Best of Both Worlds:
Merging 360° Image Capture with 3D Reconstructed Environments for Improved Immersion in Virtual Reality

by
Vijay Rajkumar

Senior Honors Thesis
Department of Computer Science
University of North Carolina at Chapel Hill
April 27, 2016

Approved:

Dr. Henry Fuchs,
Thesis Advisor

Dr. Jan-Michael Frahm,
Reader

Abstract

With the recent proliferation of high-quality 360˚ photos and video, consumers of virtual reality (VR) media have come to expect photorealistic immersive content. Most 360˚ VR content, however, is captured with monoscopic camera rigs and inherently fails to provide users with a sense of 3D depth and 6 degree-of-freedom (DOF) mobility. As a result, the medium is significantly limited in its immersive quality. This thesis aims to demonstrate how content creators can further bridge the gap between 360˚ content and fully immersive real-world VR simulations. We attempt to design a method that combines monoscopic 360˚ image capture with 3D reconstruction -- taking advantage of the best qualities of both technologies while only using consumer-grade equipment. By mapping the texture from panoramic 360˚ images to the 3D geometry of a scene, this system significantly improves the photo-realism of 3D reconstructed spaces at specific points of interest in a virtual environment. The technical hurdles faced during the course of this research work, and areas of further work needed to perfect the system, are discussed in detail. Once perfected, a user of the system should be able to simultaneously appreciate visual detail in 360-degrees while experiencing full mobility, i.e., to move around within the immersed scene.

1: Introduction

Recent breakthroughs in consumer virtual reality (VR) hardware, with the releases of the Samsung Gear VR, HTC Vive, and Oculus Rift all in the last year, have led to a growing demand for VR content. 360˚ video has become a particularly popular form of content due to ease of capture and deployment.

Monoscopic 360˚ camera rigs are readily available to videographers today. The Samsung Gear 360 and Ricoh Theta are low-cost, portable, one-click 360˚ capture systems that stitch the omnidirectional panorama on the devices themselves. Other 360˚ rigs are usually designed as mounts for multiple (usually wide-angle or fish-eye) cameras, with the intent for stitching in post-production. The Kodak SP360 is an example of a rig that is comprised of two fish-eye cameras, each capturing a field-of-view (FoV) of 235˚. Another popular spherical rig design is for six wide-angle GoPro Hero 4+ cameras, each with a 149.2˚ diagonal FoV. This rig design can easily be 3D printed from schematics available online and yields high-quality results due to the relatively large overlap between images, an ability to edit the panorama stitching in post-production with software like Kolor's AutoPano Pro, and the fact that each GoPro camera captures a high-resolution 12 MP image/2.7K video.

Once stitched, 360˚ content can be easily deployed to smartphones, virtual reality headsets, and social media platforms like YouTube and Facebook to be streamed by millions of viewers online. The simple process for capturing and sharing 360˚ content has made the medium popular among non-technical professionals and amateurs in the fields of journalism, cultural archival, entertainment (such as sports broadcast and live performance), and filmmakers exploring narrative-based "cinematic" virtual reality.

Yet, the goal of virtual reality, ultimately, is to provide the user with a fully immersive experience. Monoscopic 360˚ content falls far short in this regard. Though a user has full rotational freedom when viewing 360˚ content, he or she is unable to physically move around in the immersed environment. The moment the user attempts to move, the sense of immersion once experienced (due to the photorealistic image quality), is lost. The lack of 6 degree-of-freedom (DOF) mobility results in a less immersive experience than content rendered in a game engine, and could lead to motion sickness when the user moves but the image remains the same. Moreover, monoscopic 360˚ content fails to yield a sensation of 3D depth as the same image is rendered for both eyes when viewed through a headset. Omnidirectional stereo capture rigs exist but require elaborate and expensive camera arrays and may not be easy to use for novice users [1][2]. As a result, improving content acquisition systems for immersive real-world scenes is an imperative area of research for virtual reality.

There are a variety of ways we can reconstruct 3D real-world scenes that can be rendered in a game engine to enable a user with full 6 DOF mobility. While significant work in recent years has led to great improvements in geometry reconstruction, improvements in image quality have largely been ignored, and most 3D reconstruction pipelines output models with low-resolution vertex colors [3]. Ultimately, as visual details are lost, the poor image quality of many 3D reconstruction systems yields limited, and often ineffective, immersive experiences.

This thesis proposes the design of a system that capitalizes on the best of both worlds: merging 360˚ capture with 3D reconstruction. The result is a virtual scene in which the user is free to move about and can experience significant improvements in image quality when standing at the location of a 360˚ rig. This virtual environment is created entirely with consumer-grade equipment. As the 360˚ images we capture lack depth information, it is imperative that we resolve this to ensure a seamless transition between capture systems. As a solution, we texture map the image from the 360˚ rig to the 3D reconstructed environment. The hope is that this process can empower content creators to dream beyond the scope of the standing-point 360˚ content currently being produced without the need for expensive equipment that stretch financial resources.

