3D OBJECT RECONSTRUCTION FROM 2D IMAGES

- BY
TEAM PAST THE PIXELS

- UNDER
PROF. CRYSTAL MAUNG

AGENDA

Introduction

Objective

Dataset

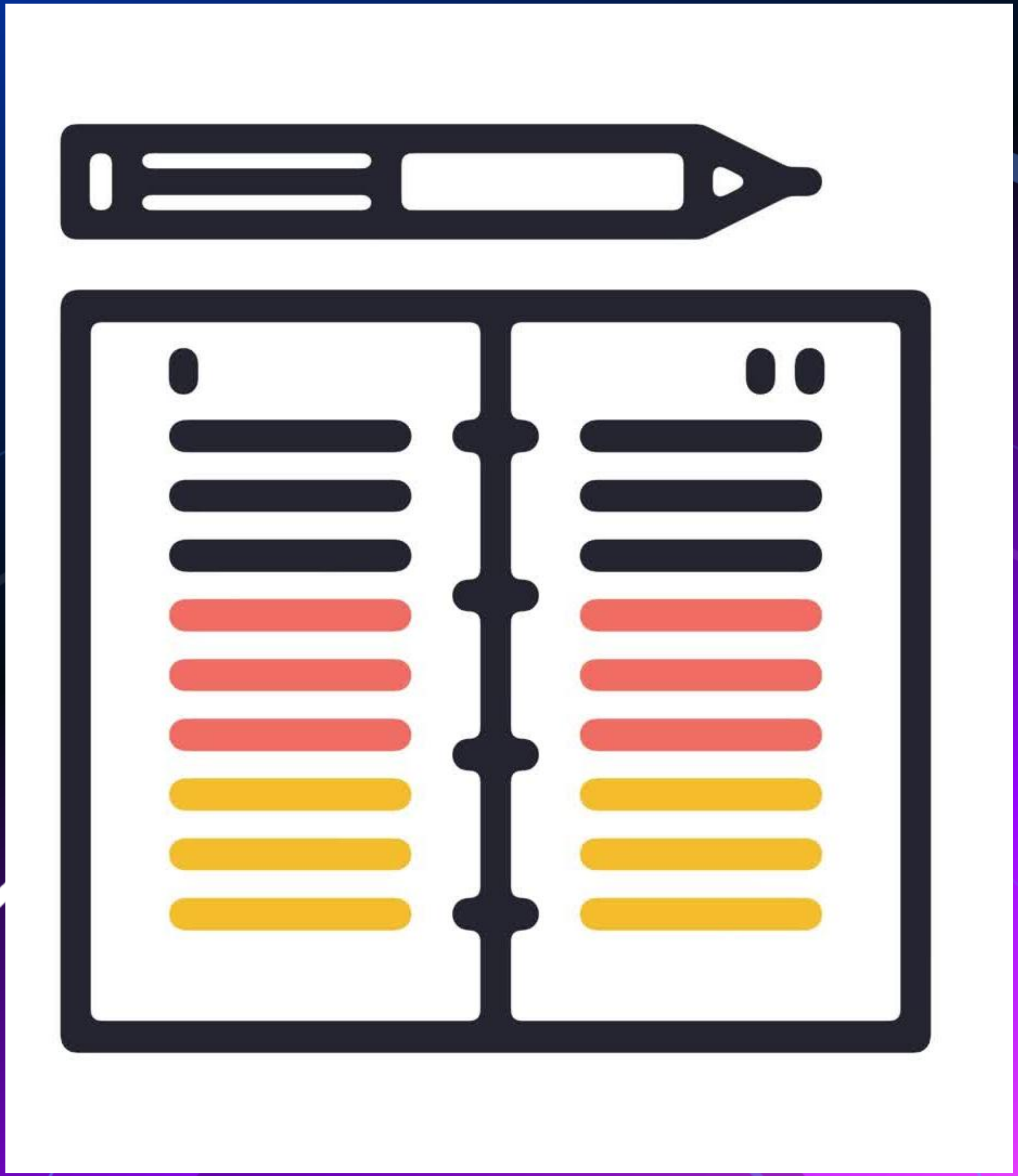
Methodology

Implementation

Key Insights

Applications

Conclusion



INTRODUCTION

- Leveraged deep learning techniques to reconstruct high-fidelity 3D models from 2D images
- Transformed 2D representations into 3D objects is crucial for applications like gaming, simulations, and augmented reality.
- Used methods like voxel grids, point clouds, and mesh generation, making the process more efficient and accessible.
- Traditional methods of creating 3D models require extensive manual effort and technical expertise.
- 3D object reconstruction plays a transformative role across industries like gaming, AR/VR, animation, and healthcare.
- Achieve fine-grained reconstruction of details like fingers, facial features, and clothing textures.

