Here is an example of a Previous Work section using the Funnel Model. The context is Volumetric Video, which is a subset of 3D reconstruction and the research question is real-time volumetric video capture.

Previous Work

3D reconstruction has been an active research area for decades. It is a huge field encompassing computer vision but is also the last decades increasingly incorporated by computer-graphics researchers. Hence, we will make a very quick overview and then focus on the most related previous work.

There are several methods to sample an environment for 3D reconstruction. Sonar and ultrasound use acoustic waves and radar uses radio waves. CT scans (X-rays) and Magnetic Resonance imaging are both popular in medicine. However, we humans are typically not only interested in the shape of objects and at a proper level of detail, but also the objects' visual appearance, since the human visual system is based on a visible light spectrum. Hence, 3D-capture devices in our context are typically based on light sensors and arranged into cameras.

Plenoptic cameras often use special lens filters to capture a light field, from which novel views for small camera movements can be reconstructed [1]. Holographic capture requires highly specialized setups and small scenes and has computer-generated holography as a subfield [2]. However, these methods do not focus on geometric reconstruction but rather the visual result from novel very nearby viewpoints.

LIDAR sends a light beam in a set of directions, measuring the time- of-flight of the beam and typically takes minutes to scan a scene. IR cameras often rely on an infrared pattern being projected onto the scene. The shape can be computed from the pattern's distortion on the image plane. Time-of-flight cameras instead use pulsed IR light or phase-shift [3]. Either way, an important drawback is that the IR pattern is easily saturated by the background light. This means that it can only be used for relatively short distances (~4m), does not capture volumetric media (gas, smoke, fire), has problems with thin details and transparency, and it typically does not work for outdoor environments due to the background sun light being too strong.

For time-of-flight cameras, an additional problem is erroneous results due to interference from several possible IR-light paths, often manifesting as large noise nearby corners and edges of the scene. Additionally, the result is often sensitive to the temperature of the camera [3] and could deviate as much as 10 cm on the Kinect v2 after 30 minutes [4] [5]. These cameras are also often more expensive and with lower image resolution compared to mass-produced RGB cameras. Nevertheless, Microsoft has built a real-time holoportation system on ToF cameras and up to 12 GPUs [6]. Since LIDAR and depth cameras mainly capture the geometry, the colors are often extracted from a coupled RGB image. View-dependent material effects are typically ignored (e.g., reflections and refractions), although AI-based solutions are rising for single-shot situations [7] [8].

For RBG-camera-based 3D reconstruction, depth can be computed via depth-from focus [9], depth from motion [10], stereo image pairs [11], or training neural networks to interpret depth from even a single RGB image [10] [12]. Passive stereo methods use two RGB images to compute depth maps via triangulation. Popular methods that achieve relatively good quality include PatchMatch [11], which has also been extended to real time [6] [13] [5]. For our project, however, we consider all these results as a lower level of reconstruction quality. In addition, stereo matching fails in the absence of distinct image details, e.g., left and right image both seeing a homogeneous white wall to mention an extreme example. Even moderate view-dependent color shifts can cause problems.

High-quality results can be achieved by combining RGB, active stereo in IR and silhouette information with Poisson surface reconstruction [14] to compute high-quality meshes for each

frame and track mesh deformations to handle topology changes [15]. Collet et al. [15] achieve compelling results but, like these types of methods, require and a large studio with green screens (for background removal) and large area-light sources for homogeneous background light. I.e., distinct light sources such as sun light or spotlights cause problems in the stereo-matching step. Their method is also computationally expensive; processing one frame on a machine with a dual 12-core processor and an AMD Radeon R9 200 GPU takes them 28.2 min. Again, view-dependent material appearance is not reconstructed.

Recently, important advances has been made in 3D capture for volumetric video by using a large set of RBG cameras processed by neural networks [16] [17] [18] [19] [20] [21] [22]. Many of these methods are referred to as Neural Radiance Fields (NeRF). One major advantage is the generality of the scenes that can be reconstructed, including volumetric materials (gas, smoke, fire), thin details like hair or laces, general light conditions, and materials even with complicated view-dependent effects such as the refractions and reflections in glass, water and otherwise glossy and/or transparent surfaces. A major drawback is their computational processing time (up to a day or days per frame using parallelization on 100-1000 cloud-CPU computers) and that they do not lend themselves for trivial parallelization to real-time performance.

Reconstructing the scene geometry from RGB images is an ill-posed problem [23]. Several solutions may exist. It is therefore natural that training an AI to learn by examples to penalize improbable solutions has shown to be a fruitful approach. The underlying idea is to optimize a volumetric representation in such a way that it matches the input images from their respective viewpoints.

Mildenhall et al. [17] use a 5D volumetric function described by a DNN, i.e., a 3D volume with 2D directional radiance in each 3D point. They use differential ray marching to compute the derivatives for the back propagation during the training. Their method is optimized by hierarchical sampling using two separate DNNs, each ray marched at separate granularity. Convergence in up to 300K iterations (1-2 days on a single high-end GPU). The number of required iterations prohibits any simple parallelization to real-time performance, since the trained weights would have to be communicated many thousands of times between the parallel nodes.

Broxton et al. [18] build a more practical system for outdoor capture on the similar principle but optimized for real-time rendering and capture for a rather small change in viewpoints. Each frame takes 28.5 CPU hours or 17 hours wall-clock time using some limited available parallelism.

Lombardi et al. [16] also use the same principle with a 5D volumetric function and differentiable rendering. However, their method uses the full video sequences to train the network in such a way that each frame improves the result of basically all other frames. This disables the method for real-time purposes since the solution inherently uses all frames to construct each frame. However, it could be interesting with a system using previous frames to improve future frames, e.g., using the principles of recurrent neural networks.