Such a system has practical application to a variety of fields. For example, a user could walk through the galleries of the Louvre in Paris, appreciating the detail of art from perspectives recommended by a curator while also experiencing the scale and grandeur of the former palace. The absence of a need for specialized equipment could enable institutions with limited financial and technical resources to open their doors to virtual visitors from around the world. Because we enable the content creator to choose the specific points where image quality is improved (i.e., where in the scene 360˚ content is captured), one can direct a user's experience, providing a sequential order in the virtual environment and thereby prompting the user with a sense of story. In the area of cinematic virtual reality, a big question is how producers can develop a new language of storytelling when, unlike traditional cinema, VR lacks a frame to direct a viewer's attention. The interplay between low and high-quality photorealism could be used as a technique for guiding narrative.

But storytelling in VR need not be limited to entertainment. Imagine our system being used in a court of law where an attorney leads a jury through the real scene of a crime in virtual reality – highlighting key locations in the scene as though reliving the crime. Or, perhaps the user is a medical student studying the steps of a complicated surgery. This student can analyze scenes from an actual surgery an infinite number of times, while being able to move around a virtual operating theatre from home. These are not new dreams, but rather the elusive goal of most telepresence researchers. This seemingly intuitive approach for content acquisition that we design, combining 360˚ capture with 3D reconstructed spaces, could trigger years of applied research and the development of practical solutions to everyday challenges.

This thesis aims to give the reader a thorough understanding of the technical foundation on which our design builds upon, guidelines for methods undertaken to implement the system, and a presentation and discussion of sample results.

2: Background

The ultimate goal of virtual reality research is to create fully immersive computer generated experiences. In his 1965 essay, "The Ultimate Display", computer scientist Ivan Sutherland inspires developers to think of the computer display not as something that simply draws dots and lines, but as a device that could be a looking glass into a wonderland [4]. In this generated world, "A chair displayed...would be good enough to sit in. Handcuffs...would be confining, and a bullet... would be fatal" [4]. It is debatable whether or not this level of immersion is actually desirable, and while we may not yet know how to make virtual handcuffs confining, decades of research has made great advancements in area of *visual* immersion. These developments, in regards to visual immersion, can be broken into four primary subcategories: display, tracking, image generation/rendering, and content acquisition [5]. This paper details a system that integrates all four categories to improve the current state of immersive telepresence (i.e., real-world) experiences. While the aspects of display, tracking, and image generation of this system utilize available consumer technologies, our primary technical developments are in the method of scene acquisition.

2.1: System for Display, Tracking & Image Generation

As of 2016, consumers have access to high quality and affordable display, tracking, and image generation technology. Most modern smartphones are equipped with all the requisite technologies. Smartphones intended for VR, like the Google Pixel XL and Samsung Galaxy S7, have screens with impressive resolutions of 2560x1440. Gyroscopes and accelerometers enable low-latency internal tracking of the device's movement, and continuing improvement in the power of mobile CPUs and GPUs gives these devices graphics capabilities superior to desktops from only a few years ago (capable of rendering several thousand polygons at 60Hz) [6]. Inserted into a stereoscopic viewer, such as the Google Cardboard, Daydream, or Samsung Gear VR, a smartphone becomes an immersive virtual reality headset that eliminates the need of a tethered base station and complicated content deployment procedures.

These smartphone systems, however, are largely limited in their immersive capabilities primarily due to two issues. The first is a severely limited field of view (FoV) – the Gear VR, perhaps the most advanced of mobile VR headsets, provides only a 96˚ horizontal FoV. In context, a human's total horizontal visual field (i.e., what is seen by either one or both eyes) is about 190˚ when eyes are stationary and up to 290˚ if they are allowed to move. The binocular visual field (i.e., what is seen by both eyes) is about 114˚

[7]. To immerse a user visually, a headset display should at least match, and ideally surpass, the user's natural FoV. This remains an issue with most consumer VR headsets, including the Oculus Rift and HTC Vive, which both yield field-of-views of 110˚.

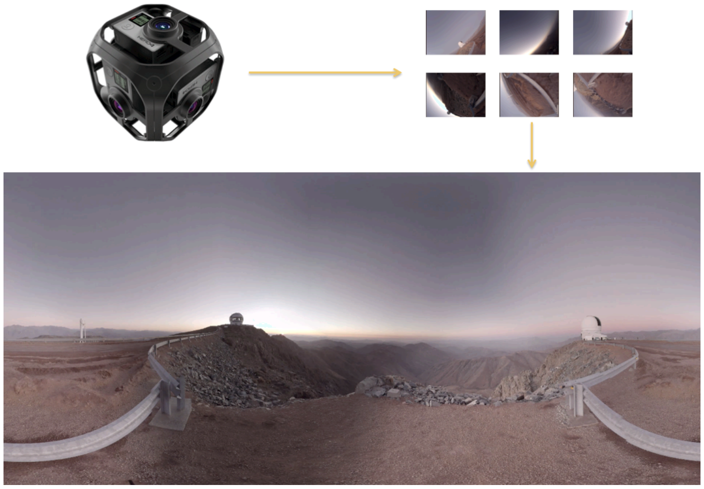
The second issue is the lack of an external tracking system that limits immersion to standing-point 360˚ content, inhibiting a user from physically moving around in the displayed environment. Internal tracking sensitivity of these systems is not currently sufficient to enable low-latency response to 6 DOF movement.

