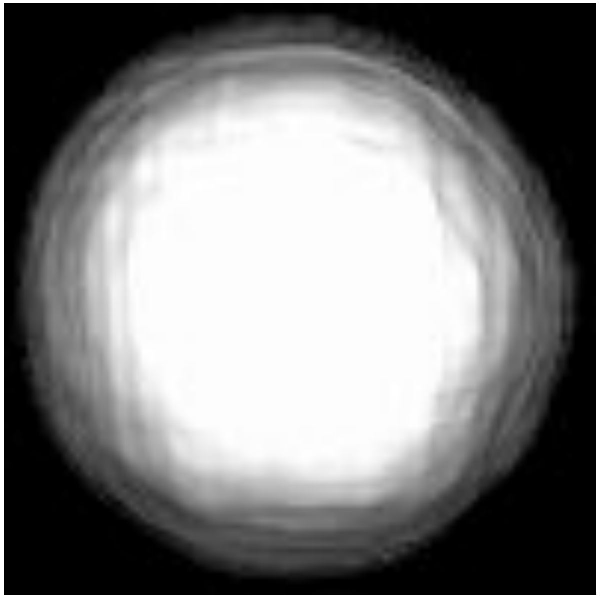
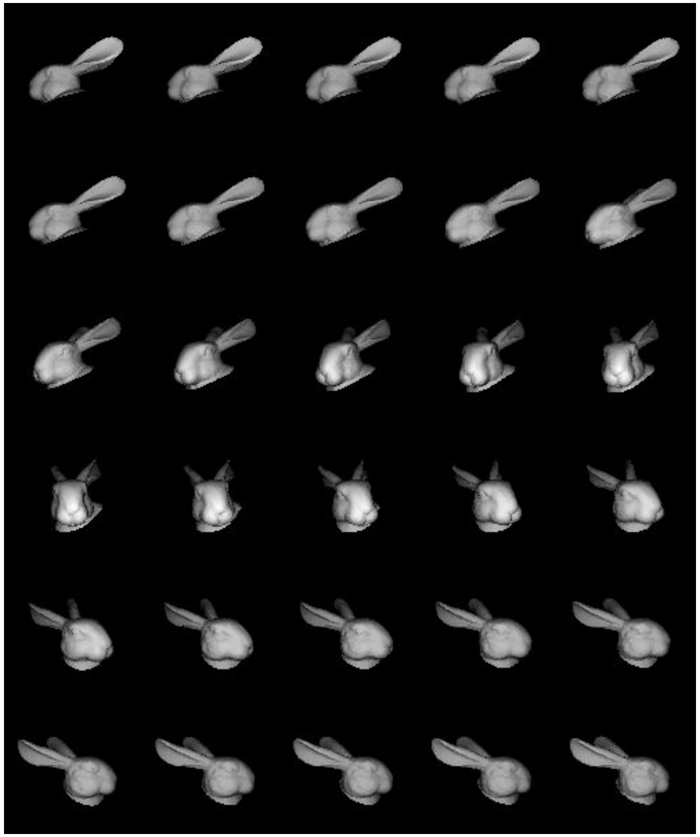


Table of Contents

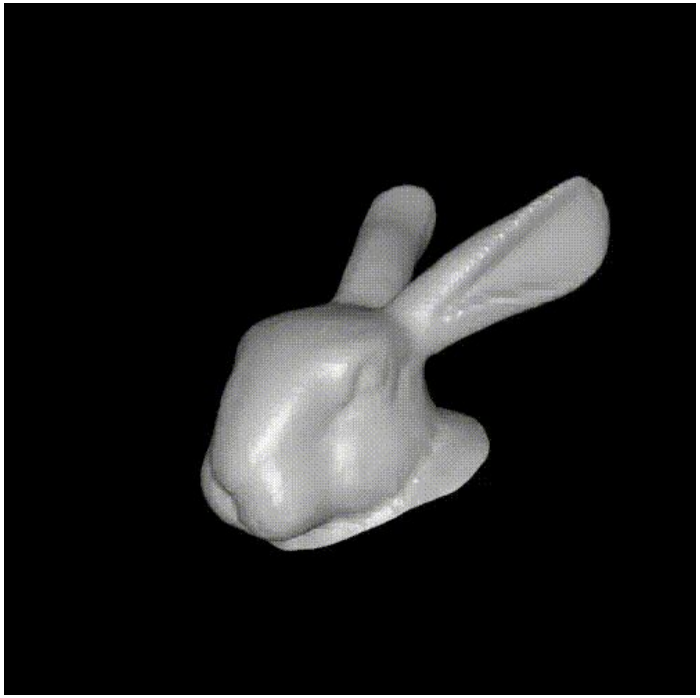


Introduction: Mesh Reconstruction

Input: Images



Output: Mesh



Traditional Approaches

Structure from Motion (e.g. colmap[1])

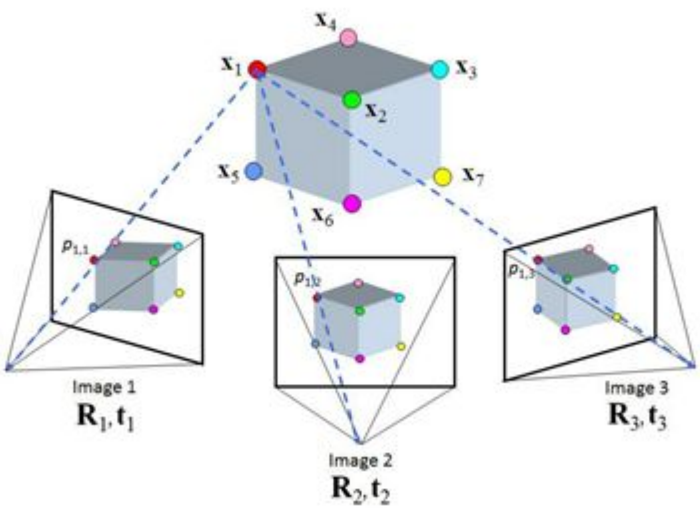


Image Source: *Yilmaz, Ozgur & Karakus, Fatih. (2013). "Stereo and kinect fusion for continuous 3D reconstruction and visual odometry"*

Voxel Carving[2]

