UNIVERSITY OF
BATH

Department of Computer Science

INDIVIDUAL COURSEWORK
Submission Cover Sheet

Please fill in both columns in BLOCK CAPITALS and post into the appropriate Coursework Submission Box.

for office use
Date and time received

This section will be retained by the Department Office as confirmation of hand-in

How to present your work

1. Bind all pages of your assignment (including this submission sheet) so that all pages can be read by the marker without having to loosen or undo the binding. Ensure that the binding you use is secure. Missing pages cannot be marked.

2. If you are required to submit part of the work on a disk, place the disk in a sealed envelope and bind the envelope into the submission.

You must keep a copy of your assignment and disk. The original is retained by the Department for scrutiny by External Examiners.

Declaration

I certify that I have read and understood the entry in the Department of Computer Science Student Handbook on Cheating and Plagiarism and that all material in this assignment is my own work, except where I have indicated with appropriate references. I agree that, in line with Regulation 15.3(e), if requested I will submit an electronic copy of this work for submission to a Plagiarism Detection Service for quality assurance purposes.

FAMILY NAME HAYNES	FAMILY NAME HAYNES
GIVEN NAME JACOB	GIVEN NAME JACOB
UNIT CODE **CM50170**	UNIT CODE **CM50170**
UNIT TITLE RESEARCH PROJECT	UNIT TITLE RESEARCH PROJECT
DEADLINE TIME & DATE 14/09/18 5:00PM	DEADLINE TIME & DATE 14/09/18 5:00PM
COURSEWORK PART (if applicable)	COURSEWORK PART (if applicable)
SIGNATURE *JHaynes*	SIGNATURE *JHaynes*

Light Probe Capture With 360 Cameras

Jacob Haynes

MSc Digital Entertainment

The University of Bath

September 2018

This dissertation may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.

Signed: *JHaynes*

Light Probe Capture With
360 Cameras

submitted by

Jacob Haynes

for the degree of MSc Digital Entertainment of the

University of Bath

September 2018

COPYRIGHT

DECLARATION

Abstract

Presented here is a proposed method for light probe capture with a 360 camera for use within image-based lighting. Image-based lighting (IBL) is a rendering technique used to capture an omnidirectional representation of a scene's lighting and is used as an alternative to simulating a scene's lighting. Light probe capture is a key technique for capturing a scene specific lighting and is an example of IBL. This is traditionally achieved by photographing a highly reflective chrome ball, however, the process is slow and tedious. With 360 cameras becoming commercially available and accessible we aim to remove the need for the chrome ball and capture the scene's lighting from one or more 360 photos.

Contents

List of Figures

Chapter 1

Introduction

Combining videos with computer generated props and actors is a challenging task in visual effects. One aspect of this process which is vital for the realism of the rendered object is the simulated lighting. Consistent local and distant lighting between real and rendered objects is required, if the visual appearance between objects does not match the realism is lost. As such, for realistic rendering with correct environmental lighting a detailed knowledge of the surroundings is required. This is typically achieved with image based lighting.

Image based lighting is typically achieved via use of a light probe [1] (see figure 1-1). A light probe is an omni-directional high-dynamic range (HDR) image that is used to record the lighting environment at a position in space. Mainly used for providing measurements of incident illumination and for providing realistic lighting for rendered graphics. The standard procedure for creating a light probe image is as follows:

1. Take photos of a mirrored ball.

2. Crop the images to the edges of the ball.

3. Identify corresponding points.

4. Merge the images into one unwrapped sphere.

The light probe captures the full dynamic range of lighting in the scene, from direct sources to areas in shadow, the probe can be thought of as a measure of radiance incident at the point.
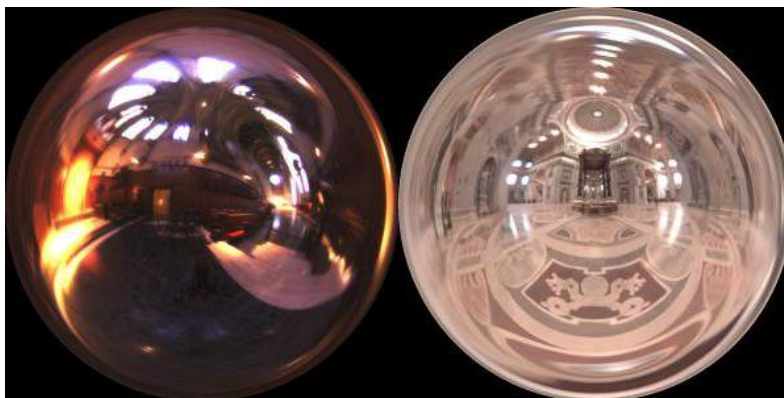
Figure 1-1: A HDR light probe capture of the Grace Cathedral, San Francisco (left) and St. Peter's Basilica, Rome (right). Assembled from two images of a reflective sphere with a dynamic range of 200000:1 [1].

Although specialist software and plugins exist to assist with this method, such as Photoshop, some steps are manual and slow such as taking the initial photos. In addition to this, the photographer is inherently in the images taken which adds additional complexity to the problem as the camera must either be an accepted reflection or removed from the images by using even more images of the ball. Typically, for a high fidelity HDR light probe, a minimum of four images, taken at 90 degrees apart, are needed to achieve a full light probe but can be achieved with two.

Traditionally, image based lighting has been limited to static lighting environments as HDR images are captured with multiple static shots often taken with exposure bracketing to capture a full dynamic range. More recently, it has become possible to capture HDR video (HDRv) [11]. This has allowed capture of dynamic environment lighting and the creation of moving light probes. HDRv cameras, however, are typically specialised cameras with high resolution and dynamic range and as such are not as widely available as a typical consumer HDR camera.

One key application of light probes is for the realistic rendering of synthetic objects into real-world scenes. First described by Debevec [1], Pessoa et. al. describe a method for achieving real-time photorealistic rendering of synthetic objects into real scenes [2]. This is a complex task requiring objects to be lit consistently with nearby surfaces as well as interact appropriately with the light and their surroundings. Objects should cast shadows, reflect, refract and emit light appropriately. In the pipeline described in [2], the team comprises

techniques regarding reflectance, illumination, and shadowing. They achieve this by developing an environment map using the light probe method described previously. The main goal of a light probe in this application is for the environment mapping used in producing realistic reflections. The basic pipeline of this reflection mapping is to use a panoramic or HDR image that is taken at the location of the object. The normals to the surface of the rendered object are then used to index into the HDR image which is used to then display the desired environment [12].
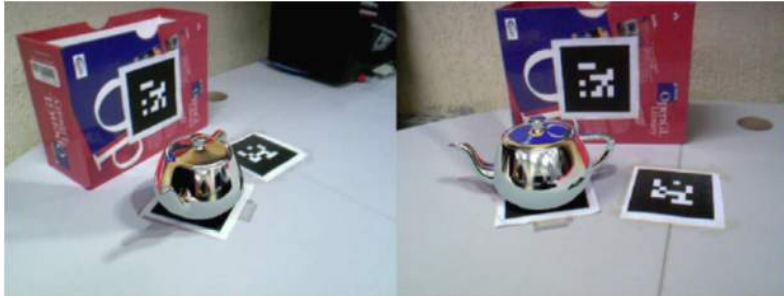
Figure 1-2: A synthetic teapot rendered into a real-world scene using the method described by Pessoa et. al. which accurately reflects the scene around it [2].

The traditional method to light probe capture using a chrome ball has an obvious issue in that the measured reflection map is set at the point of the chrome ball in space, as such real-world objects close to the ball may not be reflected properly in rendered objects that are positioned close to, but not in the place of, the ball. For example, if a real object exists between the chrome ball and the rendered object the real object may be 'reflected' on the wrong face of the rendered object. The simplest way to solve this is to take additional HDR images at the desired position of each rendered object. Taking HDR images, however, is a slow manual process, as such improving this process is the desired outcome.

I propose an alternative method to light probe capture using one or more 360 camera images. 360 cameras are becoming increasingly higher fidelity and more commercially available, as such, provide a simple solution to remove the reflective ball and the need for the photographer to be within the image. This document aims to outline the problem, proposed solution, a detailed literature review on the topic, a proposed method and corresponding algorithms, present

results, and finally draw conclusions.

1.1 The Problem & Objectives

As already discussed; the current methods of light probe capture are a slow and tedious process. The technology of 360 cameras provides a clear solution to streamlining this work flow and removing some of the unnecessary tedium of the image capture of a reflective ball. As such, the overarching objective of this project is fairly straightforward: Create an algorithm or approach for capturing a series of HDR light probes via a 360 camera. The objectives for the main goal of the project can be summarised as follows:

1. Calibrate the camera to obtain intrinsic parameters.

2. HDR 360 image capture of various scenes.

3. Create environment maps from 360 images.

    (a) Alignment of images.

    (b) Match brightness and colours between images.

    (c) Image stitching.

    (d) Blending of images.

    (e) Formatting the output.

4. Comparison between traditional light probes and 360 camera light probes.

The practical work of this project will focus on the above objectives. However, extensions to this project can be investigated should time allow. One potential extension is to investigate mapping a larger region of space using a collection of 360 image captured light probes. If, as is anticipated, the capture of light probes with a 360 camera is a fast process, capturing a collection of images around a room should be fast and simple. It should then be straightforward to map these images to each other spatially and may allow for the fast creation of a three-dimensional lighting environment map. This potentially could be accomplished by feature matching between images and generating a three-dimensional point cloud from the HDR images. As such an object could be rendered anywhere in the scene and reflect the environment. In the more in-depth literature review that follows in section 2, various extensions such as

this will be discussed in more detail as well as contemporary work in this area will be reviewed.

The literature review will be an in-depth investigation into current knowledge of HDR images, light probes, light probe creation, 360 cameras, 360 camera capture techniques, and synthesising objects into real scenes. 360 camera technology will also be discussed identifying the potential pros and cons of different models and types. As a secondary objective, the literature review will also include an investigation into environment maps, HDR videos, and moving synthesised objects in real scenes.

The major deliverable of this project is to create a software package that creates a light probe from input 360 camera images. The complexity and operation of this deliverable will be dependent on the previous work and time allowances but will be a standalone system that requires low system resources and computes the output quickly into a standard format for use in industry standard software and pipelines. This will most likely be developed in MATLAB as a proof of functionality for any algorithm created as it functions effectively as a fast prototyping tool for image processing.