OBJECTIVE

- Recent advances in image-based 3D reconstruction have enabled applications like virtual avatars, gaming, and AR/VR.
- PIFuHD introduces a novel multi-level architecture for generating high-resolution, realistic 3D models from a single image.
- Leverage PyTorch3D to dynamically deform a simple 3D source shape (sphere) into complex, user-defined objects.
- Mesh deformation allows reshaping 3D models to match target objects by minimizing geometric and structural differences.

2D VS 3D VIEW

	2D	3D
Full Form	Two-Dimensional	Three-Dimensional
Definition	Represents an object with just two dimensions, i.e. length, and height.	Represents an object with three dimensions: Length, width, and height
Representation	Flat	Life-like
Aspects	Length, and height, no depth (width).	Length, width, and height
Mathematics	The x-axis and y-axis.	The x-axis, y-axis and the z-axis.
Geometry	Rectangle, square, triangle, polygon, etc.	Cylinder, sphere, cube, pyramid, prism, etc.

DATASETS

Render People Dataset:

- High-resolution photogrammetry scans of clothed humans.
- Synthetic backgrounds using the COCO dataset for robustness against varied environments.

Data Augmentation:

- Random rotations, translations, and background changes to improve generalization.

Dataset for the objects are taken as the input from the user and image converted to obj file.

METHODOLOGY

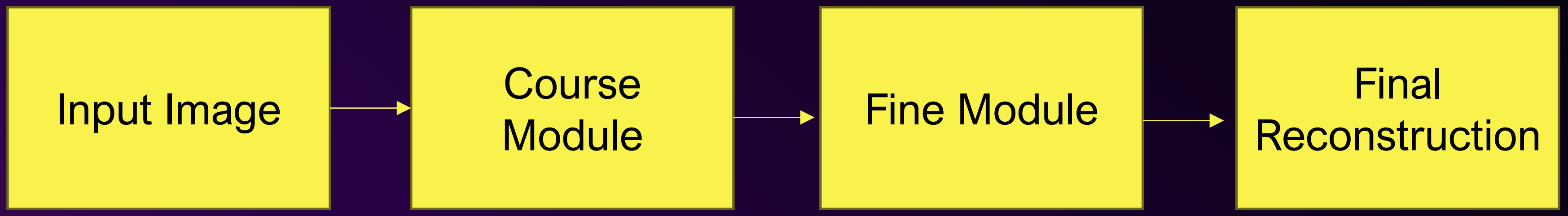
Human 3D Model

PIFuHD - "Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization“. It predicts 3D occupancy for any given point in the image space, allowing pixel-level alignment for better detail preservation

Architecture:

Coarse Module: Processes down sampled images to capture global 3D structure.

Fine Module: Uses high-resolution features to refine and enhance local details.



METHODOLOGY

Steps for Human 3D Model:

1. Input Preparation

- A 2D image of a person is provided as input, resized to 512x512 pixels for consistency.
- A segmentation mask isolates the person from the background, ensuring only the subject is processed.

2. Feature Extraction

- The model uses convolutional neural networks (CNNs) to extract pixel-aligned features, preserving spatial relationships.
- These features capture global details (body structure) and local details (clothing folds, facial features).

METHODOLOGY

3. Implicit Function Mapping

- Each pixel in the 2D image is mapped to a 3D coordinate (x, y, z) and a surface occupancy value (on/off the 3D surface).
- A hierarchical approach refines this mapping, starting with a rough geometry and adding finer details.

4. Depth and Surface Reconstruction

- The model predicts depth (z-coordinates) for each pixel, creating a 3D representation of the subject.
- It combines depth and pixel features to generate a detailed 3D surface mesh.

METHODOLOGY

5. 3D Model Refinement

- The generated 3D mesh is smoothed and cleaned to remove inconsistencies.
- Optional textures or colors are added to make the model more realistic.