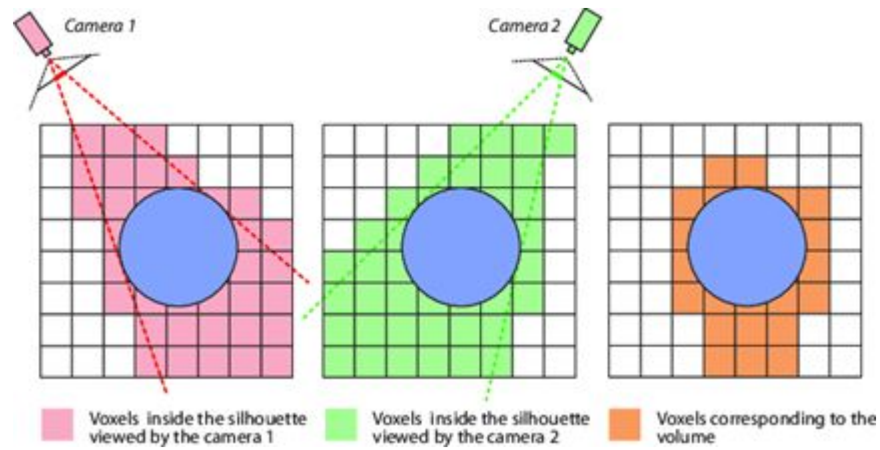
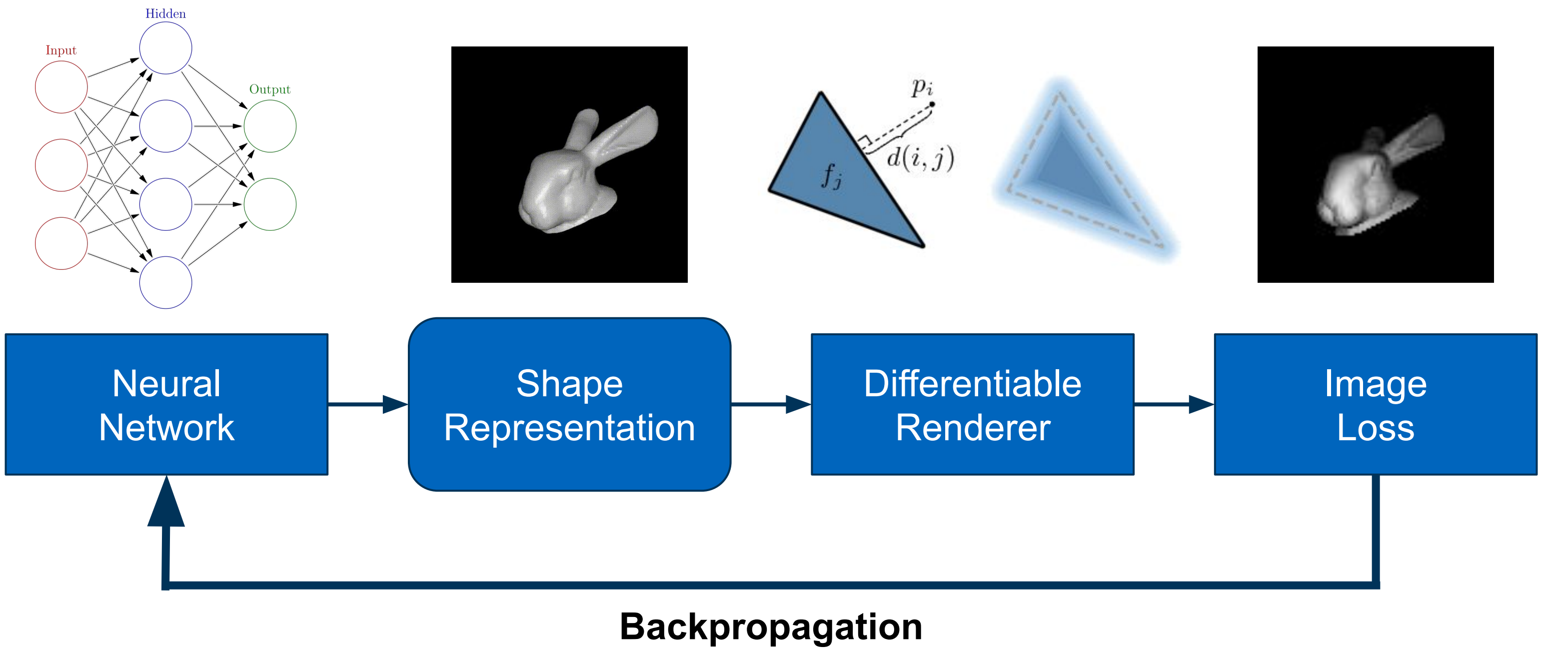


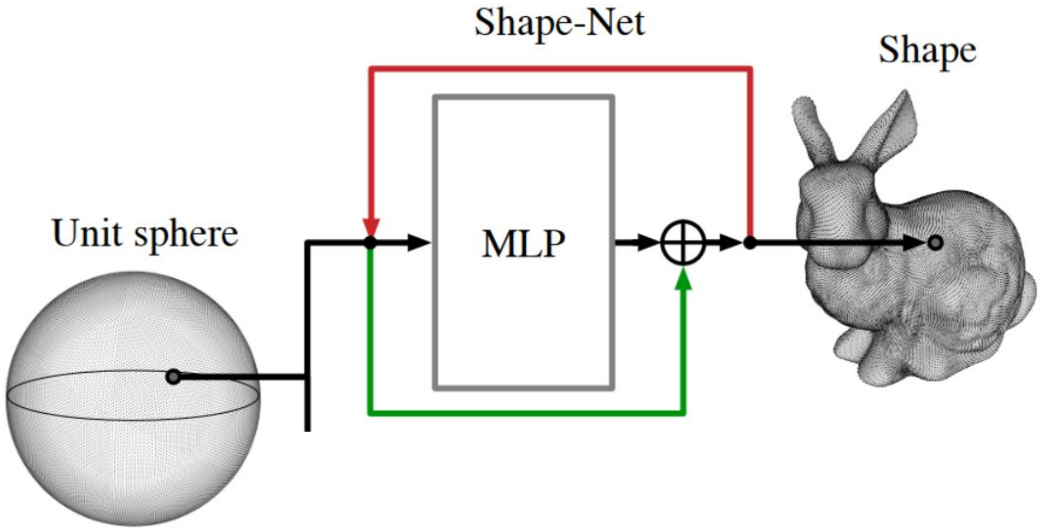
Image Source: *Hasenfratz, et al. (2004). "A Real-Time System for Full Body Interaction with Virtual Worlds"*

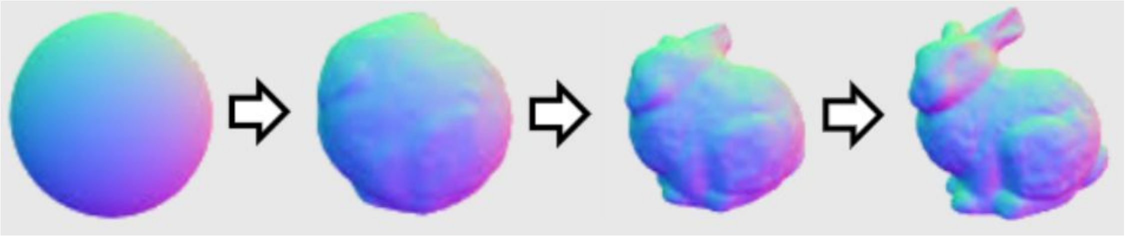
Deep Learning Approach



Diffeomorphic Neural Surface Parameterization

Model shape as a velocity field

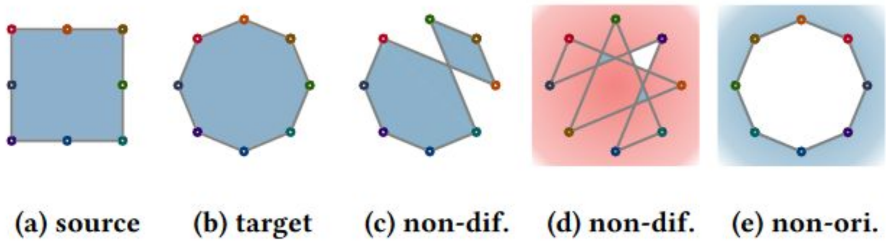




Deform Unit sphere by Integration over *t* timesteps

Cheng et al. “Diffeomorphic Neural Surface Parameterization for 3D and Reflectance Acquisition”