The remainder of this report will detail current methods, work done, deliverables produced, a critical evaluation of the work done, and a discussion on future work and possible extensions to the work. In addition to this, it will also include details on the testing of the method developed as well as a comparison with traditional methods. This report will function as a standalone document for complete replication of the work done as well as a reference document for future work carried out on the topic.

Chapter 2

Related Work

This section will explore the literature, technology and other previous work in the areas of HDR imaging and light probe capture and application. First digital and HDR images will be explored, then light probes will be discussed in more detail. This will lead to a discussion of 360 cameras, their functionality, and the technology behind them. Finally, the uses of light probes will be investigated, including environment maps, synthesising objects in images, and applying these ideas to video.

2.1   Digital & HDR Imaging

Currently, in modern digital imaging, there is a drive to produce higher-resolution images. In consumer-level digital cameras found on mobile phones, it is common for the camera to boast anywhere from 3- to over 16-megapixel sensors with 12 being almost ubiquitous in the market [13]. Alongside this drive for higher resolution images, there has also been a major shift in how digital images are thought about in relation to the large amounts of data they can capture in each pixel.

Most digital colour images are represented in terms of three colour channels; red, green, and blue. Each of these channels can represent 256 values meaning over 1.6 million colours can be assigned to each pixel. As can be seen in figure 2-1, having only the 3 colour channels does little in capturing information on scene lighting when compared to the adjacent HDR image. In this figure, a conventional image and a high-dynamic range image (HDR) are shown. It is clear that the conventional image struggles to accurately

show bright and dark areas in the same image, leading to areas of over-, and underexposure.

Figure 2-1: This shows a comparison of the same scene taken with the same 12.2-megapixel phone camera. The left-hand image is a conventional digital image, the right is an HDR image. It can clearly be seen in the conventional image the window is overexposed and dark areas such as behind the chair are too bright for an accurate representation of the scene's lighting.

Despite the fact the HDR image is displayed on paper in this review, rather than on a digital screen capable of displaying the full range of data captured by an HDR, there are still clear advantages to using HDR capture over conventional image capture. HDR capable displays are still rare and costly.

As shown above, real-world lighting conditions play a large hand in image capture. This is largely due to the face that real-world light sources can vary in brightness by many orders of magnitude and conventional digital imaging can only account for two orders of magnitude with only 8 bits available per colour channel [14]. It is typical for real-world lighting to range from starlight ($10^{-3}cd/m^2$) through typical indoor lighting ($10^2cd/m^2$) to direct sunlight ($10^5cd/m^2$) which is over 100 million times brighter than starlight.

Current displays are far from capable of reproducing this range of luminance with images typically formed with a byte per colour channel per pixel as previously discussed. Encoding to this form typically occurs when the image is captured, however, this is far from ideal as large amounts of scene information

is lost during the capture. It is, therefore, preferable to capture as much of the scene with as large a range of intensities and quantization levels as possible. This information can then be stored until it is needed to be displayed on a device that cannot display the same level of data. These images that store a range of light intensities are HDR images.

Display technology is quickly catching up with HDR images. HDR displays are available and under constant development with OLED displays developed that have the capability of accurately displaying HDR content [15]. Displaying HDR images directly, however, is not the focus of this review; instead, we are more interested in the lighting data inherently stored by HDR images for the creation of light probes and image-based lighting (IBL).

HDR images can be created in via three distinct methods:

1. By employing rendering algorithms and other graphics techniques, this is not a capture but a 3D render of a scene.

2. Using multiple conventional images of the same scene and computational photography to capture the full range of HDR data.

3. Using a purpose-built HDR camera that is capable of directly capturing HDR data.

Computer graphics rendered 3D scenes for HDR generation is possible on many modern graphics cards and systems. These systems are able to directly generate HDR images and aspects of these systems will be discussed in section 2.2 but the larger topic of rendering is not of direct concern in this review and is covered thoroughly in other documentation [16–18].

Capture via conventional imaging is usually achieved by capturing a static scene with multiple images of variable exposure time for each image. This leads to a collection of images that are combined into an HDR image. Each image in the sequence will have different pixels properly exposed with others over- and underexposed. Typically in the combining of these images, each exposure is brought into the same domain by dividing the pixels by the exposure time assuming linear colour space, if not the inverse camera response curve must also be applied. This has the effect of recovering irradiance values from the recorded radiance values. Once in the same unit of measure, the corresponding pixels can be averaged across exposures resulting in an HDR image.
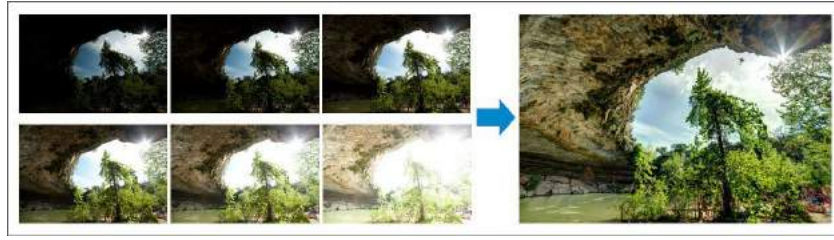
Figure 2-2: An example of 6 LDR images of different exposures which are combined to form a single HDR image [3].

In practice, this is not as straight-forward as cameras do not have perfectly linear light measurement and objects typically move between the individual exposure captures. Typically, techniques are applied to take into account the cameras response, artefacts such as ghosting and lens flare, and image alignment. Camera response curves are typically dealt with by weighting the function by the derivative of the response curve [19]. Image alignment is typically carried out via a technique from Kang et al. [20]. In this method each pixel has a motion vector computed between frames, neighbouring frames are warped and registered with each other, and finally the frames are combined into the HDR image. An alternative alignment method employs a mean threshold bitmap [21]. This method does not rely on previously knowing the camera response function and computes via bitmaps rather than 8-bit grayscale images. It forms bitmaps that identify pixels that are close to the median value of each exposure and then aligns each frame with these images. Kang et al. also propose methods of ghost removal [20] and other artefact removals.

Packages exist for creating HDR images such as HDRShop [22] which creates HDR images from a collection of standard low dynamic range images. There also now exists sensors which can accurately capture a full dynamic range of a scene in a single capture. These sensors are available in many commercial cameras such as OmniVision's HDR cameras [23]. Also, many modern cameras offer the option of directly capturing HDR images or computing an HDR image directly in a perceived single capture. One major disadvantage of directly capturing HDR data is that the raw image data that is stored takes up a large amount of storage space, less of a problem in studio scenarios but a serious issue in mobile platforms although this has been rendered less of an issue with recent advances in large solid state storage.

2.2 Image-Based Lighting

A major application of HDR images, and one we will explore thoroughly in this project is that of image-based lighting (IBL) or environment mapping [24]. A key challenge in computer graphics is simulating light and realistically lighting a scene. Traditionally this is achieved by simulating how light bounces through a scene. This is typically done by using geometric primitives to model the scene and material properties to replicate reflectance and other light-surface interactions [25, 26]. This simulated information is then used to render the scene and is a typical process for animated films.

A more recent technique, and the one of focus in this review is that of IBL. As HDR images capture the full range of light arriving at the lens, they inherently contain detailed information of the scenes light sources. This information including colour, size, shape, intensity, and location. Using rendering algorithms, it is possible to use HDR images to simulate how objects would be illuminated by light from real-world scenes. This is known as IBL and generally is carried out by using HDR images and examples of which can be seen in figure 2-3.

Figure 2-3: Examples of various synthetic objects lit by IBL techniques [4]. Shadows, reflections, and shading are all present and realistic. The background shows the environment that was captured in the HDR.

In partnership with HDR images, IBL requires two other important tech-

nologies; omnidirectional imaging, and global illumination. Omnidirectional imaging is the process of capturing images in all directions from a single point in space, traditionally this is done via light probes discussed in section 2.2.1 but, as the title of this project implies, can also be achieved with 360 cameras (section 2.3).

Global illumination is a term that captures rendering algorithms that simulate how light travels from source to target, both directly and indirectly via reflections [27]. Typically global illumination algorithms are built on ray tracing rendering methods, the first of which was proposed by Appel in 1968 [28], in combination with algorithms describing light-material interactions.

The RADIANCE system [29] is a well-known package for IBL via light probes but most modern animation software and game engines include algorithms and tools for implementing IBL. High-end software such as Maya performs IBL from HDR images and HDR environment maps with it supported by most of Maya's lighting algorithms and rendering systems such as Skydome and mental ray [30]. Blender also natively supports IBL via HDR images as part of the Cycles rendering system and is being implemented in the new Eevee physical based rendering system [31, 32]. Many game engines also support IBL; such as the Unreal engine which renders IBL via a method it refers to as Ambient Cubemaps [33], and Unity currently has an experimental implementation of IBL via a tool called Look Dev but is still very much in the development process [34].

The basic process of lighting a scene using IBL has been touched upon already in section 1 but the main pipeline for IBL is as follows:

1. Capture omnidirectional HDR images

2. Assemble the light probe

3. Model the reflectance and geometry of the scene

4. Render and illuminate the scene with the IBL environment

5. Post-processing as required

Each element of this process will now be discussed in greater detail. It is also important to note that techniques exist for IBL that do not involve HDR capture; Gardner et al. present a method that predicts the HDR lighting of a scene from an LDR image [35], and other real-time IBL solutions that do not directly use HDRs will be discussed in section 2.4.