6. Rotation and Visualization

- The 3D model is rotated using mathematical transformations (e.g., rotation matrices) for viewing.
- Tools like Open3D or Blender are used to visualize the model interactively.

The final output is a high-resolution 3D mesh saved in .obj format.Applications include animation, gaming, VR/AR, and digital avatars.
The main advantage of this approach is that it helps overcome challenges like depth ambiguity and missing information.

METHODOLOGY

Object 3D Model

Step 1: Input Handling

- Path of .obj file

Step 2: Preprocessing

- Normalize the target 3D mesh to fit within a unit sphere for effective optimization.
- Center the target object at the origin.

Step 3: Source Mesh Initialization

- Use an **icosphere** as the initial source shape for deformation.

METHODOLOGY

Step 4: Optimization

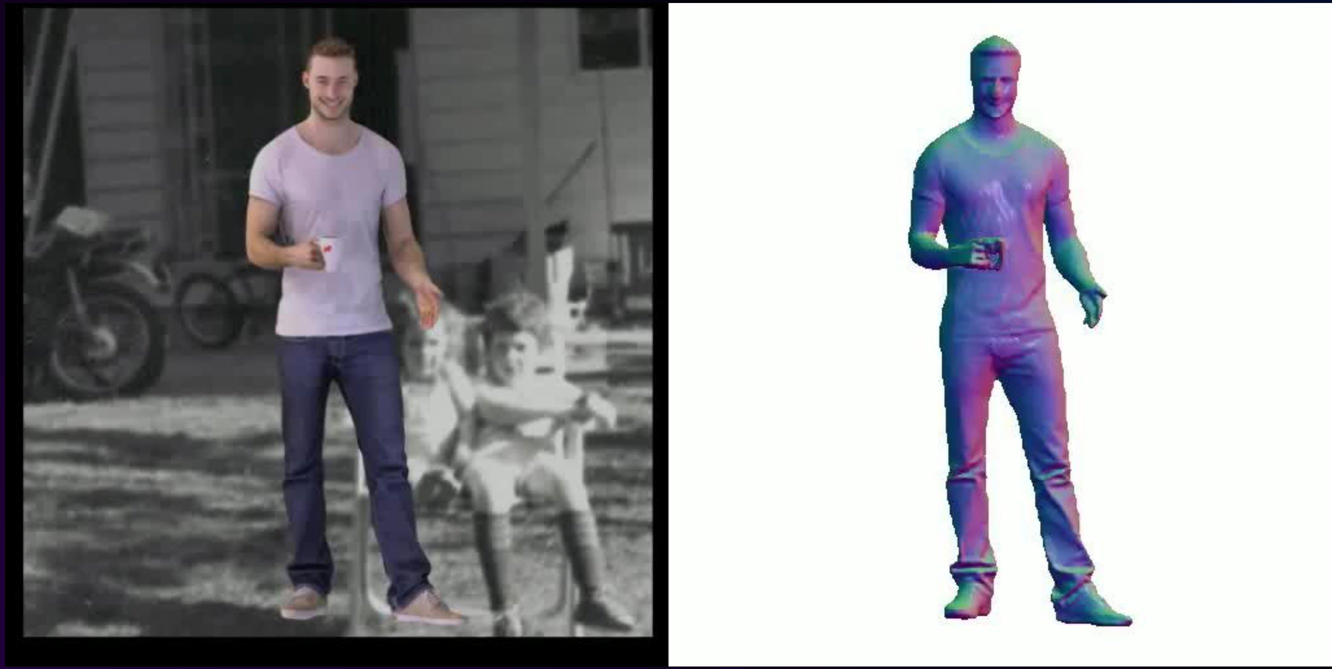
- Define trainable parameters for the source mesh vertices.
- Use **loss functions** to compute discrepancies between the source and target:
- **Chamfer Distance**: Measures geometric similarity.
- **Edge Loss**: Ensures edges maintain uniform length.
- **Normal Consistency**: Preserves smooth surface normals.
- **Laplacian Smoothing**: Maintains mesh smoothness.