Diffeomorphic Neural Surface Parameterization



+

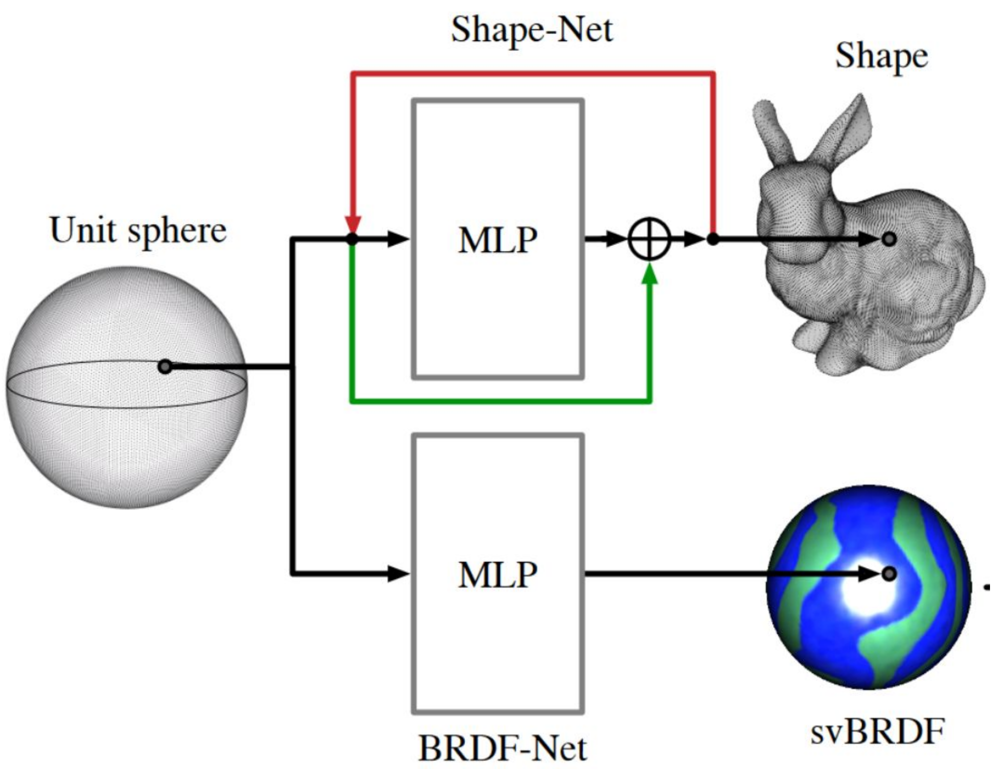
- ❖ Guarantee closed mesh
- ❖ Well-defined inside and outside

−

- ❖ Topological restriction
- ❖ <u>Restricted to trivial (black) background</u>

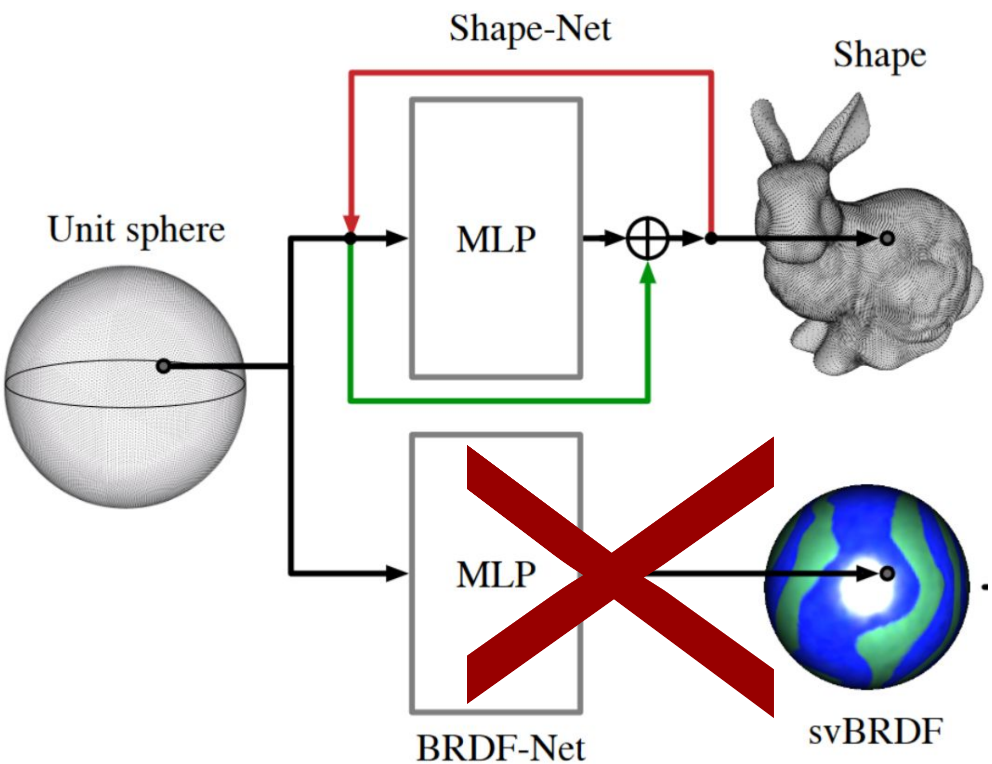
The original architecture

Separate modeling of shape and color data



Our Process

Reduce Computational Requirements



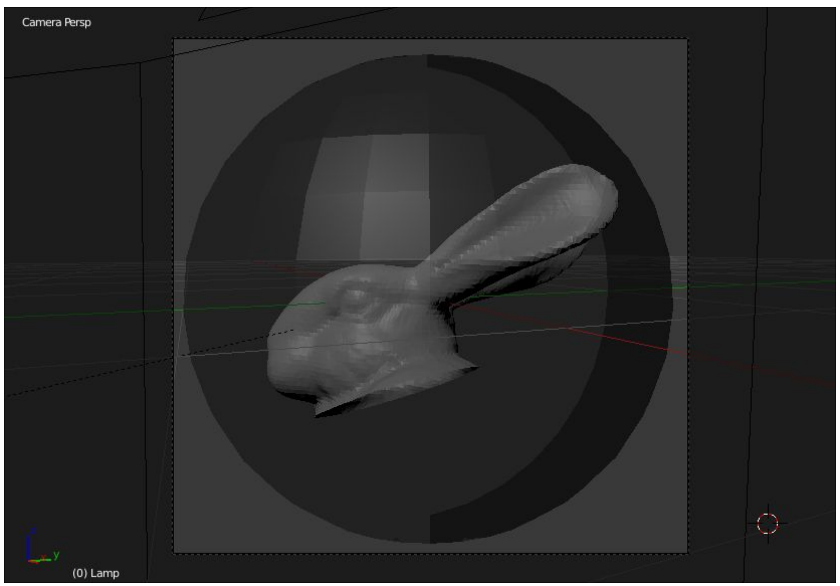
1. Simplified Renderer
2. Custom Dataset
3. Training adaptations for NeRF
4. Non-Trivial backgrounds

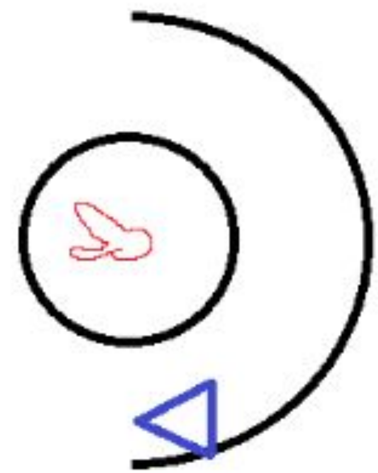
Dataset Creation

Use of Blender Software

Bunny occupies a unit sphere, while still being strictly inside, as the geometry is initialized with the unit sphere

30 different keyframes focusing on the front part of head



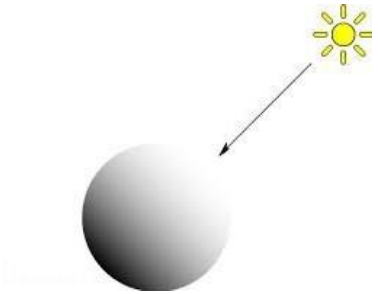


Red: object
blue: camera

Dataset Creation

Image Renderer

- Soft Rasterizer, based on original work
- Exclude specularities
- Diffuse_albedo: White
- Point Light source