2.2.1 Light Probes

As previously defined, a light probe image is an omnidirectional high dynamic range image that captures the incident light at a point in an environment. The traditional method for light probe capture is done by taking two or more HDR images of a chrome ball. A chrome ball is used as they reflect the majority of the environment they are in, not just the hemisphere that is photographed. This is due to the optical properties of mirrored spheres in that the law of reflection, derived from the Fresnel equations [36], states that the angle of reflection must be equal to the angle of incidence, as such even rays of incidence close to 180 degrees will be captured by the lens. This means only two HDR images need to be captured to obtain a good omnidirectional image, however, it is common to take three or more purely to assist in the removal of the camera from the image.
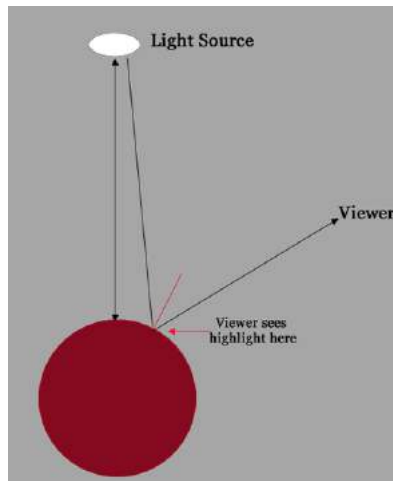
Figure 2-4: A demonstration of the law of reflection applied to a sphere. It can be seen that increasing the angle of incidence to close to 180 degrees will still allow reflected light to be captured by a wide angle lens. [5]

Ignoring the problem of generating the HDR image discussed earlier in section 2.1, and assuming the camera used outputted HDR images of the ball, the next step is to assemble the images into an omnidirectional map of the sphere's surface. This poses a matching, stitching, and warping problem to map these two images onto the sphere's surface. Many algorithms already exist to achieve this with a prominent one being Debevec and Malik's algorithm [37] or using a package such as HDRShop [22]. The completed light probe can then

be used in any of the software systems, or other capable rendering tools, as an environment map or as a texture map for an object at that point in space. Examples of light probes can be seen in figure 1-1.

The main drawbacks to using the mirrored sphere include the obvious blind spots that occur due to the reflection of the camera. Tools do exist to remove these blind spots by capturing multiple images of the sphere, although this is obviously a time-consuming process. There is also a desire for the sphere to be relatively far away from the camera to minimise the camera reflection and to keep the view orthographic. Keeping the sphere large in the image then becomes an issue as long focal length lenses are required which then typically leads to a shallow depth of field shots. There is also an issue that arises with the spheres themselves in that it is near impossible to get a perfectly reflecting sphere due to imperfections on the object. The material properties of these spheres also mean that only about 50% of the incident light is typically reflected so the captured luminance is often much darker than is representative of the scene. This requires some degree of calibration to correct and is often done by capturing the background without the sphere for measurements of the reflection and calibration of the final image from these measurements. Again adding to the time-consuming nature of the process. Other physical issues also arise with reflective spheres such as polarized reflections, derived again from the Fresnel equations [36].

Alternatives to capturing omnidirectional images that do not involve a mirrored sphere include using fish-eye lenses, panoramic cameras, and tiled photographs. These all have various pros and cons associated with them. For example, most captured tiled photographs and panoramas typical have limited verticality and this application requires a complete, uninterrupted 360 view. With the right panoramas and tiled photographs it is possible to get HDR omnidirectional images this way but again forming these images requires multiple captures of the scene in different directions. Fish-eye lenses also require two or more captures, with 180 degrees being a typical capture maximum for a single image.

2.2.2   Geometry, Reflectance, and Rendering

Once the light probe has been captured it then needs to be applied to the scene. For applications where an object is synthesised into a video scene this process

is very different as the background and other objects are already present. The bigger issues for such an application lie in realistic shadows, occlusions, and 3D object positioning in what is a 2D video. This will be discussed in more detail in section 2.4. For rendering entirely synthetic scenes with IBL the entire scene's geometry needs to be modelled and all objects require optical properties such as reflectance applied. As discussed previously, most modern engines and software support this process.

In IBL for synthetic scenes, the scene is effectively surrounded by the surface of the light probe. This surface is mapped to the inner surface of a finite sphere or cube which contains the modelled synthesised scene. This surface is defined as an 'emissive' surface in most tools so the image on the surface is treated as the source of light emission. Colour, location, and intensity information for the light rays (in a ray-based rendering method)is gathered from the HDR images and used to render the lighting for the scene. The large majority of published tools also take into account occlusions and reflections of the various objects synthesised in the scene and include them in the final render.

2.2.3 Post-Processing

Due to the vast amount of lighting information captured in HDRs, many interesting and effective post-process effects can be applied to rendered scenes using the information stored in the HDRs. Varying colour tones, exposures, brightness, white balancing, and effects such as blur are all easily and effectively applied to rendered scenes while still maintaining the photo-realistic lighting that is offered by IBL.

Tone mapping and colour grading are key in completing an IBL rendered scene. As previously discussed, most displays are incapable of accurately displaying HDR images. This means the rendered scene must be processed to map the desired tones and exposure factor to the final image. Chapters 6, 7, and 8 of High Dynamic Range Imaging [38–40] discuss in detail tone reproduction and other editing procedures of HDR images in detail so as to obtain the desired output. The functionality and uses of these post-processing tools are beyond the scope of this review but it is important to note that they are possible due to the quantity of information stored within the HDR images that allows the full range of optical effects to be explored without losing visual

realism.

2.3   360 Cameras

360 video and photography is becoming widely available to consumers. Technology which used to cost thousands now costs a fraction of the price and many consumer level 360 video and static image cameras exist of various quality. These cameras generally fit into one of two types. The first, and normally seen with the higher resolution cameras, is having a sphere of multiple cameras. The second is by having two cameras with fish-eye lenses rigged back to back. Ho and Budagavi describe a simple low-cost stitching method for two fish-eye cameras 360 capture [41]. The method proposed involves unwarping the images, measuring and correcting for distortion, and then blending and aligning the two images. Matzen et al. take this further and describe a low-cost method along the lines of the second technique [42] which goes further to include stereo disparity. Here they use two cameras rigged to capture video for VR applications but the principle remains the same - using fish-eye lenses to capture 360 videos. On the more expensive ends of the technology, the Panono system is designed to capture high-quality HDR and LDR images in 360 and follows the first technique of having multiple cameras in a spherical form [43].

There do exist low cost 360 cameras that market themselves as HDR capture cameras for use in creating environment maps. The Theta 360 camera is an example of this, which requires a stand-alone application for taking the necessary LDR images required for post-processing into a 360 HDR image [44]. This post-processing of combining these bracketed photos is also done via an external program such as Photoshop.

One reoccurring theme within the use of 360 cameras in HDR production for environment maps is that each step of the process is often carried out via different tools. The combining of the bracketed LDR photos into an HDR, any stitching that needs to be accomplished, and any warping to take into account the shape of the environment are all often carried out using different stand-alone tools. This renders the apparent ease of capture via a 360 camera almost pointless due to the vast amount of post-processing required to get the final image. Even high end 360 cameras that have built-in HDR capture, such as the Ricoh Theta V [6], require external processing into a format that resembles a

light probe output. A tool that transforms generic 360 HDR panoramic shots into the same spherically mapped format as a traditional light probe would be a valuable, time-saving asset. Such a tool could also include applications such as removal of blind spots or other artefacts that are commonly seen at the top and base of 360 captures that use only one or two images in the stitching.

Figure 2-5: An example 360 image taken using the Ricoh camera [6][7]. The camera and photographer can clearly be seen in the base of the image.

As is shown in figure 2-5, a single capture from a typical 360 camera includes unwanted aspects such as the camera itself and the photographer. This can be partially mitigated with the use of rigs and tripods but, for the average home user, stability remains a problem. The potential tool discussed prior could be developed to take in a series of HDR photos taken by the user as they walk around a localised area and stitch the images appropriately so as to remove the photographer and the camera. Methods such as Photoshops 'Image Stack Mode' could be applied to this tool to achieve this [45]. Other methods also exist that could achieve this, such as exemplar-based and depth-based in-painting algorithms [46, 47].

It is anticipated that the processing required in the software that will be developed will follow many of the unwarping and stitching steps identified by Ho and Budagavi [41]. The main difference is that the images captured for this use will be HDR format rather than conventional LDR images. Ho and Budagavi unwarp the images to return a 2D spherical projected picture for the two captures, align the images via a matched points within the overlapping region, and stitch the images with an improved alignment found via a fast normalized cross-correlation algorithm [48]. It is likely that the methods used in Ho and Budagavis work will prove highly useful in this projects work and

that any unwarping, alignment and stitching carried out will build on their work by extending their techniques to cover the multiple bracketed images within an HDR image.

2.4   Real-Time Applications of IBL

Light probes and other forms of HDR environment maps have potentially many uses in computer graphics, particularly in rendering synthetic objects to scenes. We have already discussed the uses of light probes in applying realistic, scene specific, lighting to synthesised objects, it is also possible to use HDR environment maps to assist in editing objects that exist within a scene. Typically in digital imaging editing such as blurring, warping, and re-colouring is simple. Adjusting materials of an object, however, is a manual and tedious task. Khan et al. propose a method that simplifies and automates much of this process via the use of HDR images [49]. This method uses a single HDR image and exploits tolerances of human vision to hide physical inaccuracies to deal with occlusions, transition, and reflection changes. Potentially this work could be extended on to improve on these physical inaccuracies through the use of 360 scene capture.

Mixed reality and augmented reality systems are clearly a key technology that has many applications of IBL. Synthesising objects into real-world scenes and 360 videos is an area of constant development with many methods employing IBL. Typically rendering these objects with the complex shadow and lighting algorithms used with IBL is an offline process and not available in many MR/AR applications. However, some methods that approximate the output to an appropriate level do exist. These include Debevec's median cut algorithm, Ng et al. algorithm, and Ramamoorthi et al. spherical harmonic basis approximation [50–52]. Rhee et al. take this a step further and propose a method for real-time IBL using 360 videos for use in MR applications [53]. In their method, they use LDR 360 video streams as input and estimate an IBL output, their method achieved this real-time interaction with 360 videos containing 3D virtual objects. Despite this there are obvious limitations with their approach in comparison to HDR setups; high-intensity lighting is clipped in the LDR video capture leading to a much lower quality output than from an HDR capture, although this is to be expected due to the emphasis on real time application. Iorns and Rhee also propose a method for real-time IBL,

however, theirs is based on panoramic video and still uses LDR capture as with the previous method [54].

Hajisharif et al. propose an alternative method to real-time IBL in which they shade virtual objects using a sequence of HDR light probes. Instead of using typical HDR images they use HDR video to capture the sequence [55]. This allows IBL to be as dynamic as real-world lighting environments without the hassle and manual work of capturing a series of traditional light probes.

Figure 2-6: Examples of virtual models rendered with HDR environment maps in real time for use in AR applications [8].

