3D Object Reconstruction using COLMAP

Pool Kshatriya

September 6, 2024

Abstract

This document outlines the process of 3D object reconstruction using COLMAP. The workflow involves capturing video footage, extracting frames, performing feature extraction and matching, and using Structure-from-Motion (SfM) and Multi-view Stereo (MVS) techniques to reconstruct 3D models. The final model is generated in `.ply` format and converted to `.obj` format using external tools such as MeshLab or Blender.

1 Introduction

3D object reconstruction is the process of generating a 3D model from a sequence of 2D images. This project leverages COLMAP, a powerful photogrammetry tool, to reconstruct a 3D model from video frames. The reconstruction process involves multiple stages, including frame extraction, feature detection, and dense reconstruction. This document also describes how to convert the resulting model from `.ply` format to `.obj` format, which is more widely used in various 3D software.

2 Software Requirements

The following software tools are required for the project:

- **COLMAP**: For 3D reconstruction.
- **OpenCV**: For extracting frames from the video.
- **Python**: To automate the frame extraction and preprocessing.
- **MeshLab** or **Blender**: For converting the final model from `.ply` to `.obj`.

3 Data Preparation

3.1 Video Input

The project begins by capturing a 10-second video of the object. The video is then processed using OpenCV to extract frames at regular intervals.

```python
import cv2
import os

def extract_frames(video_path, output_folder, interval=5):
    cap = cv2.VideoCapture(video_path)
    count = 0
    frame_idx = 0

    while cap.isOpened():
        ret, frame = cap.read()
        if not ret:
            break
        if frame_idx % interval == 0:
            cv2.imwrite(os.path.join(output_folder, f"frame_{count}.jpg"), frame)
            count += 1
        frame_idx += 1
    cap.release()

# Example usage
extract_frames('/path/to/video.mp4', '/path/to/output_frames')
```

Listing 1: Frame Extraction using OpenCV

4 Workflow Steps

4.1 Feature Extraction

Using COLMAP, we extract features (keypoints) from the frames to identify and match common points between different images.

```
colmap feature_extractor \
    --database_path reconstruction_output/database.db \
    --image_path output_frames \
    --ImageReader.single_camera 1
```

Listing 2: COLMAP Feature Extraction

4.2 Feature Matching

After extracting features, we match them across all frames. This step is critical for finding correspondences between images.

```
colmap exhaustive_matcher \
```

```
    --database_path reconstruction_output/database.db
```

Listing 3: COLMAP Feature Matching

4.3 Sparse Reconstruction (SfM)

In the next step, COLMAP's `mapper` is used to estimate camera poses and generate a sparse point cloud, representing the basic structure of the object.

```
colmap mapper \
    --database_path reconstruction_output/database.db \
    --image_path output_frames \
    --output_path reconstruction_output/sparse
```

Listing 4: Sparse Reconstruction (SfM)

4.4 Dense Reconstruction (MVS)

After the sparse reconstruction, dense reconstruction is applied to create a detailed 3D model of the object.

```
colmap image_undistorter \
    --image_path output_frames \
    --input_path reconstruction_output/sparse/0 \
    --output_path reconstruction_output/dense \
    --output_type COLMAP

colmap patch_match_stereo \
    --workspace_path reconstruction_output/dense \
    --workspace_format COLMAP \
    --PatchMatchStereo.geom_consistency true

colmap stereo_fusion \
    --workspace_path reconstruction_output/dense \
    --workspace_format COLMAP \
    --output_path reconstruction_output/final_model.ply
```

Listing 5: Dense Reconstruction using COLMAP

5 Converting `.ply` to `.obj`

The final model is output as a `.ply` file. To make it compatible with a broader range of software, we convert the `.ply` file to `.obj` format using either MeshLab or Blender.

5.1 Using MeshLab

To convert the `.ply` file to `.obj` using MeshLab:

- Open MeshLab and load the `.ply` file.

- Go to `File > Export Mesh As...`.
- Select `Wavefront (.obj)` and save the model.

You can also automate the conversion using `meshlabserver`:

```
meshlabserver -i reconstruction_output/final_model.ply \
              -o reconstruction_output/final_model.obj
```

Listing 6: Automated Conversion with MeshLabServer

5.2 Using Blender

To convert the `.ply` file to `.obj` using Blender:

- Open Blender and go to `File > Import > PLY (.ply)`.
- Once the model is imported, go to `File > Export > Wavefront (.obj)` to export the file.

6 Results

The final result is a 3D model of the object, reconstructed from video frames. The model was initially exported in `.ply` format and successfully converted to `.obj` for wider usability.

7 Conclusion

This project demonstrates the power of COLMAP for 3D reconstruction. By following this workflow, we created a detailed 3D model using video footage, and ensured compatibility with various software by converting the model to `.obj`. This approach is highly applicable for applications like virtual reality, gaming, and 3D printing.

8 References

- COLMAP Official Documentation
- OpenCV Documentation
- MeshLab Documentation
- Blender Documentation