Verification of existing code

Original work:

$$\mathcal{L}_{total} = \mathcal{L}_{image} + \mathcal{L}_{silhouette} + \mathcal{L}_{velocity} + \mathcal{L}_{normal\_consistency}$$

$$+\mathcal{L}_{edge} + \mathcal{L}_{laplacian\_smoothing}$$

Our aim:

$$\mathcal{L}_{total} = \mathcal{L}_{image} + \mathcal{L}_{velocity}$$

❗ **Training does not converge => Our approach fails**

Add Silhouette loss

Verification of existing code

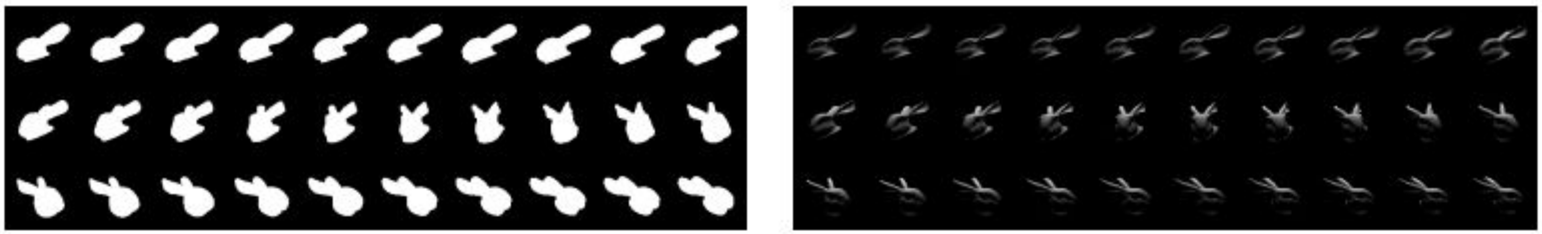
Silhouette Renderer

Renderer using a soft silhouette shader

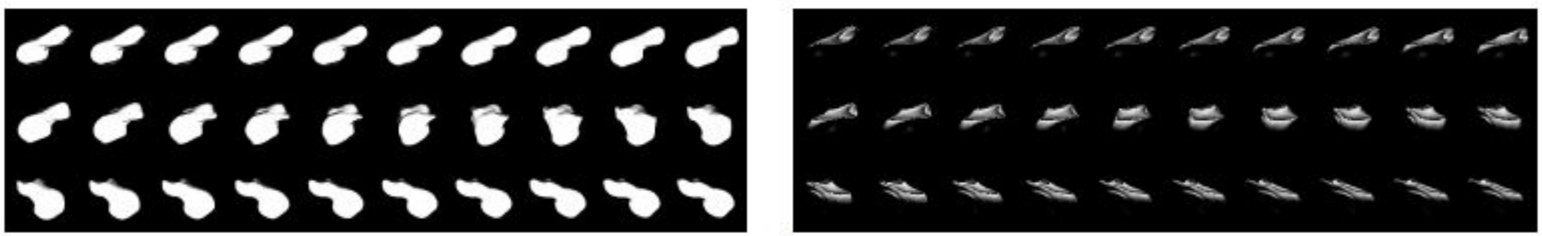
Verification of existing code

Image and mask loss

Initial tries to find out why our code fails to reconstruct the bunny



(a) Ground truth



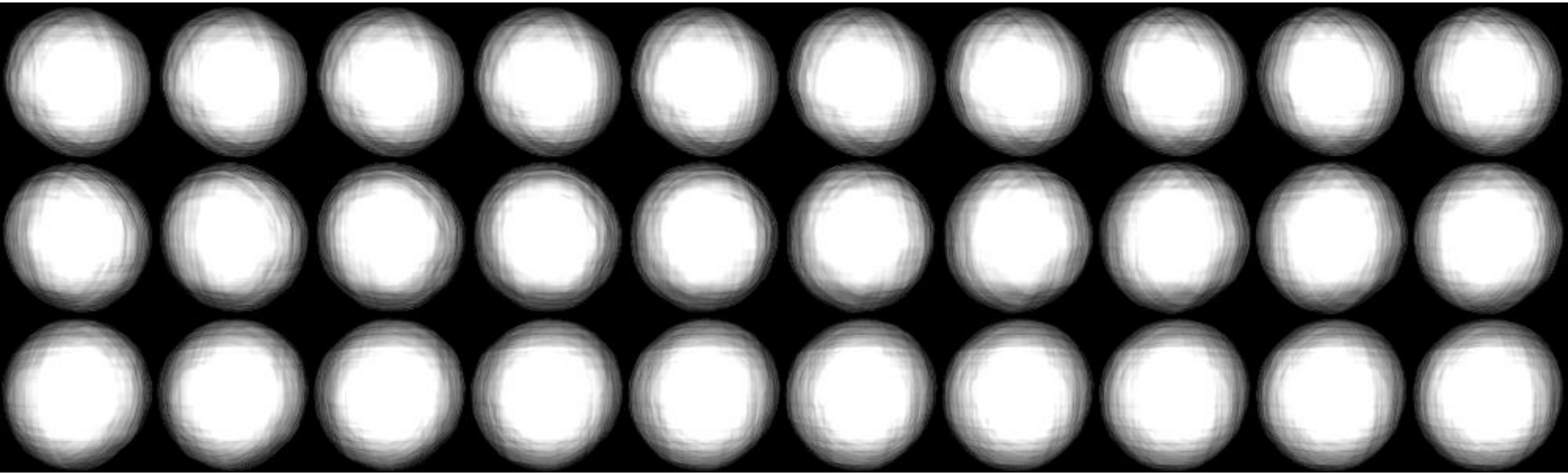
(b) Mask loss only

(c) Image and mask loss

Verification of existing code

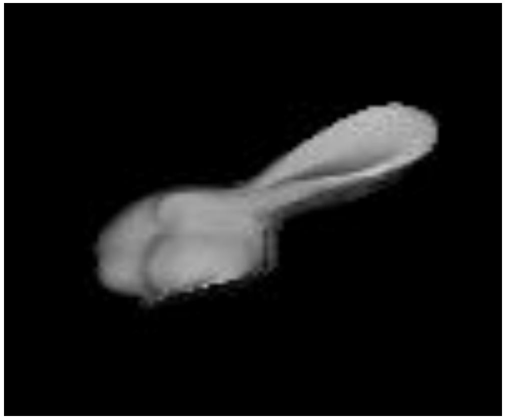
Fixed error:

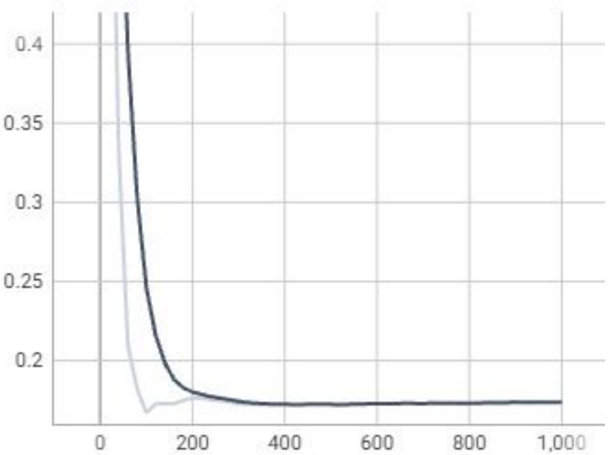
- Point light source was not positioned at the camera point but instead above the object
- Reduced learning rate

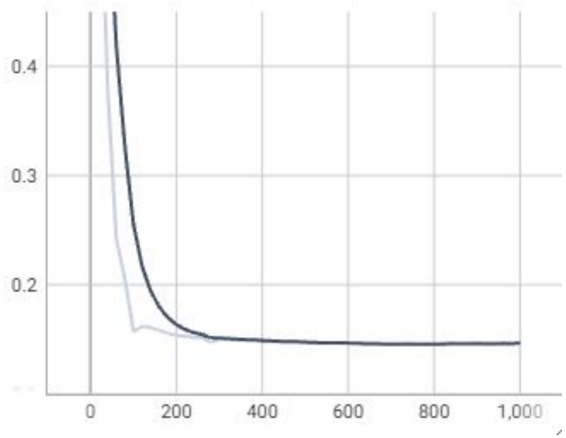


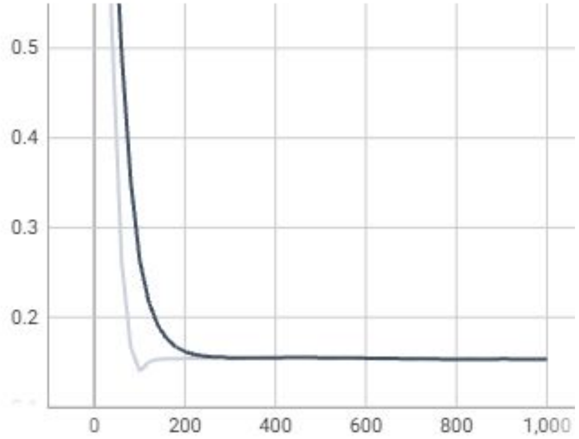
New Results with Image_loss + Velocity_loss