A great deal of work has been carried out on the photorealistic rendering of objects within augmented reality using IBL. Agusanto et al. propose an approach based on HDR images for use in IBL and environment illumination maps, examples of which can be seen in figure 2-6 [8]. The work of Kronander et al. nicely collates various IBL based works on photorealistic rendering for mixed reality applications [56]. Kronander et al. point nicely towards the early work of Devebec [1] through to HDR video work and efforts to consider temporal and spatial scene illumination variation [57].

Chapter 3

Methodology & Algorithms

The goals and motivations of this project have been outlined in chapter 1. In this chapter the methodologies used and the algorithms developed will be discussed in detail. Firstly, the specific equipment used will be discussed and justified. This will lead to an overview of the procedure required to go from an example output image to the final result. Each step of this procedure will be described in detail within relevant sections with accompanying pseudo code where necessary. The thought process throughout the project will be indicated as well as problems and solutions identified and discussed.

The full procedure can be broken down into the following sections: firstly, the formatting of the output RAW image into a format appropriate for processing. In this case, a TIFF file format will be used for easy manipulation within MATLAB. Secondly, calibrating the camera so as to obtain the correct intrinsic camera properties. These intrinsic properties will be used within the unwarping process to transform the image into an equirectangular format. Thirdly, correcting the colour of the two images to ensure that brightness and colours are uniforms across the image. This will be done by computing an affine matrix to transform the pixel values. Penultimately, the alignment, stitching, and blending of the images performed using matched points. And finally, the removal of unwanted objects will be discussed via median blending of multiple aligned 360 images.

3.1 Equipment

The equipment used in this project is fairly limited, this is in line with the goal of this project in that a low cost, simple consumer solution for the creation of light probes is the objective. The full list of equipment is as follows:

1. Insta360 one camera [9]

2. Micro SD card and USB reader

3. Computer. All code is non-parallel and not GPU enabled.

Software used is as follows in its entirety:

1. MATLAB R2017b (Code was made and used on MATLAB but could easily be exported in C)

2. OCamCalib: Omnidirectional Camera Calibration Toolbox [10]

3. dcraw [58]

There is no reason that this method could not be applied to any other 360 cameras with some minor adaptations to the initial image formatting or to set-ups with more than two images required for the 360 coverage. The Insta360 one camera used outputs images in Adobe DNG RAW format, as such dcraw was used to convert the output files into TIFF format, any other conversion tool could also be used as long as it maintains the HDR format of the image.

The tool dcraw was chosen as it is small, open source, and lightweight using only standard C libraries. The tool also provides high levels of control over the file conversion, in this case, the output file was chosen to be written in linear 16-bit TIFF format with the white balance obtained from the average of the whole image.

The camera calibration was performed using OCamCalib and the full algorithm developed does rely on some of the functions within this toolbox for undistorting the initial images. MATLAB was used to write and run the full algorithm but the code could be exported in C or rewritten using OpenCV without changing the algorithm.

3.2   Initial Image Extraction

The Insta360 one camera can be used to capture RAW images in a DNG format. The camera has two lenses as shown in figure 3-1, the output images from the two lenses are concatenated and presented as a single DNG image as shown in figure 3-2.

A key feature of the camera that is worth noting, is that the lenses are offset with one above the other. This may cause alignment issues later due to this slight transformation, but this will be discussed alongside the results.

The output images are converted to TIFF format via dcraw and then split in half to two identically sized images. These images are labelled as upper and lower images for the remainder of this paper. It is also worth noting that HDR images can typically appear very dark; this is due to the full dynamic range of the image being displayed and typical displays cannot show this well. As such most images shown in this paper are LDR versions of HDR images purely to allow the reader to see the image properly. Each image will be identified in the caption as HDR or LDR.
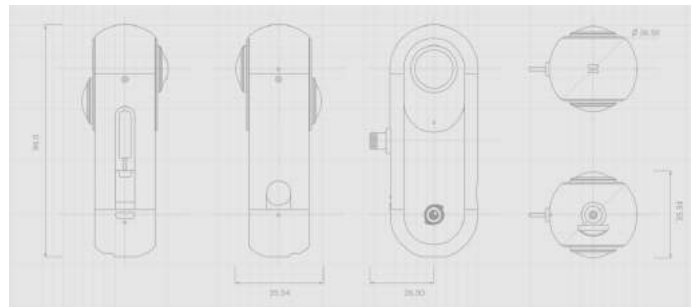
Figure 3-1: A schematic diagram of the Insta360 one camera used. As can be seen, the camera has two lenses and each 360 image is produced from a single capture from each lens [9].

Figure 3-2: An example output image from the Insta360 one camera. The image has been converted to a PNG for viewing here. As such this example is not truly HDR, the true output image is in a DNG file format which is then converted to a TIFF file via dcraw. It can be seen each lens has over 180 degrees field of view by looking at the corresponding calibration shapes A through E, this was confirmed by the manufacturer to be 270 degrees.

3.3 Camera Correction

Camera calibration was performed using the OCamCalib toolbox [10] using the calibration pattern provided within the toolbox and can be seen in figure 3-3. The calibration procedure is outlined within the toolbox but can be summarised as follows. The toolbox is designed to calibrate one lens at a time and as such, each lens was calibrated separately and labelled as upper and lower calibrations. Images were captured of the calibration pattern at various angles and positions; 16 were used for the lower lens and 15 for the upper. All

calibration images can be seen in appendix A.
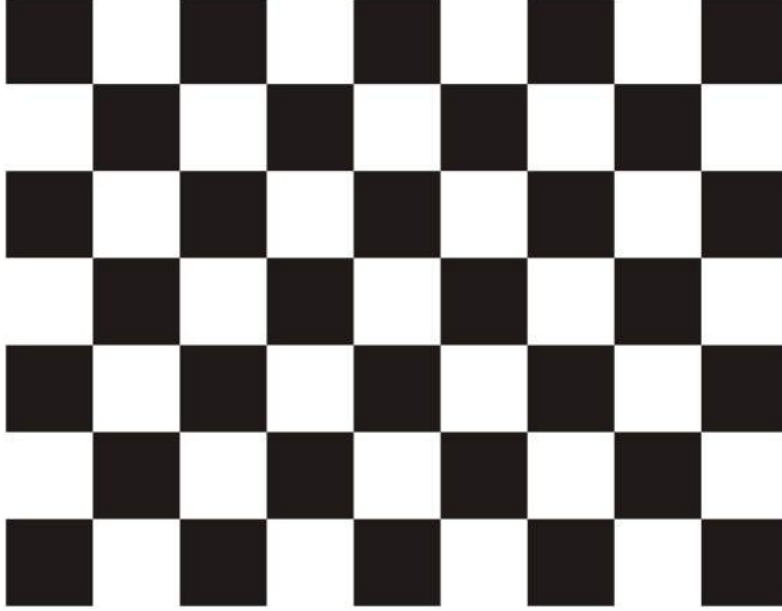
Figure 3-3: The calibration pattern provided within the OCamCalib toolbox that was used within the calibration procedure [10].

The calibration process involves converting the images into grayscale JPG images. The corners of each square of the calibration pattern are then labelled along with the size of each square recorded. The calibration is then performed via a procedure identified in [59] and explained briefly bellow. The calibration is then refined by the toolbox and the extrinsic position of each chequerboard is checked for consistency.

The method used by the OCamCalib toolbox is based on an approximation that fisheye lenses can be modelled as a central system. A central camera is a system in which every optical ray intersects to a unique point and is described in more detail within [60]. The calibration of an omnidirectional camera, as in this case, focuses on finding the relation between a 2D pixel point and a 3D vector on a unit sphere, centred on this central system intersect point, as can be seen in figure 3-4.
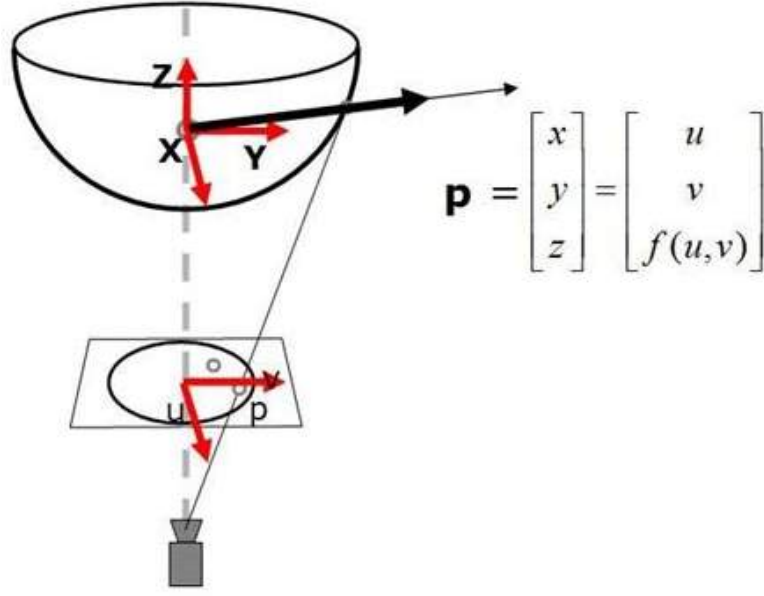
Figure 3-4: The relation between the 2D pixel points and the 3D unit sphere vector coordinates for the omnidirectional camera model [10].

The model can be described as follows, firstly let p be a pixel of the image with pixel coordinates (u, v) with respect to the centre. P is then the corresponding 3D vector from the single effective viewpoint with coordinates (x, y, z). As it is assumed that the camera and the central point are perfectly aligned, (u, v) and (x, y) are proportional. What is needed is the function that maps p to P, the proportionality can be included in the function, f, as

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u \\ v \\ f(u, v) \end{bmatrix} \tag{3.1}$$

Due to the rotational symmetry of the problem, f only depends on the distance of a point to the image centre ($\rho = \sqrt{u^2 + v^2}$). The function, $f(\rho)$, is described as a polynomial whose coefficients are estimated by the toolbox.

The full results can be found in table 3.1. It can be seen that the affine parameters are very small so the full camera matrix is close to the identity matrix. This was expected and supports the accuracy of the calibration. The main issue found during the calibration process was that the corner detector within the OCamCalib toolbox regularly failed to identify the corners of the

calibration pattern. This meant that many of the images had all the corners labelled manually, which on a high-resolution image was not the most accurate process. The reprojection error ended up being fairly large, an average of 13.48 pixels across each chequerboard image, although despite this good results were acquired so the calibration was accepted.