The Oculus Rift and HTC Vive provide better alternatives for consumers. Though both systems must currently be tethered to a base station PC, the respective headsets significantly surpass mobile VR systems in a few important aspects. These systems both have a display refresh rate of up to 90 Hz, relying on the processing power of the PC for image generation, and both feature low-latency external tracking that provides a user with 6 DOF mobility. Hence, the user has the ability to move in any direction within an immersed environment. A response to the slightest of head movements makes a notable difference in immersive quality. Since a primary goal of our system is to take advantage of mobility within a 3D reconstructed environment, we implement the system on an HTC Vive due to its capability for room-scale tracking.

2.2: Content/Scene Acquisition

With the increasing accessibility of VR headsets, capturing content for VR has become an important area of research. The acquisition of real-world environments for access in VR has two primary approaches: 360˚ (omnidirectional) video/image capture and photogrammetric 3D reconstruction. This thesis demonstrates how we can improve the quality of VR immersion by combining the two.

2.2.1: Background on 360˚ Cameras



Workflow (above): 6 GoPro images taken from spherical rig are stitched together to generate a 360˚ panorama

360˚ images, also known as photospheres, are created by stitching together a number of photographs, taken in different directions from the same position, to form a single spherical panorama. The images are stitched in the same way as all panoramas. As a brief overview: corresponding feature points are detected, the transformation between images is estimated based on the pinhole camera projection model, and images are blended at the seams. Popular 360˚ capture systems like the Ricoh Theta and Samsung Gear 360 take care of this process internally. Software like Kolor's AutoPano Pro allows non-technical users to stitch 360˚ panoramas from custom camera rigs and edit stitch-lines between images.

The ability to edit stitch-lines is an important tool because of inconsistencies that appear where images overlap. This problem arises because cameras forming a 360˚ rig cannot have the same optical center for all cameras. When simultaneously using multiple cameras, it is impossible for cameras to share the same optical center. As a result, objects (e.g., a person or a tree) that appear in the overlapping region of two cameras may appear displaced along stitch lines. Stitching software exists to allow editors to hide these inconsistencies. Unless a photosphere is generated with a single camera (with the use of a parabolic mirror) [8] or the images are reconstructed [2], a seamless representation of a scene is very difficult to achieve. Nevertheless, this is a problem that will only lessen with the advancement and eventual widespread use of stereo and light field 360˚ systems that accurately capture and project the geometry of a scene.

Monoscopic 360˚ capture rigs, like the GoPro Omni which is constructed from 6 GoPro Hero 4+ action cameras, yield high-resolution panoramas. With each camera capturing a 12-megapixel image, the stitched panorama yields photorealistic image quality. Rig designs for fisheye cameras, like that for Kodak SP360, excel in the one regard that only two cameras are needed to get a 360˚ panorama. But where these rigs fail in comparison to the GoPro rig, is in the extremely short focal length that is characteristic of fisheye cameras. As a result, only nearby objects appear lifelike in photospheres captured by fisheye rigs. Objects further away from the rig decrease significantly in size. Content shot on all monoscopic rigs, however, fail to give a user a sense of 3D depth and 6 DOF mobility. While users wearing virtual reality headsets may experience a sense of presence due to photo-real image quality in all directions, immersion is limited by the deficiency of visual depth and mobility.

2.2.2: Background on 3D Reconstruction

The 3D digitization of real-world objects and environments is a fundamental field of research in computer vision and graphics. Modern VR headsets allow us to view complex 3D models in virtual reality, but 3D reconstruction has long had relevance for professionals working in design, robotics, film & videogames, and cultural heritage.

Methods for acquiring 3D geometry can be classified as either active or passive scanning. Active scanning technologies include LIDAR laser scanners, time-of-flight scanners, such as the Microsoft Kinect 2.0, and structured light scanners, like the original Kinect. These active systems are ideal for highly accurate, real-time geometry acquisition, but (except for LIDAR) are usually restricted to indoor use and there is some concern that light emitted could harm objects of cultural value [9]. Using a Kinect for 3D reconstruction of indoor scenes is an attractive solution as each RGB image has an associated, precise depth map. As a result, geometry can be reconstructed from the depth maps alone. *KinectFusion* is a method for accurate real-time mapping of complex indoor scenes [10]. It enables the rapid reconstruction of scene geometry using a handheld commodity depth camera, by implementing a simultaneous localization and mapping (SLAM) algorithm to register camera pose with the 3D data. With *Realtime 3D Reconstruction at Scale using Voxel Hashing*, Neissner et al. develop on the *KinectFusion* method and give perhaps the most advanced system for real-time 3D reconstruction for large-scale environments [11].

