A Flexible Pipeline for VR Video Creation

Anthony Dickson[1] Jeremy Shanks[1] Jonathan Ventura[2] Alistair Knott[3] Stefanie Zollmann[1]

[1]University of Otago [2]California Polytechnic State University [3]Victoria University of Wellington

Abstract

Recent advances in Neural Radiance Field methods have enabled high-fidelity novel view synthesis for video with dynamic elements. However, these methods often require expensive hardware, take days to process a second-long video and do not scale well to longer videos. We create an end-to-end pipeline for creating dynamic 3D video from a monocular video that can be run on consumer hardware in minutes per second of footage, not days. Our pipeline handles the estimation of the camera parameters, depth maps, 3D reconstruction of dynamic foreground and static background elements, and the rendering of the 3D video on a computer or VR headset.

Aims

- Convert monocular video to 3D Video
- Run on consumer grade hardware
- Run in reasonable amount of time
- Playback in VR

Related Work

- Soccer on your Tabletop [4]
  - Designed for broadcasts of football games
  - Viewable with AR Headset
  - Does not reconstruct the background

Figure 1. Example of the Soccer on your Tabletop [4] approach showing: the input video (left); the 3D reconstruction of the game (middle); and the view using an AR device. Figure adapted from the original paper [4].

- NSFF [2]
  - Space-Time Interpolation
  - Photo-realistic
  - High hardware and compute requirements

Method Overview

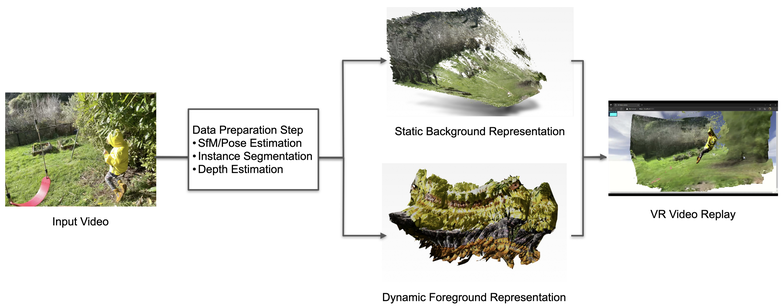

Figure 2. Our method consists of four main steps: data preparation; foreground reconstruction; background reconstruction; and rendering.

Data Preparation

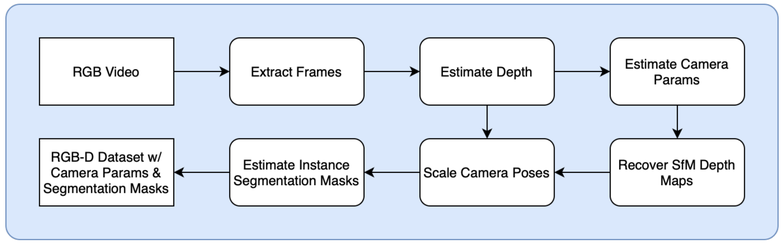

Figure 3. Overview of the data preparation process. We use the DPT model [3] for depth estimation, Detectron2 [7] for instance segmentation and COLMAP [5, 6] for camera parameter estimation.

Dynamic Foreground Reconstruction

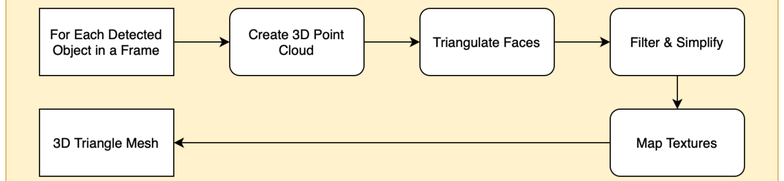

Figure 4. We reconstruct the foreground meshes for frames where at least one person was detected. Meshes and textures for each frame are merged/packed into a single mesh/texture atlas which are then saved to a single glTF formatted file. Each mesh is labelled with its frame index for use in the renderer.

Static Background Reconstruction

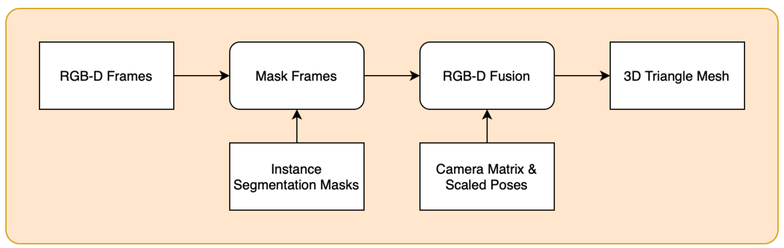

Figure 5. We reconstruct a static mesh for the background. The foreground in the frames are masked out with instance segmentation masks. We use a RGB-D fusion method [8] that takes camera poses as input because RGB-D fusion methods (e.g., BundleFusion [1]) use frame alignment algorithms that do not work well with estimated depth maps.

Rendering

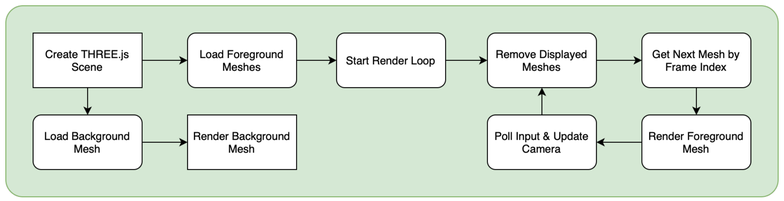

Figure 6. Overview of our cross-platform WebXR renderer.

Preliminary Results

Figure 7. Comparison of VR Video output from the TUM walking xyz sequence using: ground truth pose and depth (left); estimated pose and ground truth depth (middle); and estimated pose and depth (right).

Figure 8. Comparison of NSFF (left) and our method (right) on the running kid sequence.

	NSFF [2]	VR Video (Ours)
Processing Time	about 2 days	1 minute
GPU	4x RTX 2080 Ti	RTX 2080 Ti
GPU Memory	4x 10 GB	10 GB
Resolution*	512×288	640×360
Frame render time	6000 ms	< 16 ms

Table 1. Comparison of NSFF [2] compared to VR Video (Ours) in terms of hardware and compute requirements for the running kid sequence (75 frames). *For NSFF this the render resolution, for VR Video this is the frame resolution during processing. Our web based renderer renders at the resolution of the web browser window.

Acknowledgements

We gratefully acknowledge the support of the New Zealand Marsden Council through Grant UOO1724.

References

[1] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017.

[2] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021.

[3] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[4] Konstantinos Rematas, Ira Kemelmacher-Shlizerman, Brian Curless, and Steve Seitz. Soccer on your tabletop. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4738–4747, 2018.

[5] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

[7] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[8] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1802–1811, 2017.

dican732@student.otago.ac.nz