	Polynomial coefficients		Affine parameters			Center	
	Direct	Inverse	c	d	e	Row	Column
Upper	-8.71e+02 0.00 3.50e-04 -1.41e-07 1.22e-10	1464.98 885.65 -82.65 119.59 99.18 -53.77 58.14 43.97 -84.21 7.23 77.78 -6.42 -38.23 -2.01 9.54 2.58	0.99	0.00	0.00	1725.75	1725.07
Lower	-9.34e+02 0.00e+00 2.55e-04 -2.45e-08 9.46e-11	1481.14 812.60 -58.07 152.72 50.27 -4.35 50.56 -24.27 -10.89 52.82 3.44 -29.79 -2.53 8.81 2.57	0.99	0.00	-0.00	1728.49	1725.90

Table 3.1: Table of calibration parameters computed for this specific camera.

3.3.1   Unwarping Images

The camera calibration results are then used to format the image into a 2D equirectangular projection using the undistort function within the OCamCalib toolbox. Firstly the equirectangular (resultant) image pixel indices are converted to polar coordinates, theta and phi, centred at the middle pixel of the image. These polar coordinates are then converted to 3D Cartesian vectors using the following equation:

$$
\begin{aligned}
x &= -\sin(\phi) \\
y &= \cos(\phi)\sin(\theta) \\
z &= -\cos(\phi)\cos(\theta)
\end{aligned}
\tag{3.2}
$$

The 3D vector is then used with the world2cam function within the OCamCalib toolbox to return the pixel coordinates of the corresponding pixel in the original fisheye image. The world2cam function computes this using the calibration polynomials. The returned coordinates are then used to give pixel values to the equirectangular coordinates. The unwarped images are as follows:

Figure 3-5: An example image unwarped into equirectangular projections. The image is brightened for ease of viewing.

3.4 Image Alignment

3.4.1 Feature Matching

Once the images are unwarped into an equirectangular form, the next step is to identify and match features between the images for use in aligning the two image components. Each image covers a field of view of 270 degrees so the edges of each image overlap and contain matched points. As such the two parts can be aligned by projecting the images onto a unit sphere and rotate one around the sphere's surface to align the points with those in the second image. This rotation matrix will be set result for the camera and can be treated as a calibration step. In the implementation described here, two feature matching procedures will be described. One fully automated which will be described here and one with manual feature selection which will be described in section 3.4.2. The user indicates which procedure to use at the start of the program.

The automated feature matching is performed on grayscale versions of the images using SURF feature detection. A problem that was overcome at this stage was that the HDR images are very dark, due to the high range of brightness captured, and features are hard to extract. This can be seen in figure 3-6. To overcome this, the images are brightened and enhanced using histogram equalization. Although greatly diminishing the visual quality of the image with colours distorted and visual artefacts created, this is irrelevant as the extracted features are in the same location on the original images so the brightened image is only used for this step.

The feature extraction was drastically improved by this brightening and enhancement, the comparison between extracted features in both versions of the image can be seen in figure 3-7. Once features were extracted from the two images the features are matched to provide a collection of matched points between the images, shown in figure 3-8.

Figure 3-6: A Grayscale version of the equirectangular image can be seen in the top panel. It can be seen that the image is incredibly dark, due to the HDR pixel values being displayed with a small range of possible values (255 per channel). Feature extraction is highly unreliable due to this as the pixel value gradient across edges is small. The brightened image can be seen in the lower panel, edges and features are greatly enhanced for detection.
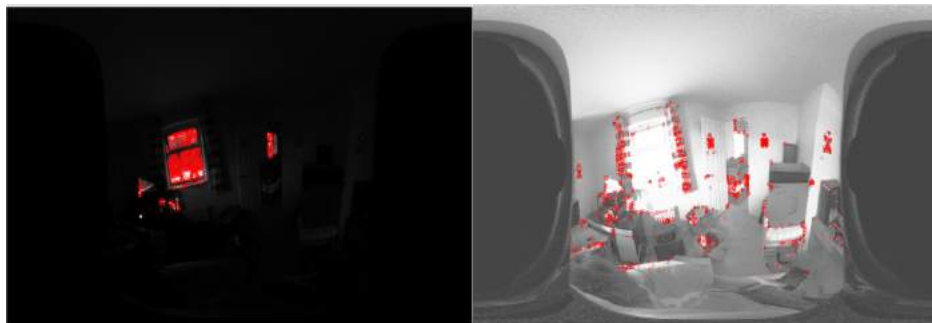
Figure 3-7: A comparison of extracted features between the dark and enhanced images. It can be seen that many more useful and distinct features were successfully extracted in the enhanced image whereas in the dark image they are overwhelmed by small bright spots.
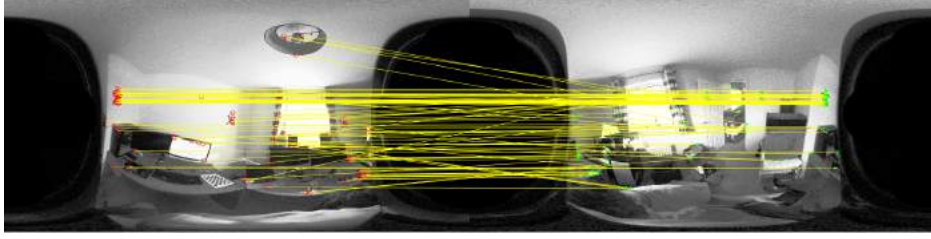
Figure 3-8: Features matched between the two equirectangular images. Clearly, some features are mismatched, although there are also clearly many well-matched features.

It is clear when looking at figure 3-8 that, although many of the matches are accurate, there are poor matches that exist. As such a method of ignoring these mismatches must be employed. The edges of each image overlap and can be aligned as described at the beginning of this section. As the rotation matrix required between these two images will be consistent between captures, the goal here is to get the best possible combination of matched points to compute the rotation matrix to use on all future captures with this camera. As such, to improve matches, calibration patterns are used within the images to get more accurate matches. The calibration images used can be seen in figure 3-9.

Figure 3-9: The capture used for acquiring strongly matched points between the images, the image has been brightened for clarity. The calibration patterns used are highlighted in red. The patterns were chosen for easily identifiable corner points for matching between the unwarped images.

In addition to the calibration images, a manual feature matching implementation is proposed in section 3.4.2. The automatic feature process, however, implements RANSAC to attempt to get an accurate rotation matrix and was found to be sufficient. The pseudo-code for the feature matching process follows.

Feature Matching Pseudo-code

Result: Takes two HDR equirectangular TIFF images and returns
             the index locations of matched points.
Convert HDR TIFF images to grayscale;
Enhance grayscale images with histogram equalization;
Detect SURF features;
Extract features and relevant points;
Match features and return matched pairs;
Identify the indices of the matched points;
Return the index locations;

Algorithm 1: Feature detection and matching algorithm between two
HDR images.

3.4.2 Manual Feature Matching

As this feature matching step is being treated as a calibration step for the
camera in question, a manual feature matching option was implemented. The
automatic matching was found to be sufficient, however, the manual option
was left in the final code. The manual selection functions by displaying all
matched points between the two images and prompts the users to select a
selection of good matches to be used for computation.

A fully manual feature selection and matching process was also imple-
mented and selectable at the initial program launch. With this option, the
user is prompted to manually indicate a collection of points on the lower and
the upper image which are matched by the order of selection. As such the
user must select the matching points in the second image in the same order
as they were selected in the first image. This fully manual method does allow
the user to select points on images with poor SURF features.

3.4.3 Convert To Unit Sphere

The matched points are currently located within the 2D equirectangular pro-
jection, each image represents a 360-degree view and is centred at 0 by 0
degrees. The transformation requires the images to be rotated around the
imaginary sphere to leave the rotated image to appear centred at the edge
of the equirectangular projection. This is achieved by first transforming the

image, and the matched points, onto a 3D unit sphere.

Firstly the pixel index is re-centered at the middle of the image, as shown in figure 3-10. The re-centred index locations are then converted to degrees. As the total width is equivalent to 360 degrees and the height 180 degrees it is simple to divide the index by the number of pixels corresponding to each degree. The degrees are then converted to radians and the planar spherical coordinates are converted to Cartesian coordinates on a 3D unit sphere.
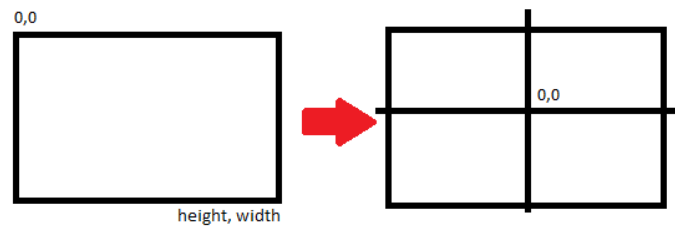
Figure 3-10: The equirectangular pixel indices are re-centered with (0,0) at the centre of the image.
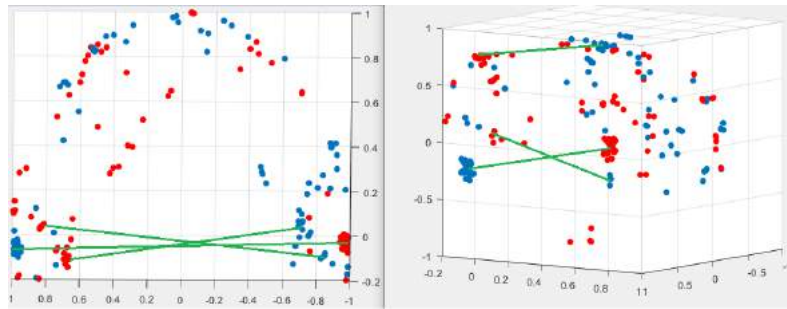
Figure 3-11: The 3D spherical projections of the non-rotated matched points. The top-down view is on the left and a side view of the same points is on the right. It can be seen that rotating one collection of points around the sphere will allow the matched points (some of which are indicated with green lines) to align.