Passive scanning systems are attractive because no special hardware equipment is required. In fact, any consumer-grade camera can be used to collect the dataset of images required to reconstruct geometry. The tradeoffs for an accessible capture system, however, are an elaborate software system needed to process unstructured data, and the requirement for well-textured surfaces for feature detection.

The three steps in the standard pipeline for passive geometry reconstruction are 1) Structure-from-Motion (SfM) for camera parameter estimation (including pose, focal length and radial distortion) and to construct a sparse point cloud; 2) Multi-View Stereo (MVS), to generate a dense point cloud by triangulating visual correspondences between images; and, 3) Surface Reconstruction to produce a surface mesh of the reconstructed geometry. *VisualSfM* [12] is a software solution for SfM. *PMVS* [13] provides a separate solution for MVS, and *Poisson Surface Reconstruction* [14] for mesh reconstruction. Fuhrmann et al. [9] offer one of the first complete open source pipelines that integrate all three steps into an end-to-end software system. We implement this system.

The development of structure-from-motion is one of the most important achievements in photogrammetry and computer vision. First presented in 1994 [22], SfM reconstructs the parameters of cameras from sparse correspondences between images in an unstructured image collection. Camera parameters consist of those both extrinsic and intrinsic parameters. Extrinsic parameters are the camera orientation and position, and intrinsic parameters include the focal length and radial distortion of the lens. To do this, first, features must be detected. Fuhrmann et al. implements both SIFT and SURF feature detection algorithms [9]. Both systems are among the top performing algorithms for feature detection and are scale and rotation invariant. Features are then matched. Because points that correspond between images are subject to the epipolar constraints of the pinhole camera model, false correspondences are removed. This process of matching features can take a long time because every image is matched with all other images (hence, a quadratic computational time).

With the known camera parameters, we can reconstruct the dense geometry. This is where we implement the multi-view stereo algorithm. Fuhrmann et al. use the *Multi-View Stereo for Community Photo Collections* approach developed by Goesele et al. [15]. Given known camera poses, MVS extracts the depth of every pixel in each registered image. As a result we get a depth map for every image registered during the SfM stage. The depth maps are then fused together to create a dense point cloud representation of the scene.

Fusing depth maps into a globally consistent representation can be a challenging problem. Fuhrmann et al. implement a Floating Scale Surface Reconstruction (FSSR) approach detailed in the work by Fuhrmann and Goesele[16]. Their method does not interpolate regions with insufficient geometric data. As a result, it ignores the reconstruction of regions without enough geometric data. This is useful, because many reconstruction algorithms will hallucinate incorrect geometry, requiring a user to manually clean up the model. Fuhrmann and Goesele's system finalizes the fusion of the surface mesh by implementing a variant of the *Marching Cubes* algorithm [17]. Implementing the Poisson Surface Reconstruction algorithm is an alternative method to recover a 3D mesh of the scene.

2.2.3 Brief Background on Texture Mapping

Applying realistic textures to reconstructed models is essential to generating photo-realistic models. Waechter et al. note that that many state-of-the-art 3D reconstruction pipelines use per-vertex colors [3]. As a result, image quality is limited to the resolution of the 3D mesh. For optimization reasons, usually the result is a low-resolution mesh that yields blurry results. The alternative, texture mapping, is important for creating realistic models without increasing geometric complexity.

Texture mapping from registered images is a two-step approach. First, the algorithm must select which image view (or group of blended views) to texture each face of the mesh. The second step is to optimize consistency between adjacent texture patches. In their paper, *Let There Be Color! Large-Scale Texturing of 3D Reconstructions,* Waechter et al. provide a pipeline for what can be considered the state-of-the-art for texturing 3D reconstructions [3]. Their system ensures photo-consistency between images used to create the texture map as well as color consistency. Since MVS systems naturally include image views that overlap, Waechter et al.'s solution maps only a single view to each face, thereby improving the resulting sharpness of the texture [3]. We use their texture mapping system.

2.3: Similar Work

A few projects have attempted similar work. *6-DOF VR Videos with a Single 360 Camera* [1] similarly addresses the lack of depth information and 6 DOF mobility in content captured by monoscopic 360° rigs. The authors present a novel warping algorithm that synthesizes new views based on the rotational and translation motion of the viewpoint. Their method elegantly attains 3D data of a scene by capturing images from a rotating 360° rig and applying standard SfM and dense reconstruction algorithms. Ultimately, however, their method does not enable full mobility around a scene, but only slight motions of a headset. Their system similarly also does not deal with dynamic scenes.

Though not intended for use in VR, Krispel et al. propose a method for automatic texture and orthophoto generation from registered panoramic views [18]. Their method involves generating a point cloud with a laser scanner, and capturing a high resolution panoramic image at the same location. Rectangular planar regions are identified from the surface of the 3D model generated, and an orthographic view is created per patch. Their method simply constructs orthographic views from the panorama and does not actually map the unique images used to construct the panorama. Thus, there will inherently be projection errors due to camera alignment.