Validation Sets









Loss = 0.01895

Loss = 0.01959

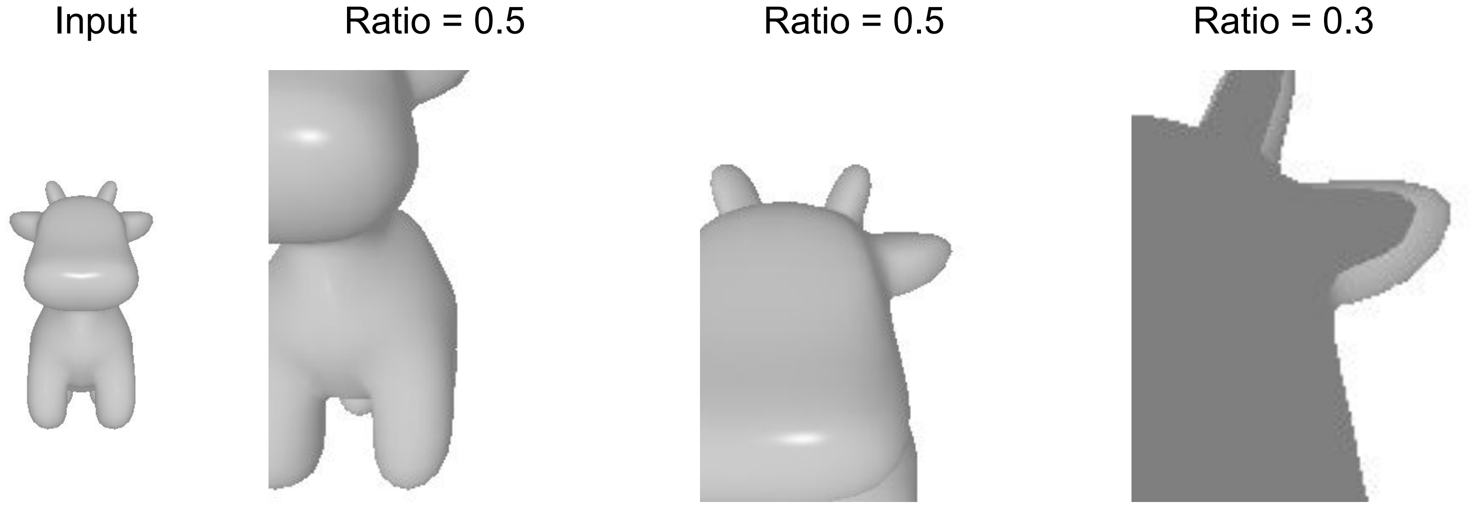
Loss =0.01483

Training with Randomly Cropped Images

Main idea : NeRF => shoot a ray at every single pixel of the rendered image

Goal: Reduce computational cost

Approach: Multiplying the z-axis of the translation matrix (T) with the crop ratio



Final Approach: crop a random 2D offset value on the rendered ground truth and predicted mesh

Training with Randomly Cropped Images

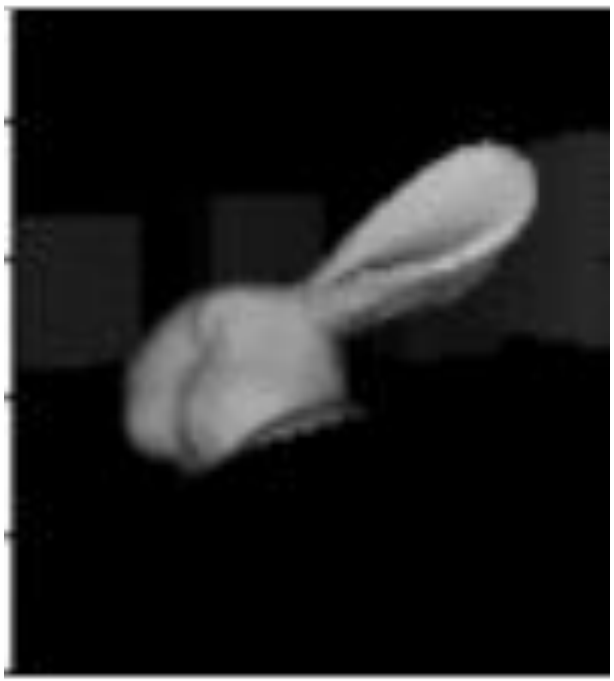
Ground Truth

Trained without cropping

Trained with cropping

Add non-trivial Background

White cubes as a background
Try to reconstruct the mesh with a background

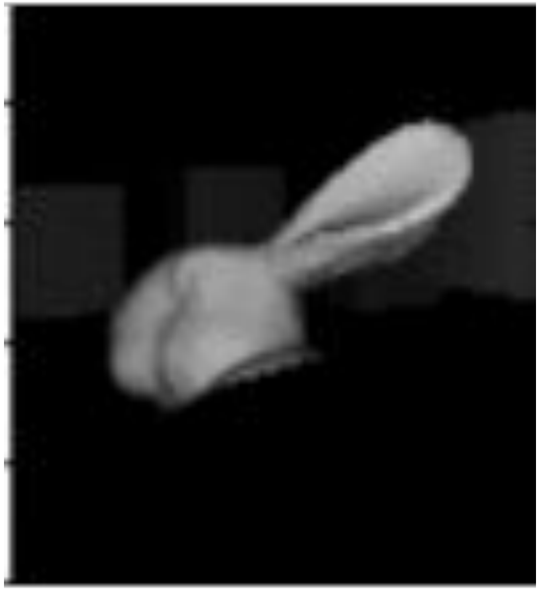


Preliminary Tests for NeRF

Main idea: Use a NeRF to obtain the background color

First check: use of the ground truth background as input to our rendering function

The loss only tries to reconstruct the bunny



Experiments with different background setups

Ears never have a very high contrast to the background =>reconstruction of ears fails

Approach : No background behind ears

Experiments with different background setups

Missing viewpoints behind the bunny => reconstruction of ears fails

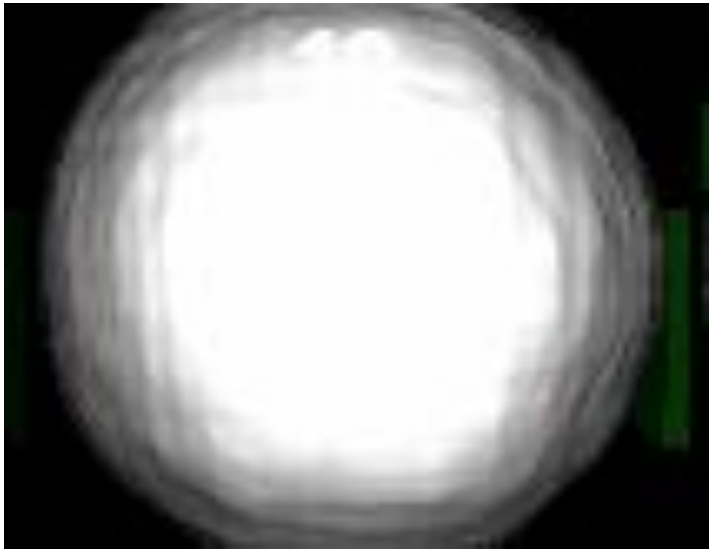
Approach:Full rotation around the head

Experiments with different background setups

Object and the background same color => difficult to be distinguished

Approach: colored cubes, white bunny mesh

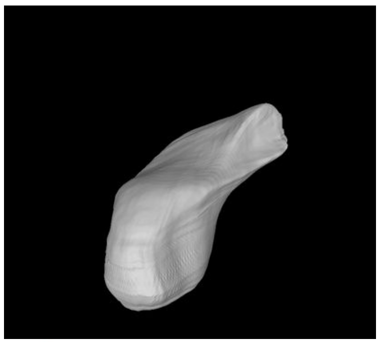
Effect of Early Stopping on the Results: Training