Spherical Projection Pseudo-code

Result: Input 2D equirectangular indexed matched points and return
the same points on a 3D unit sphere in spherical coordinates.
Compute the pixels per degree in in the x and y directions;
Compute the coordinates of the center pixel;
Re-centre the pixel indices around the centre pixel; Convert the
coordinates to degrees by dividing by the pixels per degree in the
respective direction;
Convert to radians;
Concatenate the spherical coordinates with a $r = 0$.;
Convert spherical coordinates to Cartesian coordinates;
Return 3D Cartesian coordinates;

Algorithm 2: Convert matched points in an equirectangular projection
onto a 3D unit sphere.

3.4.4   Rotation Matrix Computation

The computation of the rotation matrix is computed via a RANSAC algorithm
which uses 7 random matched points and 3000 iterations to find the best
rotation matrix with a 0.3 radians inlier threshold. The rotation matrix with
the greatest inlier count is then kept.

Estimation of the rotation matrix required is performed via the Absolute
Orientation (ABSOR) tool [61]. This tool finds the rotation that best maps
one collection of points to another in the least squares sense. Based on a
quaternion method developed by Horn [62], the function returns a 3 by 3
rotation matrix. This rotation matrix is used to transform the second set of
points and the distance between matched points is computed via a distance
metric. The matrix which provides the most inliers is kept and returned.

The distance computation for deciding if a rotated point is an inlier or
not is a distance metric that computes the angular difference between the
two points on the 3D spherical surface. This is computed by finding the four
quadrant inverse tangent between the magnitude of the cross product between
the two-point vectors and the dot product of the same vectors.

RANSAC Pseudo-code

    Result: Returns the optimal rotation matrix and inlier match points
            from input matched points.

    Set up parameters for number of points required (n) which is 7 for the
     rotation matrix, iterations (k), and inlier threshold (t);

    for $1 : k$ **do**

        Select n random matched pairs;

        Compute rotation matrix (R) for the selected points;

        Apply rotation matrix to all points in the second image;

        Measure the distance between the rotated points and the
         unrotated matched points in the first image;

        Compute which of the rotated points that are within the distance
         threshold of the first image points;

        Store R and inliers;

    end

    Return the R which has the highest number of inliers;

Algorithm 3: RANSAC algorithm for computing the optimal rotation matrix.

3.4.5   Image Merging

Now the rotation matrix has been computed the second image can be rotated into alignment with the first and the images merged. The rotation matrix was computed on 3D Cartesian coordinates so the first step in rotating the image is to convert all the pixel indices to 3D Cartesian coordinates on a unit sphere. This is performed similarly to the process in section 3.4.3, however, instead of being performed only on the matched points the conversion is performed on all pixel indices.

Once the equirectangular image to be rotated has been mapped to a 3D unit sphere, the rotation matrix can be applied to each of the coordinates. The rotated Cartesian coordinates can then be converted back to equirectangular representation and the pixel value can be found via interpolation of the relevant original images pixels.

Now the rotated equirectangular image has been formed and is aligned with the first image the two images can be blended. The lens on this camera each have a field of view of 210 degrees, as such, it is safe to assume that 180

degrees of each image is needed to cover the full projection while minimising warped areas. The blending was performed by creating two masks, one for the first image which covers the $0-90$ and $270-360$ degrees in the horizontal axis, the second which covers the $90-270$ degrees. The edges of these two masks are complementary gradients and can be seen in figure 3-12. The masks are applied to each image and the images are then added to each other to give a resultant image.
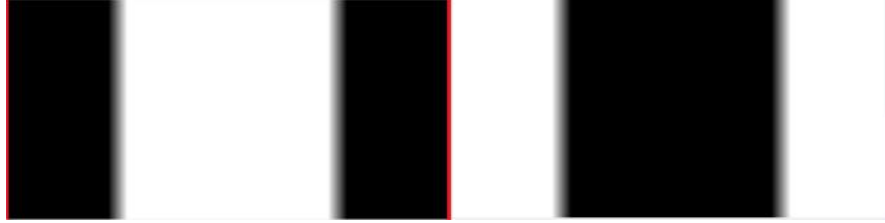
Figure 3-12: The masks that are applied to each image before combining. It can be seen they are complementary to each other and cover opposing regions.

Image Merging Pseudo-code

Result: Returns aligned and merged images from two unaligned
               images and a rotation matrix.
Convert the equirectangular image to be rotated to a 3D spherical
  projection as discussed in algorithm 2;
Rotate the image by the rotation matrix;
Convert back to equirectangular projection;
Create the two masks;
Apply masks to respective images;
Combine images;
Return resultant merged image;

Algorithm 4: The image rotation and merging algorithm for two equirect-angular images.

3.5 Colour Calibration

It is clear upon inspecting the merged image that there is a significant difference between the colours at the join, as can be seen in figure 3-13. This is most likely due to the two original HDR images having very different levels of

Figure 3-13: The combined image clearly shows the colour differences between the two images. This can be seen distinctively at the join between the two images.

overall brightness. As the data structure of the RAW image is in an uncapped uint16 format the same points in each image could have different RGB values associated to them. They will still be the same colour but at different intensity levels.

To overcome this a colour calibration of sorts was performed. This involved using the inliers from the alignment RANSAC step as matched points. These points are then used in a second RANSAC algorithm which computes an affine transformation for the RGB colour profile to transform one to the other. Here the same ABSOR function is used as in the alignment step but instead of locations, the pixel values are used.

The distance metric used is the ΔE metric [63]. ΔE is a colour distance or difference metric which is an alternative to the Euclidean distance that more accurately represents visual difference as opposed to mathematical difference. The ΔE value is computed in the LAB colour space as:

$$\Delta E = \sqrt{(L_2 - L_1)^2 + (a_2 - a_1)^2 + (b_2 - b_1)^2} \qquad (3.3)$$

A ΔE value of 2.3 corresponds to a just noticeable difference. This just noticeable difference value is used as the threshold value for the RANSAC algorithm. Using this metric did involve a transformation from the RGB colour space to the LAB colour space and as such normalisation to a 0-1 range from the uint16 format was required.

The computed affine transformation can then be applied in the image blending stage to the pixel values of the rotated image. Unlike the rotation matrix from the alignment stage, this colour transformation matrix will be different for each picture. As such, rather than being computed once and used every time, it will be computed for each pair of images.

Figure 3-14: The brightened colour corrected image, it can be seen that the colours between the images are much closer, albeit still not perfect. This is likely due to different exposures used in the two images.

Colour Calibration Pseudo-code

> **Result:** Computes an affine transformation matrix for transforming
>   colours using matched points.
> Get pixel values of matched points;
> Convert RGB values to LAB;
> Perform RANSAC on the matched points pixel values, a 3x3 affine
>   matrix is computed;
> Compute ΔE for all matched points;
> Label points with a ΔE value less than the just noticeable difference
>   threshold as inliers;
> Return the affine matrix with the most inlying points;

Algorithm 5: Generates an affine matrix for transforming the colours of one image to more closely match the second image.

3.6 Object Removal

A common problem with the capture of light probes is that a 360 capture, by nature of 360 degrees, also captures the photographer. Methods exist to remove objects such as this, as discussed earlier in section 2.3, but a fast simple solution is proposed and implemented here that is enabled by the use of a 360 camera. Once the camera is calibrated with the upper and lower alignment rotation matrix computed the following can be performed.

This method requires a minimum of three images taken with the photographer in different positions. These images are loaded and split into upper and lower parts as discussed previously. They are then undistorted, the colour is corrected between the halves, and then the halves are aligned using the calibrated rotation matrix and merged. All of this was discussed in detail in the previous sections.

The main difference occurs at this point. The individual 360 images need to be aligned with one image. This is performed similarly to the individual half alignments performed earlier. Features are extracted in each image and matched, the images are transformed to 3D unit spheres and RANSAC is performed to compute a rotation matrix to align each image to the first.

At this point, all images are undistorted, cover a full 360 degrees, and are aligned to the first image. The only difference between them is the area of the image occluded by the photographer. The median of the pixel values is taken between all the images. The pixels which are occupied by the photographer, in the majority of cases, will be significantly different to the occluded background. As such they will appear far from the median value. The median values for all pixel locations are used in the final reconstruction giving the resultant image with the photographer removed as shown in figure 3-15.

Figure 3-15: The left-hand panel shows the zoomed-in location of the photographer in a histogram equalised capture. The right hand-image shows the median pixels of the four images. It can be seen that the photographer is removed, this is due to the other three images being identical in this region so the median is one of those pixel values.

Object Removal Pseudo-code

> **Result:** Constructs a 360 light probe image from a collection of 360
>   captures and removes the photographer.
>
> Load each image;
> Split each image into upper and lower components;
> Undistort each component;
> Match colours between each half;
> Merge each upper and lower component;
> Extract features from first image;
> **for** *All other images* **do**
> > Extract features;
> > Match points to image one;
> > Convert points to 3D unit sphere;
> > Compute rotation matrix via RANSAC;
> > Rotate the image to align to first image;
>
> **end**
> Compute the median pixel value for each pixel between all aligned
>   images;
> Reconstruct resultant image from the median pixel values;

Algorithm 6: An algorithm for the removal of an object that moves location between a collection of images, such as a photographer.

Chapter 4

Results

This chapter outlines the capture, process and formation of an example 360 light probe using the method and code outlined in chapter 3. Intermediate results will be shown and discussed to identify any potential issues, artefacts and improvements to the method. The intermediate results that will be discussed include image capture, image unwarping, alignment between upper and lower components, colour correction, component merging, and finally object removal. Discussion points will be collated and expressed as conclusions to the project in chapter 5.

4.1   Image Capture

The image captured is a generic outdoors image with no close objects to help maximise the chance of success in the alignment step. An outdoor scene was chosen as the corners and edges of buildings act as strong, easily matchable features to help with the alignment step. A total of four images were taken with the photographer rotating around the camera, approximately 90 degrees in each capture, for the object removal step. The images were captured in raw format, the raw images can be seen in figure 4-1 along with histogram equalised versions of the images for ease of viewing. The second set of images was taken with the camera on the floor and the photographer walking around the camera, these can also be seen in figure 4-1. The first set of images is designed to test the image alignment between the 4 images. The second set was designed to prove the effectiveness of the object removal without any effects due to the alignment step.