Jump: Virtual Reality Video details the methods implemented to produce omnidirectional stereo Google's Jump 360˚ capture system [2]. Their camera rig features 16 GoPro action cameras and utilizes the overlap between images to reconstruct accurate depth estimation for every pixel. Their 3D reconstruction algorithm then interpolates geometry between camera views to create a consistent 3D 360˚ view in all directions. Their system, however, only supports head rotation (i.e., three degrees of freedom) with minimal 6 DOF for head movements. They similarly note that capturing 6 DOF VR videos is a critically important topic for future work. Not only is the camera system expensive, but the stitching pipeline is also just as intensive and must be run on several computers to yield timely results.

Matterport provides a commercial product with the intended application of virtual tours. Their rotating camera system captures both depth and RGB data with an active scanning system and is placed in different locations around a scene to reconstruct the environment in 3D. Their product has two modes. Either a user can walk through a relatively poorly colored 3D environment, or a user can appreciate high quality 360˚ images projected into 3D space at specific points of interest and teleport from point to point in a manner similar to that of Google Street View.

3: Method

The method for our proposed system can be divided into four parts. First, we collect our dataset by 1) capturing RGB images of the scene we want to reconstruct, and 2) capturing 360˚ images at points of interest in the same scene. With these images, we acquire and reconstruct the 3D geometry of the scene, using the three-step pipeline detailed in the Fuhrmann et al. paper, *MVE - A Multi-View Reconstruction Environment* [9].

The pipeline features an incremental Structure-from-Motion (SfM) algorithm to generate a sparse point cloud, a Multi-View Stereo (MVS) implementation to extract a dense point cloud from the unstructured image set, and, at the end, a surface reconstruction algorithm to construct a 3D mesh from the point cloud.

The third step in our method uses a texture-mapping algorithm by Waechter et al. for large-scale 3D reconstructions. We generate a texture-map exclusively using the views captured by the 360˚ rig. This step ensures that the quality of the images from the 360˚ camera rig translate accurately to the 3D environment.

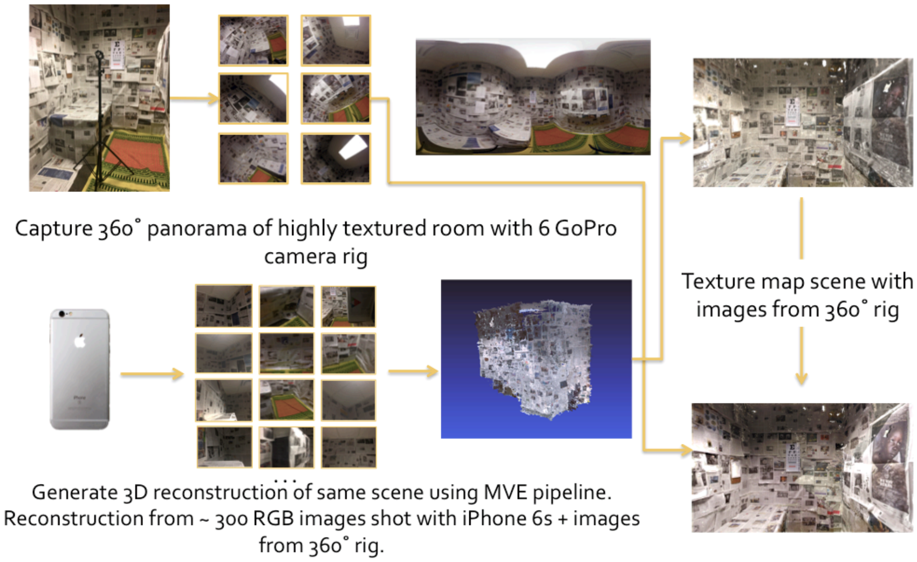
Finally, we import the 3D models and their respective texture maps into a 3D rendering engine (ex. a game engine like Unity 3D) for deployment to the HTC Vive. We add markers into the scene to indicate where in the scene the user will see an improvement in image quality – this would be due to the fact they would see the same continuous 360˚ panorama from the 360˚ capture system with the addition of 3D depth.

3.1: Notes on Data Acquisition

The 3D reconstruction method we implement requires a set of RGB images that capture the entirety of the environment we want to reconstruct. As Fuhrmann et al. suggest, “to successfully reconstruct a surface region, it must be seen from at least five views” [9]. The MVS algorithm we implement requires this benchmark in order to reliably triangulate all 3D positions when creating the dense depth map for each image. Fuhrmann et al. also note that unless the dataset becomes very large, more photos will not hurt quality. This specific MVS algorithm succeeds in preventing hallucinations of non-existent geometry. The paper also notes that for triangulation to work, the image set must reflect a parallax effect between images. SfM relies on this parallax to accurately triangulate tracked features.