Step 5: Results and Output

- Iteratively deform the source shape to match the target mesh and save the .obj file

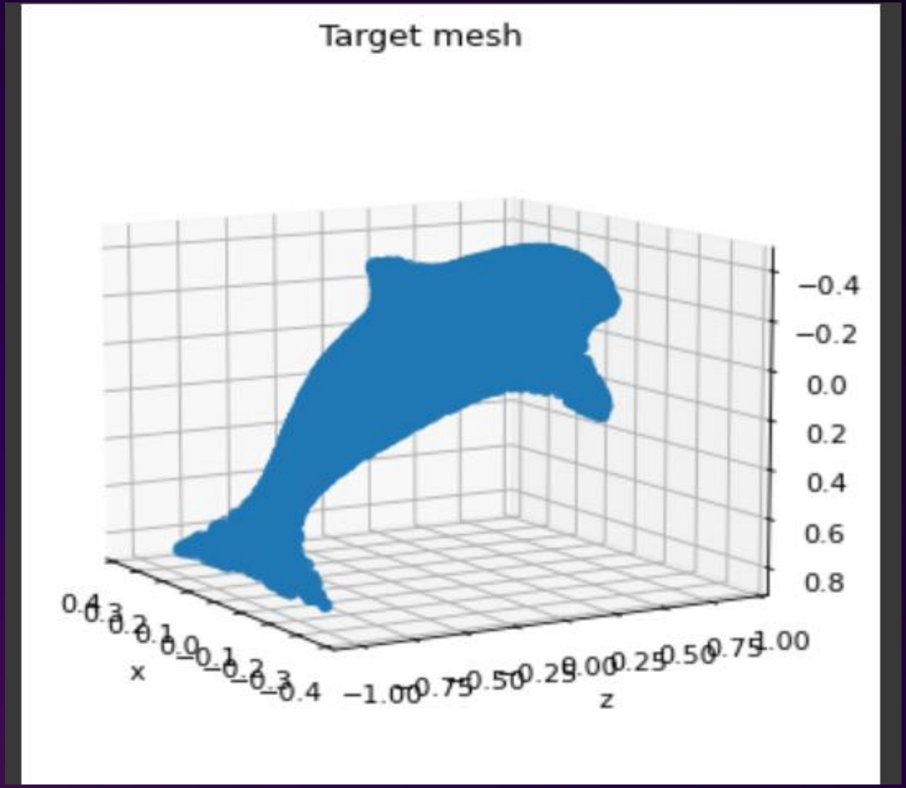
IMPLEMENTATION

1. Resolutions: Input resolution: 1k (1024x1024 pixels).
2. Feature resolution: Up to 512x512 in the fine module.
3. Framework: TensorFlow/PyTorch-based implementation.
4. Training Details: Loss Function: Extended Binary Cross-Entropy Loss.
5. Sampling Strategy: Gaussian perturbation near surfaces for sharp reconstructions.