Figure 4-1: Two sets of images were captured. The first 4 rows are one set, with rows 3 and 4 being histogram equalised versions for visualisation. Each set has 4 images with the photographer in a different position in each one. The first set was taken with the camera in the hand of the photographer, the second with the camera static on the floor and the photographer a couple of meters away.

4.2 Unwarping

The image unwarping was performed using the camera calibration parameters found earlier via the OcamCalib toolbox. The exact parameters can be seen in section 3.3. Each image is transformed separately with the respective lens calibration parameter. The rear-facing lens has its image flipped horizontally to account for the negative direction in relation to the first image when unwarping into an equirectangular format.

Figure 4-2: The unwarped images. The first 4 rows are one set, with rows 3 and 4 being histogram equalised versions for viability.

4.3   Upper & Lower Alignment

The rotation of the second image component into alignment with the first is performed using the rotation matrix computed previously using the calibration images as discussed in section 3.3. The rotation matrix used was as follows:

$$R = \begin{bmatrix} -0.99591 & -0.089274 & -0.01383 \\ 0.089566 & -0.99574 & -0.022112 \\ -0.011797 & -0.02326 & 0.99966 \end{bmatrix} \tag{4.1}$$

The rotation matrix, R, can be interpreted as a close to 180-degree rotation about the z-axis, as shown in figure 4-3. This is to be expected as the cameras are positioned at 180 degrees to each other plus some translation due to the offset of the lenses which is most likely the cause of the variation from the 180-degree rotation in R.
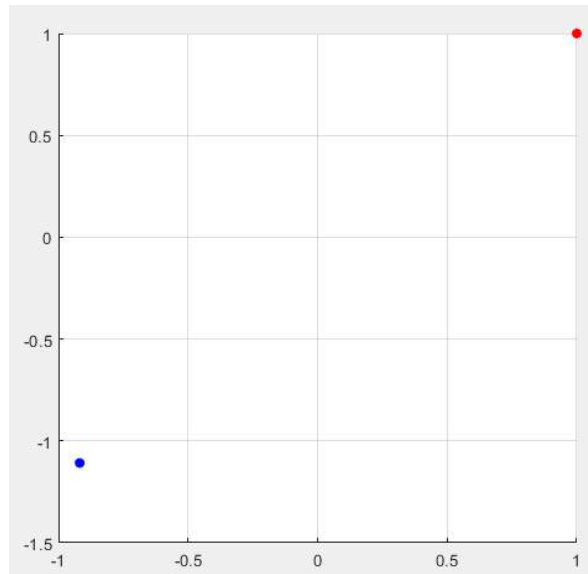
Figure 4-3: A visualisation of the rotation matrix. The red point at (1,1,1) was rotated by R resulting in the blue point, which is 0.035 below the plane. It can be seen that the result is close to a 180-degree rotation about the z-axis.

This matrix was used for all pairs of images, this is because the camera used was identical and the lens position is fixed in relation to the other lens. As such the rotation matrix will be the same for every image captured. The rotation matrix computed earlier was shown to provide a good alignment between the

components so no further computation for the alignment was required. The alignment could be improved further by attempting a more robust RANSAC procedure with more iterations and more matched points, however, in this case, the alignment is adequate for the proof of concept without significantly increasing the computation time. The two sets of aligned images can be seen in figure 4-4.

Figure 4-4: The two sets of aligned images. Colour discrepancies can be seen as the images have not been corrected in this version. The photographer can also be seen in both sets.

4.4 Colour Correction

It is clear upon inspection of the aligned images in figure 4-4 that the colours between the two merged image components are not matched, one component is significantly darker than the other. This is due to one lens being pointed towards a light source, in this case the sun, and the other being pointed away from the light source. The colour correction is performed before the merging step via a RANSAC method using matched points as discussed in section 3.5. The colour corrected images can be seen in figure 4-5.

Figure 4-5: The two sets of colour corrected images. It can be seen that the second set had the colours correct much more successfully than the first. In the first set, various colour errors occurred with blue and red tints being applied in the colour correction. In the second set, the correction appears good apart from the final image picking up a small red tint.

The errors in the colour correction are likely due to the matched points used in the RANSAC process. It is likely that there were poorly matched features which caused the computation of the affine matrix to be inaccurate. RANSAC probably failed here as many of the well-matched features were on the red brick houses with a much smaller amount on the rest of the scene, as such the colour matching between the houses most likely overwhelmed the other points. Manually choosing points for the colour correction would likely provide a more accurate result but would require manual selection for every image taken. As such the non-colour corrected images will be used for the object removal step as it proved very unreliable to get a series of images that had the colours corrected accurately.

4.5   Object Removal

The object removal and formation of the final light probe image were performed on both sets of images. Firstly, the summation of the absolute differ-

ences between the four 360 degree images where the camera was static can be seen in figure 4-6. Here it can be seen that the photographer moves between each capture and that all four images are perfectly aligned. As such no further alignment is required between the images and all that is needed is the median pixel value for each pixel. This final result can be seen in figure 4-7.

Figure 4-6: A summation of the differences between the four images. The photographer can be seen at different positions in each of the four images. The rest of the background is static and unchanging as the camera is static on the floor, the only exception is some cloud movement in the sky.

Figure 4-7: The combined image with the photographer removed. This image can be used as an HDR light probe, the combination of the images was effective as they did not need any further aligning. A histogram equalised version can be seen in figure 4-8

Figure 4-8: Histogram equalised version of figure 4-7. It can be seen that the image components are well aligned and the photographer has been successfully removed from the image.

This image is a great success for the method. However, a key issue with this image is that the camera was static. In the other set of images, the camera was held and rotated with the photographer. What is shown in figure 4-9 is a composite image of the four 360 images of the handheld set after they have been aligned with each other. The magenta and green show differences in intensities between the images. It is clear that they are not well aligned, there is misalignment between some of the images despite the RANSAC rotation alignment. This could be due to poor matches between the images and thus a poor rotation matrix computed via RANSAC. What is more likely is that the camera undergoes some translation rather than just rotation, it is near impossible to hold a camera completely static. This could be solved potentially by also computing the translation between each image and applying it. However, manual feature selection could also be used to help compute a better rotation matrix.

Despite the slight misalignment, the final output image is fairly effective as a light probe. The photographer is mostly removed from the image, as can be seen in figure 4-10, and the blurring is fairly minor. For mid-long distance reflections using IBL, this image would be adequate in use as an HDR light probe. However, it can be seen in the histogram equalised version of the image in figure 4-11 that the photographer is not fully removed. This is most likely due to the misalignment of the images meaning the range of pixel values is greater and the median value is not so well defined as the background colour.

The histogram equalised version of the static camera image with the pho-

tographer removed can be seen in figure 4-8. On closer inspection of the brightened image, it can be seen that the photographer has been successfully removed. It is clear that the static captured images can be used to make a highly successful light probe which successfully ignores moving objects. In addition to this, in its current implementation, that the held camera capture has flaws within the alignment of the images.

One interesting point to note in regards to the colours of the final image is that in the hand-held case where the camera is rotated, the colours end up appearing normalised across the whole image. This is due to the light source, that causes the mismatch across the two lenses, moves in relation to the camera. This can be seen more clearly in 4-11. So that when the medium value for each pixel is taken the image appears to be colour corrected, although it is not a true correction and errors can be more accurately seen in the histogram equalised image.
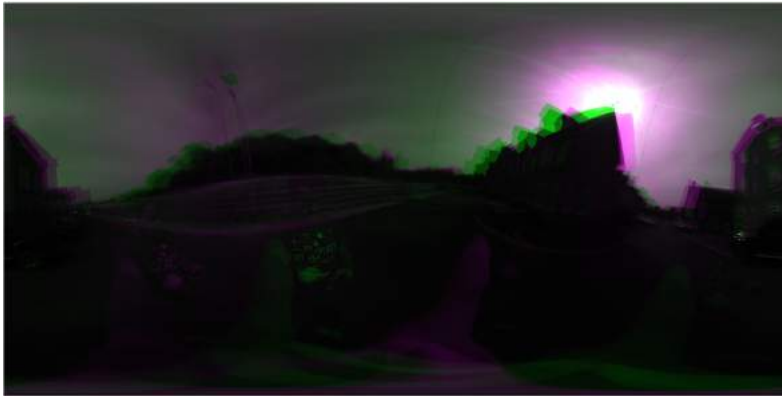
Figure 4-9: A composited image of the four 360 degree images of the handheld set with each image in a different colour band. The magenta and green shows where there are different intensities. It can be seen that the images change slightly between each frame, even after the alignment between them.

Figure 4-10: The combined image with the photographer removed. The photographer was effectively removed apart from the small portion at the base of the image where the camera was held. The image is blurred and clearly, some areas between the images do not align well.
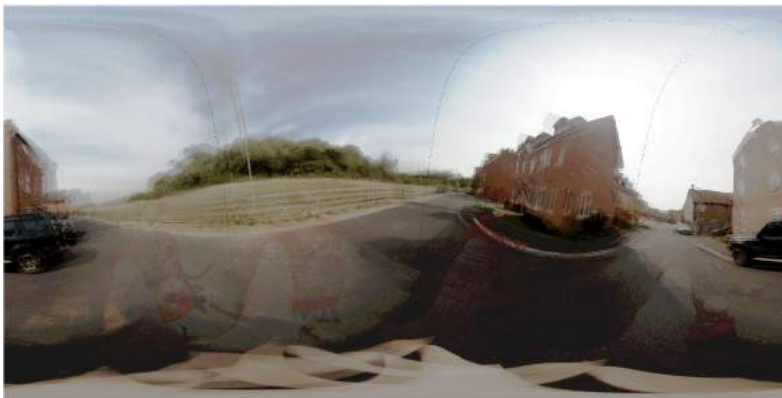
Figure 4-11: Histogram equalised version of figure 4-10 for improved visualisation. Key points to note are the misalignment of edges and some objects such as the lamp post and house. Also, note that the photographer is not fully removed.

Chapter 5

Conclusions and Future Work

In this chapter, conclusions will be drawn based on the work performed in the earlier chapters and the results discussed previously. Potential improvements to the method and implementation will be discussed as well as successes identified. There will then be a brief discussion on potential future work and extensions to the method.