We capture the same scene from specific points of interest with our 360˚ camera rig. The camera rig we use is constructed from six GoPro Hero4+ action cameras. We make sure to independently calibrate the GoPro cameras and undistort the images as the severe distortion makes it difficult to match features automatically as the SfM system attempts to estimate camera pose.

4: Results


Capture 360˚ panorama of highly textured room with 6 GoPro camera rig
Texture map scene with images from 360˚ rig
Generate 3D reconstruction of same scene using MVE pipeline. Reconstruction from ~ 300 RGB images shot with iPhone 6s + images from 360˚ rig.

Workflow (above): Implementation combining 3D recon. & 360 capture systems

In the figures below, we show images from a 3D reconstruction of a highly textured room. The room is covered in newspaper to emphasize the improvement in image quality. The images show the significant improvement in image quality when the cameras when the scene is texture mapped with images from the GoPro cameras in the 360˚ rig. The images used to reconstruct the scene were taken with an iPhone 6S. At the time of writing this thesis, we have not implemented the final results as we could not get all six cameras from the GoPro images to register with the SfM system. Although GoPro images were calibrated and undistorted independently of the SfM system (using a checkerboard pattern and the Camera Calibrator app within MATLAB's Computer Vision System Toolbox [24]), the SfM system struggled to find a significant number of correspondences between several of the GoPro images and iPhone images. The algorithm implements a SIFT and SURF feature detection, thus feature matching should be robust to scale and rotation. This is an issue worth investigating further that may have a simple solution.