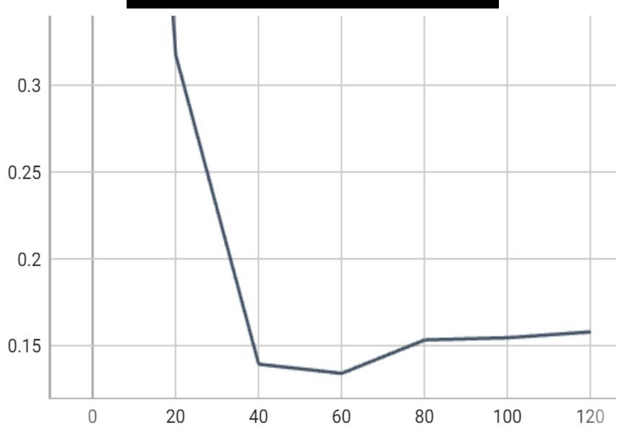


- After some time, there is no progress

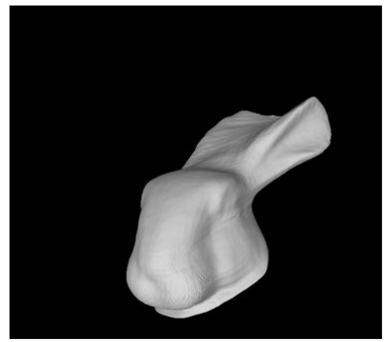
Effect of Early Stopping on the Results

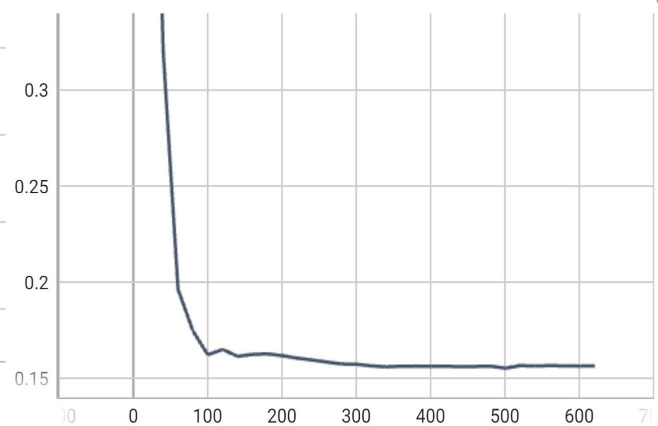
120 iterations
~ 12 minutes



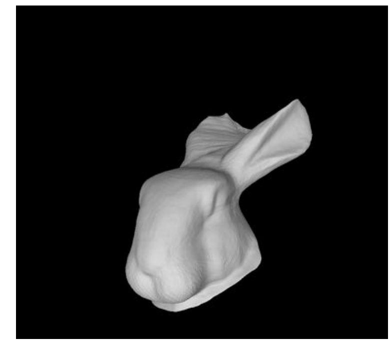


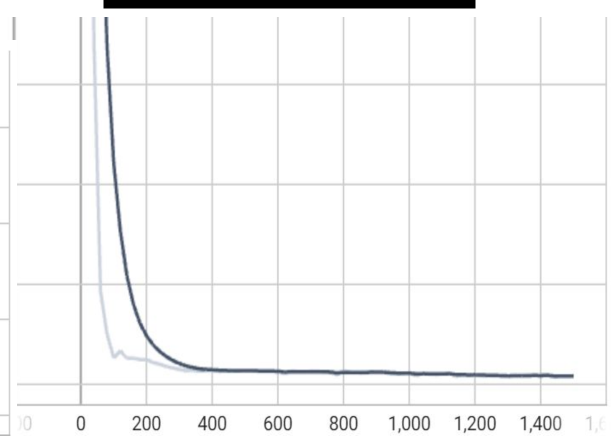
620 iterations
~ 1 hour





1500 iterations
~ 2 hours 30 minutes

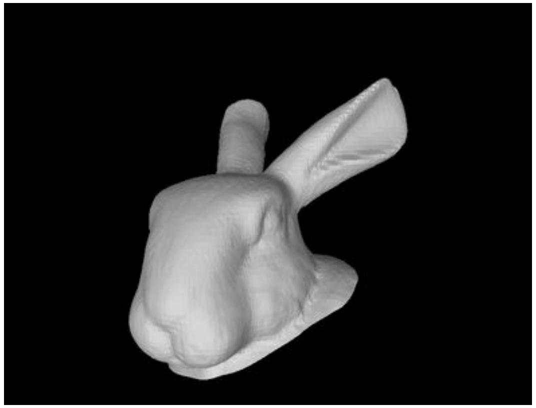




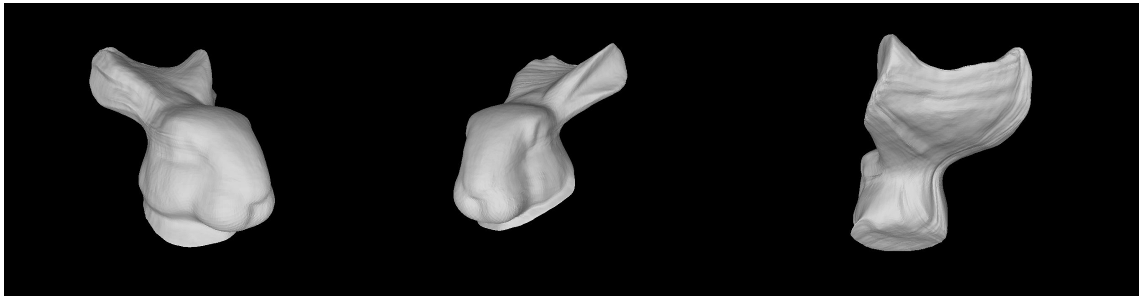
- **Challenge:** details around the eyes and the ears

Effect of ShapeNet's Size on the Results

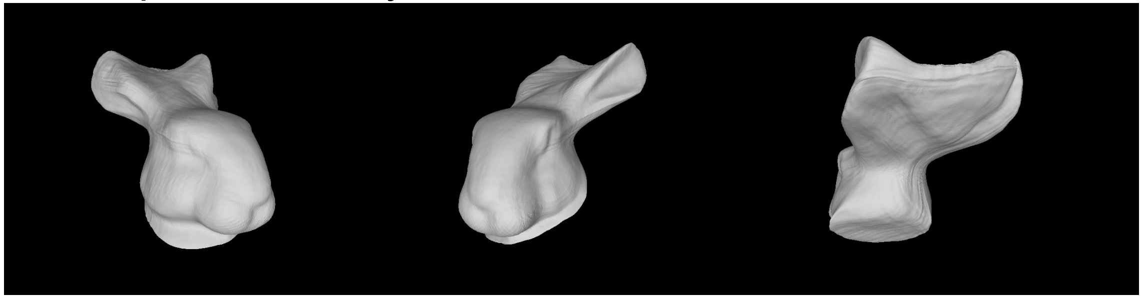
- **Groundtruth**



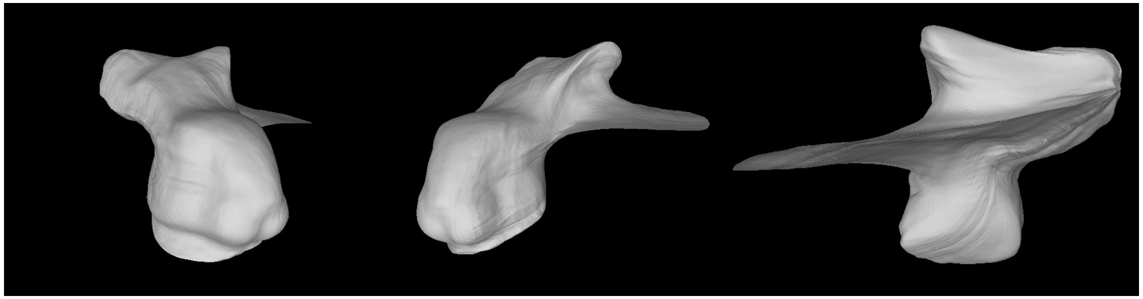
- ShapeNet with layers **256,256,256,3**: ~ 1 hour 40 minutes