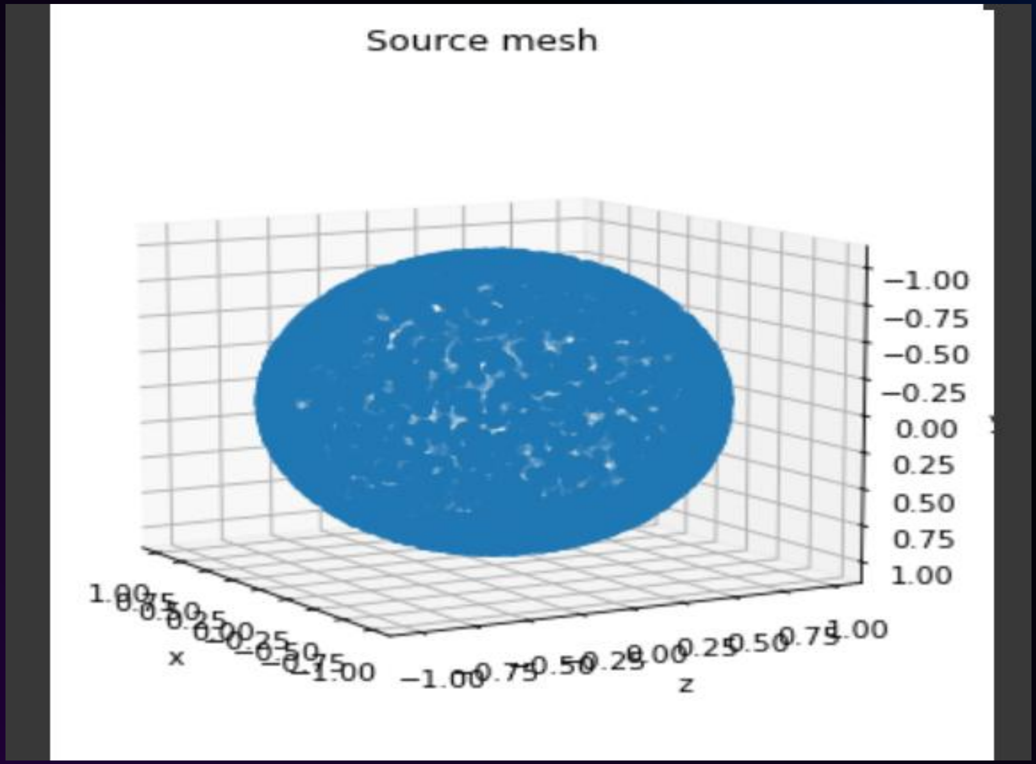


IMPLEMENTATION

1. Input Mesh

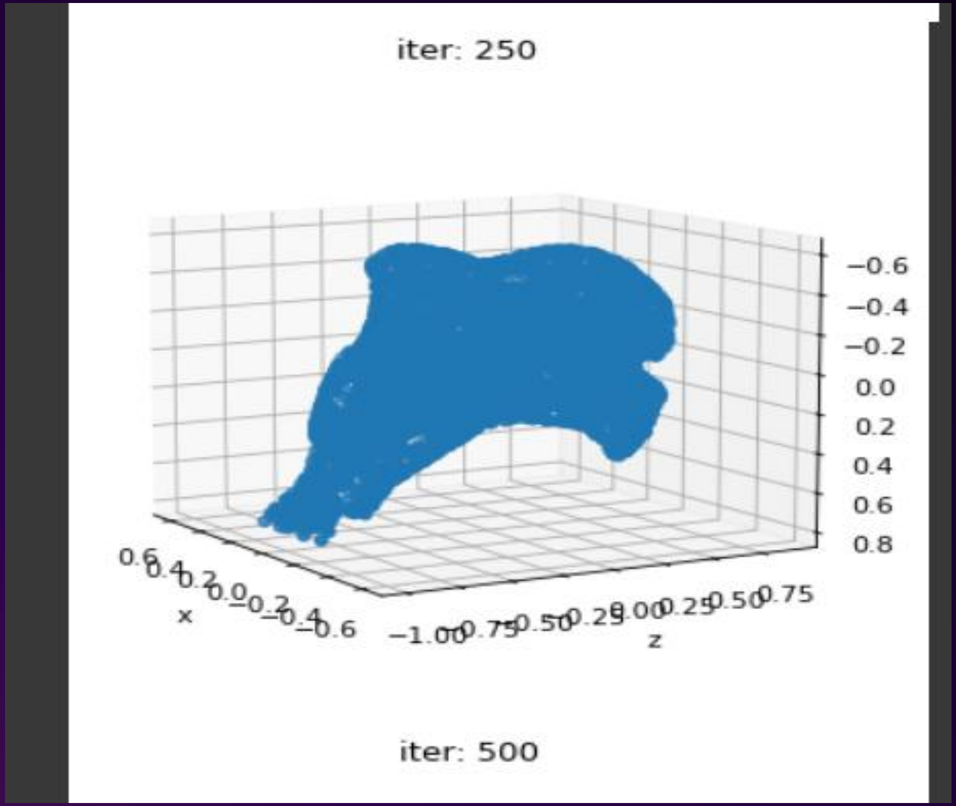


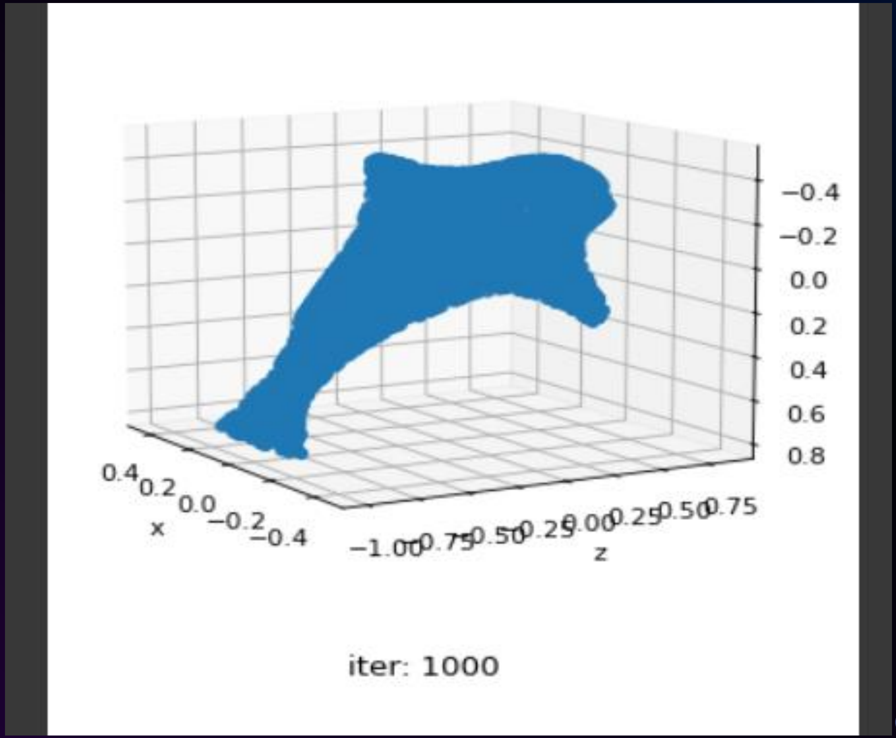
2. Source Mesh



IMPLEMENTATION

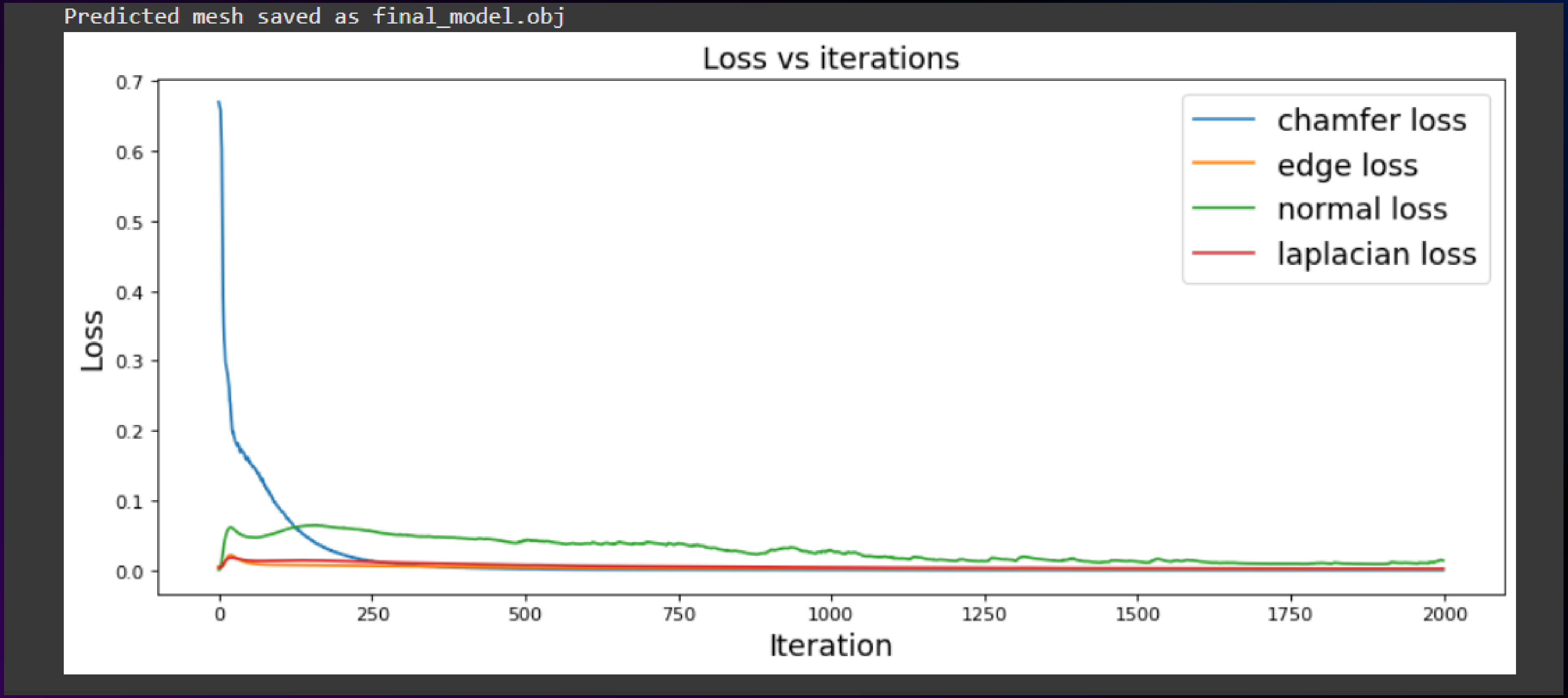
3. Optimization Process





IMPLEMENTATION

4. Loss Graphs



KEY INSIGHTS

Dynamic Mesh Deformation

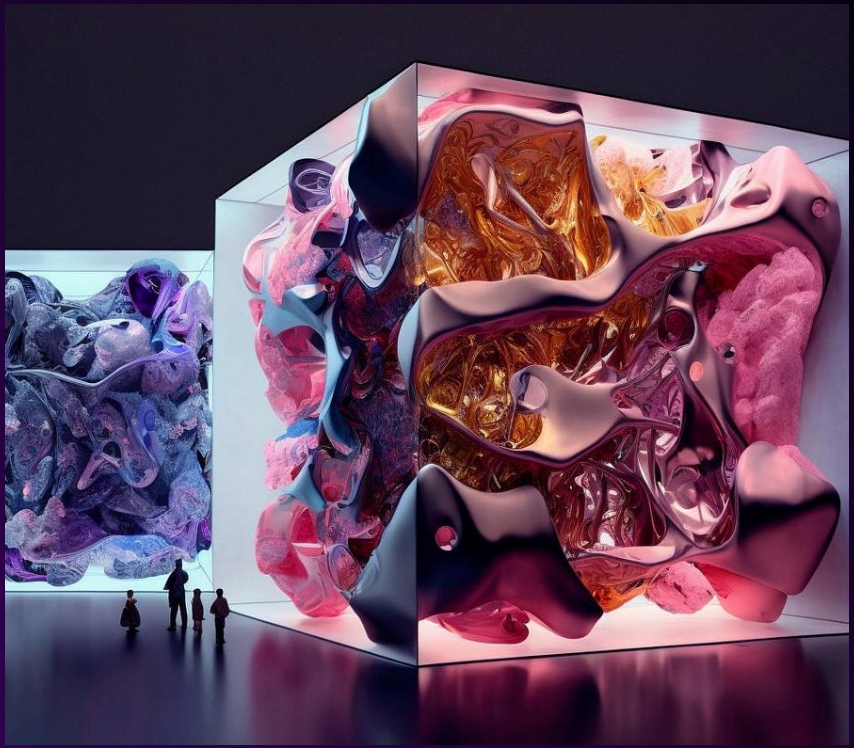
Optimization Techniques

User-Friendly Input

Real-Time Visualization

Scalability and Versatility

APPLICATIONS



1. Animation and Gaming
2. Medical Imaging
3. Product Design and Prototyping
4. Virtual and Augmented Reality (VR/AR)
5. Digital Art and Creative Industries
6. Education and Research
7. Robotics and Simulation

CHALLENGES

Memory Limitations: Handling high-resolution inputs (1k) requires significant computational resources.

Depth Ambiguity: Difficulties in reconstructing occluded body parts based solely on front-facing views.

Holistic vs. Local Reasoning: Balancing the need for global context and fine-grained detail extraction.

Approach to Challenges: Efficient memory usage through multi-level design. Using surface normal prediction for backside reconstruction.

CONCLUSION

The project demonstrates an efficient pipeline for transforming 2D representations into detailed 3D models using **PyTorch3D** and optimization techniques.

It highlights applications in gaming, AR/VR, and product design, offering a flexible solution for generating 3D content.

THANK YOU

Riya Badal Kapadia – rxk230132

Rajeswari Subramanian – rxs230174

Nandhana Suresh Kumar – nxs230139

Nivedha Shankar – nxs230138