3D Scene with Vertex Colors **Texture-Mapped Results**

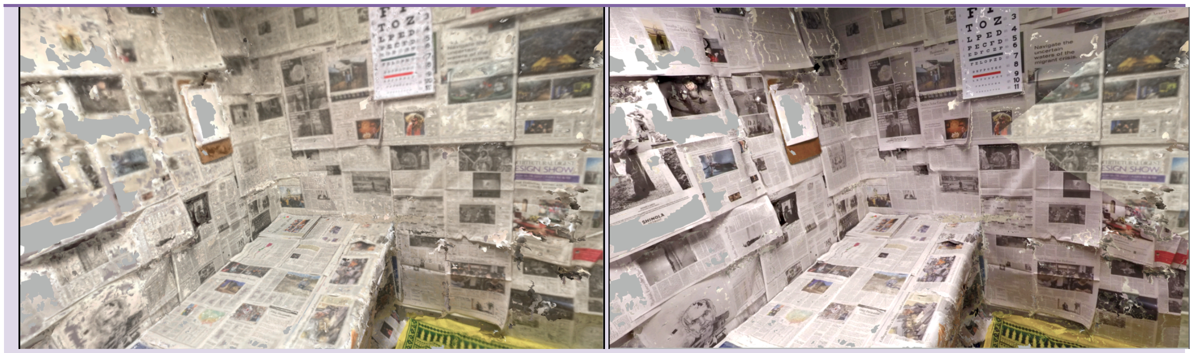


Figure 1. Perspective View

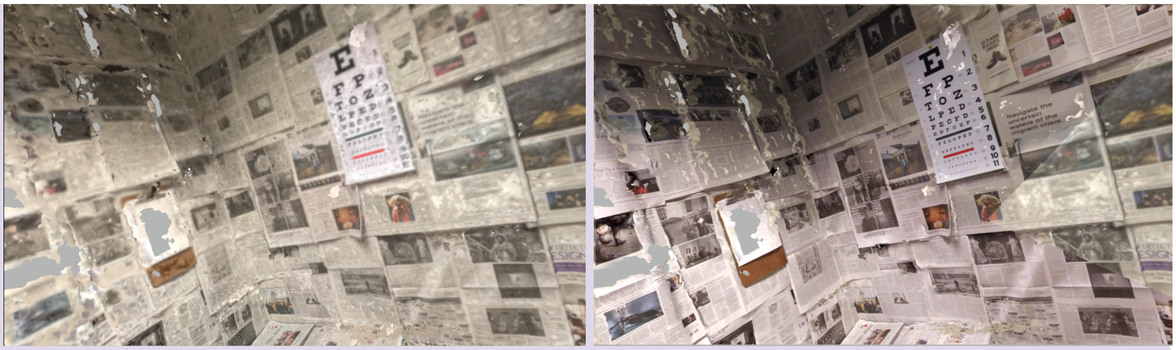


Figure 2. Close-Up View

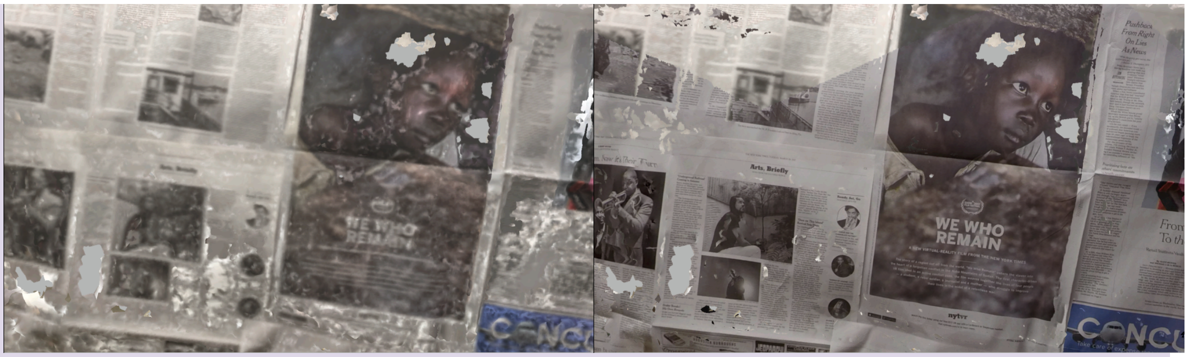


Figure 3. Extreme Close-Up View

As seen in Figure 1, the contrast between 3D scenes is immediately apparent. There is a noticeable increase in holes in the geometry of the texture-mapped scene. We get this result because only geometry from the selected views that are texture-mapped are displayed. The texture-mapped results shown above are overlaid on the more complete geometry from the vertex-colored scene. Technically, if a user stands exactly in the location of the 360˚ rig, and we were to successfully recover the complete geometry of the scene, the user would see no holes, but instead see exactly what they would see in a 360˚ panorama of the scene with the addition of depth information. This kind of precision is very difficult to achieve as it would require much more precise tracking of the user's headset (let alone eyes) than we have available via the HTC Vive.

Figure 2 shows how image sharpness is improved significantly. The contents of photographs shown in the newspapers were originally unidentifiable but are now clear. It is notable that a user can in fact read the bottom row of the eye chart shown in the scene.

In Figure 3 we show how, in the textured results, the legibility of text is dependent on how close the visual details are to the views being used to texture the results. Perhaps this is an indication that we would get a better results if, instead of using exclusively the images from a stationary camera rig, we generate the texture from all camera views used to reconstruct the geometry. That being said, if cameras are not perfectly registered, the likelihood of noticeable seams between texture patches increases with more cameras. The 360˚ rig would ideally also ensure more consistent results when texturing scenes – a user standing in the location of the rig would see the same content they would see in a regular photosphere, only mapped to 3D geometry.

It should be noted that when viewed in the HTC Vive, the image quality is not as sharp as when viewed on a PC. Text, for instance, is not as clearly legible when viewed in the headset. We believe that this proves that our results surpass the limitations of current headset display technology.

In designing the UX of our final virtual world, we implement a few design considerations. In a completed system, we would overlay the two environments so that holes in the 3D geometry exclusive to the textured results are less visible. We toggle the visibility of the textured results depending on where the user is standing. If the user's headset collides with the marker floating in 3D space designating the location of a 360˚ rig, the user will see the image quality improve. In this sense, the marker (perhaps designed as an orb textured with the 360˚ image) acts as a view-port revealing once obscure visual details.

5. Conclusions

By combining 360˚ capture with 3D reconstructed environments, we have designed a system that can be applied in a multitude of ways to improve immersion in telepresence VR content. The stark contrast between a 3D environment texture-mapped using Waechter et al.'s algorithm with images from the 360˚ rig and the same environment colored with vertex colors is indicative of the importance of texture mapping. Using exclusively images from the 360˚ rig to texture the scene ensures consistent results for a user when looking in all directions.

The designed system has the potential to work very well, but the implementation we detail is not yet perfect. In particular, our implementation of the *Multi-View Environment* pipeline does not yield ideal results for reconstructing indoor spaces. There are too many holes in the recovered geometry. This, however, is a temporary limitation. While using a passive reconstruction system enables the designer to reconstruct 3D geometry without the need for any special equipment, most indoor environments have large areas of un-textured surfaces. Even if walls are textured, it is likely that the floor and ceiling are not. An active scanning method, such as *KinectFusion*, or one of the projects that have developed from that work such as Niessner et al.'s "Realtime 3D Reconstruction at Scale using Voxel Hashing" [11], would yield better geometry as they are not reliant solely on RGB data.

Our initial goal was to map the texture from the 360˚ rigs to 3D geometry generated with *KinectFusion*. We successfully manually registered GoPro cameras (using all of the same math) with images used to reconstruct geometry from a Kinect. We were, however, unable to accurately texture map the GoPro images to the 3D geometry using Waechter et al.'s system. As Waechter et al. point out, relatively little work has been published on intuitive texture mapping pipelines [3]. To improve the realism of VR experiences that simulate the real world, it is imperative that further research is conducted to improve the ease of integrating state-of-the-art texture mapping techniques, given camera parameters. Waecheter et al. claim to provide a solution to this, but for this project, we were only able to successfully get their algorithm to work on datasets using the *MVE* reconstruction pipeline from the authors's colleagues at TU Darmstadt.