- ShapeNet with layers **128,128,128,3**: ~ 1 hour 30 minutes



- ShapeNet with layers **64,64,64,3**: ~ 1 hour

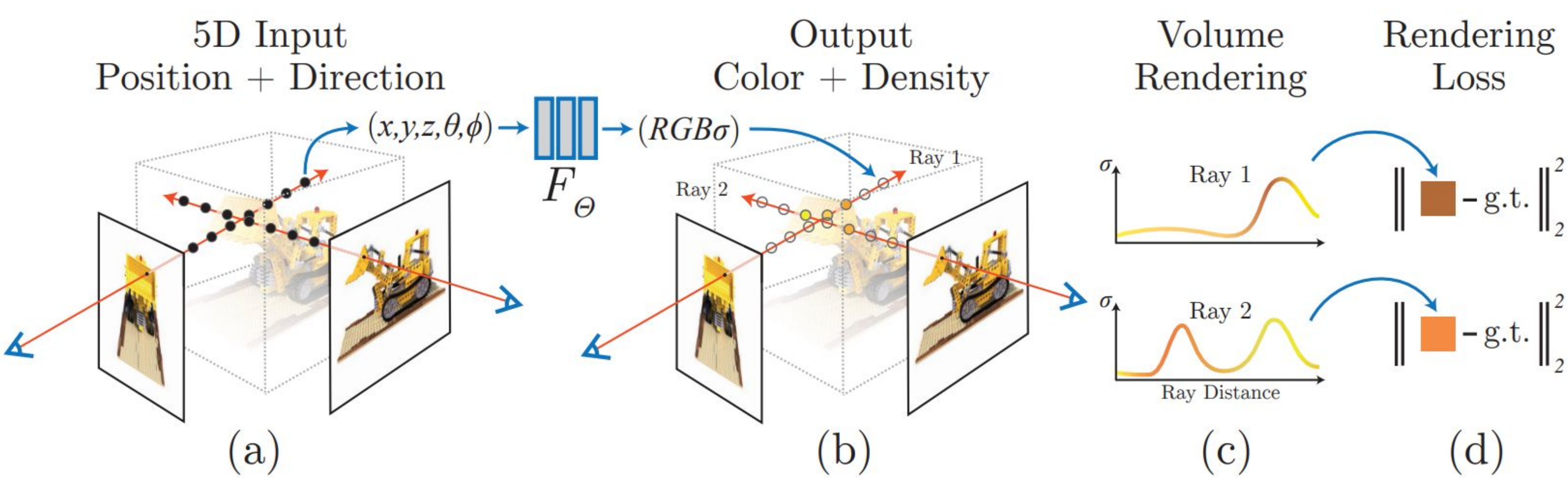


Future Work: NeRF

- **What is NeRF?**
  - Synthesizing novel views of scenes given multi-view images
  - Scene representation as continuous 5D function
    - **Input:** spatial location (x,y,z) and viewing direction (θ, φ)
    - **Output:** volume density and view-dependent emitted radiance

Neural Radiance Field (NeRF)

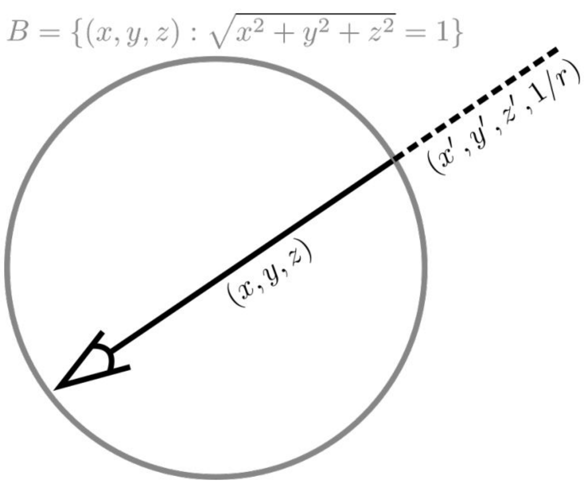
- **Problem:** Assumption of bounded scenes
  - impossible to have high-resolution in the foreground and the background
- **Solution:** Model background and foreground separately

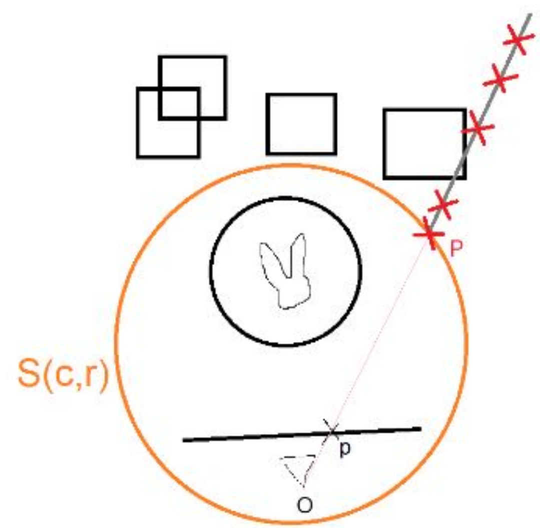


Mildenhall, Ben, et al. "Nerf: Representing scenes as neural radiance fields for view synthesis." *Communications of the ACM* 65.1 (2021): 99-106.

Future Work: NeRF++

- **Aim:** Learn the background and reconstruct the mesh simultaneously
- **How:** NeRF++ to model the background separately from the foreground i.e. two NeRFs
  - integral along the outer NeRF: inverted sphere parameterization
- Already implemented in our code using the existing VolSDF implementation[1] and is ready to be integrated and tested





[1]:Yariv, Lior, et al. "Volume rendering of neural implicit surfaces." Advances in Neural Information Processing Systems 34 (2021)
[2]:Zhang, Kai, et al. "Nerf++: Analyzing and improving neural radiance fields." arXiv preprint arXiv:2010.07492 (2020).)