Our idea is based on the work by Lombardi et al. [16], Mildenhall et al. [17], and Broxton et al. [18]. We replace … with a …, which makes … Yada yada…

References

[1] G. Wetzstein and I. Ihrke and D. Lanman and W. Heidrich and K. Akeley and R. Raskar, Computational Plenoptic Imaging, ACM SIGGRAPH Course Notes, 2012.

[2] Y. Peng and S. Choi and N. Padmanaban and G. Wetzstein, , Neural Holography with Camera-in-the-loop Training, ACM Trans. Graph. (SIGGRAPH Asia), 2020.

[3] Giancola, S. and Valenti, M. and Sala, R., A Survey on 3D Cameras: Metrological Comparison of Time-of-Flight, Structured-Light and Active Stereoscopy Technologies, SpringerBriefs in Computer Science, Springer international Publishing, 2018.

[4] V. Kämpe, S. Rasmuson, M. Billeter, E. Sintorn, and U. Assarsson, Exploiting coherence in time-varying voxel data, " in Proceedings - 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D), 2016.

[5] Sverker Rasmuson, Erik Sintorn, Ulf Assarsson, A low-cost, practical acquisition and rendering pipeline for real-time free-viewpoint video communication, The Visual Computer, 2020.

[6] Orts-Escolano, S., Rhemann, C., Fanello, S., Chang, W., Kow- dle, A., Degtyarev, Y., Kim, D., Davidson, P.L., Khamis, S., Dou, M., Tankovich, V., Loop, C., Cai, Q., Chou, P.A., Mennicken, S., Valentin, J., Pradeep, V., Wang, S., Kang, S.B., Kohli, P., Lut, Holoportation: Virtual 3D teleportation in real-time., Proceedings of the 29th Annual Symposium on User Interface Software and Technology, UIST '16, pp. 741– 754. ACM, 2016.

[7] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, A. Bousseau, Single-image SVBRDF capture with a rendering-aware deep network, ACM Transactions on Graphics, Aug., pages 1-15, 2018.

[8] J. Riviere, P. Gotardo, D. Bradley, A. Ghosh, T. Beeler, Single-Shot High-Quality Facial Geometry and Skin Appearance Capture}, ACM Trans. on Graphics, jul, vol 39, no 4, 2020.

[9] P. Grossmann, Depth from focus, Pattern Recognition Letters, Volume 5, Issue 1, Pages 63-69, 1987.

[10] Xuan Luo, Jia-Bin Huang, R. Szeliski, K. Matzen, J. Kopf, Consistent Video Depth Estimation, ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH), vol 39, no 4, 2020.

[11] M. Bleyer, C. Rhemann, C. Rother, Patchmatch stereo–stereo matching with slanted supportwindows., BMVC, 2011.

[12] J. Kopf, K. Matzen, S. Alsisan, O. Quigley, F. Ge, Y. Chong, J. Patterson, J.M. Frahm, Shu Wu, Matthew Yu, Peizhao Zhang, Zijian He, P. Vajda, A. Saraf, M. Cohen, One Shot 3D Photography, ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH), vol 39, no 4, 2020.

[13] H. Nover, S. Achar and D. Goldman, ESPReSSo: Efficient Slanted PatchMatch for Real-Time Spacetime Stereo, 2018 International Conference on 3D Vision (3DV), pp. 578-586, 2018.

[14] Kazhdan, M., Bolitho, M., Hoppe, H., Poisson surface reconstruction, Proceedings of the Fourth Eurographics Symposium on Geometry Processing, SGP '06, pp. 61–70. Eurographics Association,, 2006.

[15] Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A., Sullivan, S., High-quality streamable free-viewpoint video, ACMTrans. Graph. 34(4), 69:1–69:13, 2015.

[16] Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y., Neural volumes: Learning dynamic renderable volumes from images., ACM Transactions on Graphics (SIGGRAPH 2019)., 2019.

[17] B. Mildenhall, Pratul P. Srinivasan, M, Tancik, J. T. Barron, Ravi Ramamoorthi, Ren Ng, , NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis., ECCV 2020., 2020.

[18] M. Broxton, J. Flynn, R Overbeck, D. Erickson, P. Hedman, M. DuVall, J. Dourgarian, J. Busch, M. Whalen, P. Debevec, Immersive Light Field Video with a Layered Mesh Representation, ACM Transactions on Graphics (Proc. SIGGRAPH), vol 39, no 4, pp. 86:1-86:15, 2020.

[19] Martin-Brualla, R., Radwan, N., Sajjadi, Mehdi S. M., Barron, Jonathan T., Dosovitskiy, A., Duckworth, D., , NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections, CVRP 2021, 2021.

[20] Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A., Local light field fusion: Practical view synthesis with prescriptive sampling guidelines, ACM Transactions on Graphics (SIGGRAPH), 2019.

[21] Sitzmann, V., Zollhoefer, M., Wetzstein, G., Scene representation networks: Continuous 3D-structure-aware neural scene representations, NeurIPS, 2019.

[22] Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, Pratul P. Srinivasan, Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields, arXiv:2103.13415v1 [cs.CV] 24 Mar., 2021.

[23] Petr Kellnhofer, Lars C. Jebe, Andrew Jones, Ryan Spicer, Kari Pulli, Gordon Wetzstein, Neural Lumigraph Rendering, arXiv:2103.11571v1 [cs.CV] 22 Mar 2021.

[24]

[25]

[26]

[27]

[28
]

[29
]

[30
]

[31
]

[32
]