Finally, our system only deals with static scenes. While there are plenty of applications that would benefit from improvement in image quality for static scenes, an ideal telepresence environment would also feature dynamic and semi-dynamic objects. Dou and Fuchs [19] explore the addition of recorded dynamic and semi-dynamic objects and Chabra et al., [20] expand on this work to optimize placement of commodity depth cameras. *DynamicFusion* [21] builds upon work from *KinectFusion* to deal with reconstruction and tracking of non-rigid scenes in real-time. *Holoportation: Virtual 3D Teleportation in Real-Time,* a paper from Microsoft Research, details a real-time motion capture system for human characters in VR and augmented reality [23]. Texture mapping objects that move in real-time would be a significant challenge worth pursuing among those in the computer graphics community, with potential benefits for a wide array of users.

6: References

[1] Huang, Jingwei, et al. "6-DOF VR Videos with a Single 360-Camera". IEEE VR 2017 (2017).

[2] Anderson, Robert, et al. "Jump: Virtual Reality Video". SIGGRAPH Asia 2016 (2016).

[3] Waechter, Michael, et al. "Let There Be Color! Large-Scale Texturing of 3D Reconstructions". ECCV 2014 (2014).

[4] Sutherland, Ivan. "The Ultimate Display". ARPA Information Processing Techniques. (1965)

[5] Brooks Jr., Frederick P., "What's Real About Virtual Reality?" Special Report. University of North Carolina at Chapel Hill. (1999).

[6] Steed, Anthony, Julier, Simon. "Design and Implementation of an Immersive Virtual Reality System Based on a Smartphone Platform". IEE 3DUI 2013. (2013)

[7] Howard, Ian P., Rogers, Brian J., “Binocular Vision and Steropsis”. P. 32. Oxford University Press. (1996)

[8] Gluckman, Joshua, et al. “Real-Time Omnidirectional and Panoramic Stereo”. (1998)

[9] Fuhrmann, Simon, et al., “MVE – A Multi-View Reconstruction Environment”. Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage. (2014)

[10] Newcombe, Richard, et al., “KinectFusion: Real-Time Dense Surface Mapping and Tracking”. ISMAR 2011. (2011).

[11] Niessner, Matthias, et al. “Realtime 3D Reconstruction at Scale using Voxel Hashing”. ACM TOG 2013. (2013).

[12] Wu, Changchang. “Towards Linear-Time Incremental Structure From Motion”. 3DV 2013. (2013).

[13] Furukawa, Y. Poce, J. “Accurate, Dense, and Robust Multi-View Stereopsis”. PAMI 2012. (2012).

[14] Kaszhdan, M., Hoppe, H., “Screened Poisson Surface Reconstruction”. ACM TOG 2013. (2013)

[15] Goesele, M. “Multi-View Stereo for Community Photo Collections”. ICCV 2007. (2007)

[16] Fuhrmann S., Goesele, M. “Floating Scale Surface Reconstruction”. ACM SIGGRAPH 2014. (2014).

[17] Kazhdan, M. et al. “Unconstrained Isosurface Extraction on Arbitrary Octrees. SGP 2007. (2007).

[18] Krispel, U., et al. “Automatic Texture and Orthophoto Generation From Registered Panoramic Views”. ISPR 2015. (2015).

[19] Dou, Mingsong, Fuchs, Henry.” Temporally Enhanced 3D Capture of Room-sized Dynamic Scenes with Commodity Depth Cameras” IEEE VR 2014. (2014)

[20] Chabra, Rohan et al., “ Optimizing Placement of Commodity Depth Cameras for Known 3D Dynamic Scene Capture.” IEEE VR 2017. (2017)

[21] Newcombe, Richard, et al., “ DynamicFusion: Reconstruction and Tracking of Non-rigid Scenes in Real Time”. CVPR 2015. (2015)

[22] Armstrong, M. et al., “Euclidean Reconstruction from Uncalibrated Images”. BMVC 1994. (1994).

[23] Orts-Escolano, S. et al., “Holoportation: Virtual 3D Teleportation in Real-Time”. ACM UIST 2016. (2016).

[24] MATLAB and Computer Vision System Toolbox Release 2016b, The MathWorks, Inc., Natick, Massachusetts, United States.

7: Acknowledgments

From the School of Media and Journalism, I’d like to thank Steven King for the opportunity to create and experiment with 360˚ content alongside him during this past year, and for opening countless doors as well as letting me use his 360˚ gear for this project. The members of the Telepresence Research group (now known as Graphics & Virtual Reality) have been a constant source of inspiration, and it has been an honor to work alongside them. In particular, special thanks to Giovanni Avveduto, Haiwei Chen, and Rohan Chabra for helping me find my footing as I began my first research experience in Computer Science. I’ll always remember that Diane Pozefsky suggested I consider writing an honors thesis, and I greatly appreciate her for pushing me to work outside my comfort zone. Thank you to Jan-Michael Frahm for reading my thesis and for introducing me to the world of 3D computer vision during your class. And finally, to my adviser, Henry Fuchs, whose limitless energy and passion for the subject never ceases to amaze me, thank you for being an understanding and inspiring mentor during this past year. Thank you for your time and wisdom.