Future Work: NeRF++

- Separation of foreground and background is crucial for our approach

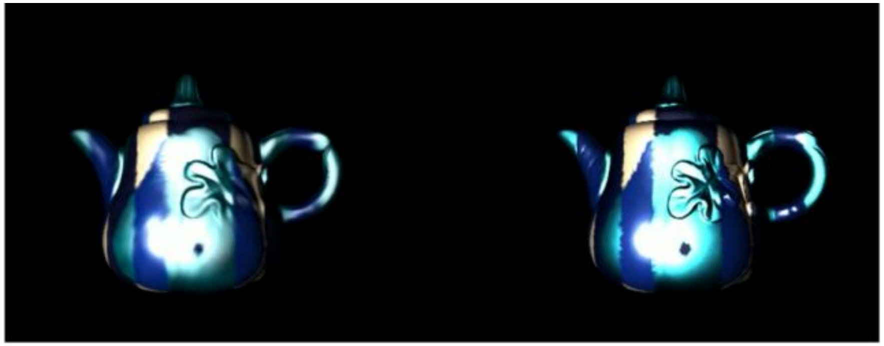
(a) bounding volume for the truck only (b) bounding volume for the entire scene

(a) NeRF++ prediction (b) predicted foreground (c) predicted background

Future Work: Complex Light Source

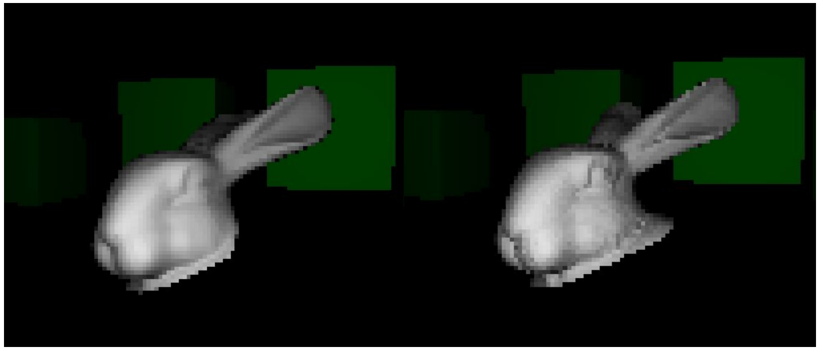
- **Aim**: Reconstruct objects with specular reflectance
- **How**: Use more complex light sources

The original approach:



DNS GT

Our approach:



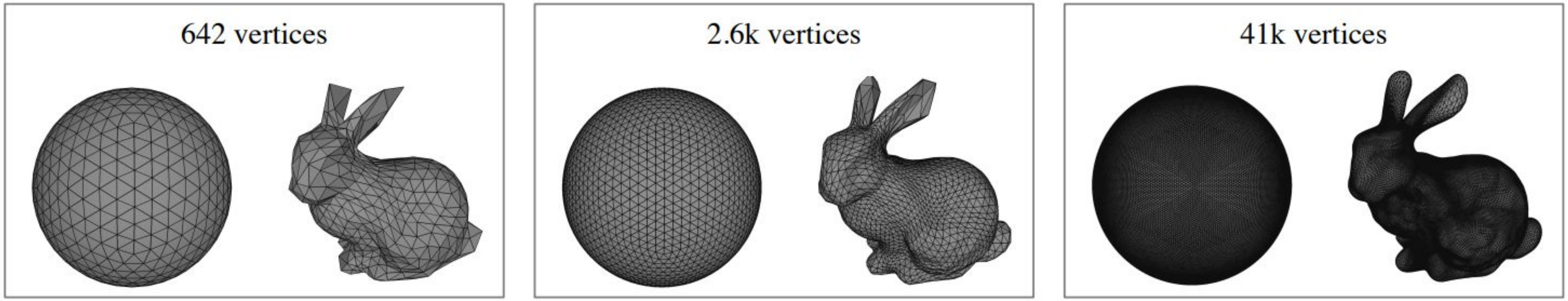
Ours GT

- **Specular** albedo and **diffuse** RGB color
- **Diffuse** RGB color

Cheng, Ziang, et al. "Diffeomorphic neural surface parameterization for 3d and reflectance acquisition." ACM SIGGRAPH 2022 Conference Proceedings. 2022.

Future Work: Improve Geometric Details

- Improve the geometric quality
    1. Use higher resolution images
        - Coarse-to-fine strategy: gradually increase rendering resolution or the sampling rate

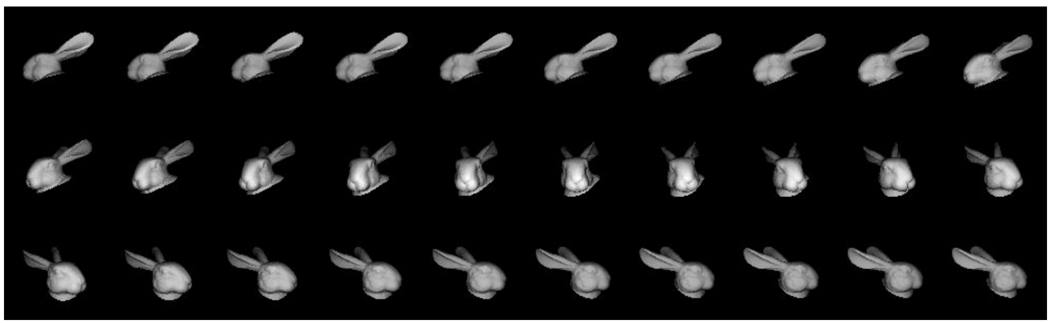


Cheng, Ziang, et al. "Diffeomorphic neural surface parameterization for 3d and reflectance acquisition." ACM SIGGRAPH 2022 Conference Proceedings. 2022.

Summary

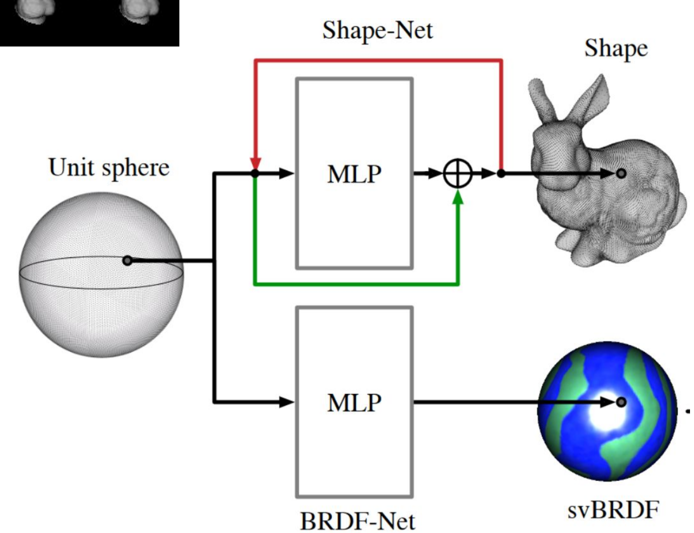
Initial Goal:

Investigate background handling in mesh reconstruction approach



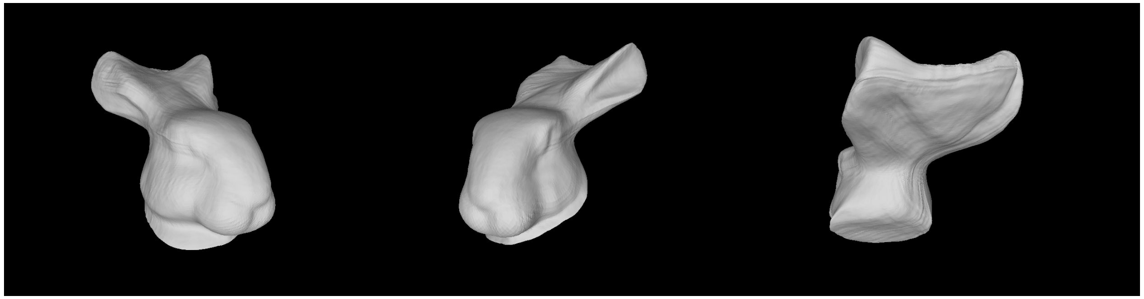
Custom Datasets

Simplified Approach

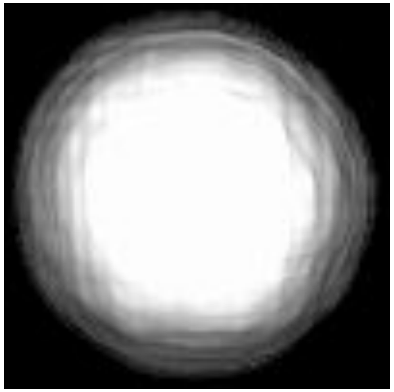


Summary

Convincing reconstruction with limited dataset coverage



Result:
- Implementation allows input of separately modeled background



Further Work required:
- Background modeling (e.g. via NeRF)
- Reintroduction of light complexities (BRDF-Net)