5.1 Successes & Conclusions

The main goal of this project was to form a functional HDR light probe from images captured using a 360 camera. A 360-degree HDR light probe image was successfully produced from two unaligned images captured by the two cameras, each with a 270-degree field of view as indicated by the manufacturer. This was achieved by unwarping the raw images to equirectangular projections, features were then matched between the image and transformed to 3D spherically projected points. A RANSAC process was then used to find the best rotation matrix to rotate the second image around the unit sphere to align it with the first image. The inliers of the RANSAC process are then used as matched points for the RANSAC computing of an affine transformation matrix to be used for transforming the pixel values of one image to align them more accurately with the colours in the other image. The colour corrected, aligned images are then merged into the final HDR light probe using complimentary image masks.

The HDR light probes formed this way proved to be effective light probes with good alignment. It is worth noting that the images used for this process

were designed to produce good matched points and a strong alignment, the rotation matrix computed via this method can be used as a calibration matrix for aligning all image pairs.

This method of creating a light probe via a single capture is highly successful but experiences a couple of problems. Firstly, the photographer can be seen in the image. An object removal method was then proposed and demonstrated. Two sets of four images were captured, one set with the camera static on the floor and one set with the camera held by the photographer and rotated. The four images are then aligned and the median pixel values used to remove the photographer. The static camera approach proved to be highly successful with the photographer fully removed from the final image and no alignment issues as the camera was static. The handheld method was less effective as there were alignment issues between the four images, it proved to be difficult to get a successful RANSAC rotation matrix between the images. These alignment issues also led to poor inlier selection so the colour correction proved ineffective in the handheld case.

To summarise, a single 360-degree capture can successfully be used to form an HDR light probe with effective colour correction between the two components. In addition, an object such as a photographer was successfully removed using a collection of images from a static camera to form the HDR light probe. Finally, a handheld approach to the object removal was attempted and had minor success but suffered from alignment issues.

5.2   Potential Improvements

The main area of the method which requires further refinement is the multi-image alignment step within the object removal for the handheld camera set. It proved to be difficult to get good feature matching and alignment using the RANSAC method implemented. This was due to some scenes having poor features and difficulty matching points between images, another issue was the translation of the camera between captures which was impossible to account for in a rotation alone. One possible improvement is a more robust RANSAC implementation with more iterations, this would be a more computationally demanding process and significantly slow down the process, there is also a chance that there would still be alignment issues as the quality of the matched points varies depending on the scene. A second solution would be to manually

select points to compute the rotation matrix, which would be a slow and tedious additional process. An additional issue which may be the main problem with the alignment issues is the possible translational movement of the camera during capture.

A secondary area of improvement is within the colour correction algorithm. The method is effective but can provide poor affine transformation matrices if the matched points are poor. In the single image process, the RANSAC inliers from the component alignment are used as the matched points for the colour correction. In the multi-image process, there are no inliers from this step as it is not computed due to the use of the pre-computed rotation matrix. As such, all matched points are used within a RANSAC process for the affine matrix computation. Similarly to the multi-image alignment, it proved to be difficult to get a good result and the best results were acquired not using the colour correction step. Again, similarly to the multi-image alignment, a manual process could be used to select matching points but it would be a slow process which may not be fully effective. Potentially a more successful approach would be to use the calibrated rotation matrix to compute inlying points and use those points to get the pixel values for the colour correction. An alternative approach would be to normalise the full image in relation to the brightest point across the two components in an attempt to correct discrepancies due to varying brightness.

5.3 Future Work

This section discusses, a collection of possible extensions and future avenues of work on this topic. The improvements mentioned before should be a priority; that or an alternative form of computing or verifying the matched points for use in the alignment and colour correction steps is required. As a more substantial objective, it would be interesting to extend this method to frames of 360-degree video. This amount of captures could potentially be used to create a 3D visual map of an area which could be used as an environment map for synthetic lighting that includes foreground object occlusion, this could be achieved by triangulating matched points between video frames. The 3D points could then be used to render a rough 3D projection of the environment.

An alternative method for allowing a synthetic object to move through a scene and accurately reflect the environment would be to capture a small

collection of HDR light probes in a local region and interpolate between them as the synthetic object moves. This would not provide accurate reflections and not take into consideration occlusions but potentially would be sufficient for rough reflections.

Bibliography

[1] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 189–198, New York, NY, USA, 1998. ACM.

[2] Saulo A. Pessoa, Guilherme de S. Moura, Joao Paulo S. do M. Lima, Veronica Teichrieb, and Judith Kelner. Rpr-sors: Real-time photorealistic rendering of synthetic objects into real scenes. *Computers & Graphics*, 36(2):50 – 69, 2012. Virtual Reality in Brazil 2011.

[3] HDR bracketing. https://promotesystems.com/pages/hdr-bracketing. Accessed: 18/05/2018.

[4] P. Debevec. Image-based lighting. *IEEE Computer Graphics and Applications*, 22(2):26–34, March 2002.

[5] David Briggs. Sphere reflections. http://www.huevaluechroma.com/021.php. Accessed: 26/05/2018.

[6] Ricoh Theta V HDR 360 Camera. https://theta360.com/en/about/theta/v.html. Accessed: 18/05/2018.

[7] 360 Image artefacts. https://www.forbes.com/sites/geoffreymorrison /2016/09/24/360-degree-camera-tips-and-tricks-get-the-best-photo-and-video-spheres/#e878cde3d8ff. Accessed: 18/05/2018.

[8] Kusuma Agusanto, Li Li, Zhu Chuangui, and Ng Wan Sing. Photorealistic rendering for augmented reality using environment illumination. In *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed*

and Augmented Reality, ISMAR '03, pages 208–, Washington, DC, USA, 2003. IEEE Computer Society.

[9] Insta360 One Camera. https://www.insta360.com/product/insta360-one/. Accessed: 27/08/2018.

[10] Davide Scaramuzza. OCamCalib: Omnidirectional camera calibration toolbox for Matlab. https://sites.google.com/site/scarabotix/ocamcalib-toolbox. Accessed: 27/08/2018.

[11] Jonas Unger and Stefan Gustavson. High-dynamic-range video for photometric measurement of illumination. In *Proc.SPIE*, volume 6501, pages 6501 – 6501 – 10, 2007.

[12] James F. Blinn and Martin E. Newell. Texture and reflection in computer generated images. *Commun. ACM*, 19(10):542–547, October 1976.

[13] Samsung Galaxy S9+ review: More of the same, yet better overall. http://libproxy.bath.ac.uk/login?url=https://search-proquest-com.ezproxy1.bath.ac.uk/docview/2015479739?accountid=17230. 27/08/2018.

[14] J. A. Ferwerda. Elements of early vision for computer graphics. *IEEE Computer Graphics and Applications*, 21(5):22–33, Sep 2001.

[15] Ye Zhengnan, Qiu Jueqin, Xu Haisong, Luo M. Ronnier, and Westland Stephen. 81-4: Image quality evaluation of HDR displays. *SID Symposium Digest of Technical Papers*, 48(1):1192–1195.

[16] A. El-Mahdy and H. El-Shishiny. High-quality HDR rendering technologies for emerging applications. *IBM Journal of Research and Development*, 54(6):8:1–8:15, Nov 2010.

[17] Jiangtao Kuang, Hiroshi Yamaguchi, Changmeng Liu, Garrett M. Johnson, and Mark D. Fairchild. Evaluating HDR rendering algorithms. *ACM Trans. Appl. Percept.*, 4(2), July 2007.

[18] M. Robart, S. Hill, and E. Tanguy. Global illumination for advanced computer graphics. In *2008 Digest of Technical Papers - International Conference on Consumer Electronics*, pages 1–2, Jan 2008.

[19] T. Mitsunaga and S. K. Nayar. Radiometric self calibration. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 1, page 380 Vol. 1, 1999.

[20] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High dynamic range video. *ACM Trans. Graph.*, 22(3):319–325, July 2003.

[21] Greg Ward. Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures. *Journal of Graphics Tools*, 8(2):17–30, 2003.

[22] HDRShop. http://www.hdrshop.com/. Accessed: 17/05/2018.

[23] HDR sensor camera. http://www.ovt.com/hdr. Accessed: 26/05/2018.

[24] N. Greene. Environment mapping and other applications of world projections. *IEEE Computer Graphics and Applications*, 6(11):21–29, Nov 1986.

[25] Michael. O'Rourke. *Principles of three-dimensional computer animation, modeling, rendering, and animating with 3D computer graphics*. Norton, 1995.

[26] James D. Foley. *Introduction to Computer Graphics*. Addison-Wesley, 1993.

[27] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. *SIGGRAPH Comput. Graph.*, 18(3):213–222, January 1984.

[28] Arthur Appel. Some techniques for shading machine renderings of solids. In *Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference*, AFIPS '68 (Spring), pages 37–45, New York, NY, USA, 1968. ACM.

[29] Greg Ward. Radiance synthetic imaging system. http://radsite.lbl.gov/radiance/. Accessed: 17/05/2018.

[30] Image based lighting with Mental Ray. https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/GUID-F6833C2C-3795-414D-972D-4EFC4A3D123D-htm.html. Accessed: 17/05/2018.

[31] Image based lighting with Cycles. https://docs.blender.org/manual/en/dev/render/cycles/world.html. accessed: 17/05/2018.

[32] Image based lighting with Eevee. https://wiki.blender.org/index.php/Dev:2.8/Source/Viewport/Eevee. Accessed: 17/05/2018.

[33] Image based lighting with Ambient Cubemaps. https://docs.unrealengine.com/en-us/Engine/Rendering/LightingAndShadows/AmbientCubemap. Accessed: 17/05/2018.

[34] Experimental image based lighting with Look Dev. https://docs.unity3d.com/2018.1/Documentation/Manual/LookDev.html. Accessed: 17/05/2018.

[35] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics*, Volume 36 Issue 6 Article No. 176, 2017.

[36] Avijit Lahiri. *Basic Optics - Principles and Concepts.* Elsevier, 2016.

[37] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 369–378, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[38] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. 06 - the human visual system and hdr tone mapping. In Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, editors, *High Dynamic Range Imaging*, The Morgan Kaufmann Series in Computer Graphics, pages 187 – 221. Morgan Kaufmann, San Francisco, 2006.

[39] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. 07 - spatial tone reproduction. In Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, editors, *High Dynamic Range Imaging*, The Morgan Kaufmann Series in Computer Graphics, pages 223 – 323. Morgan Kaufmann, San Francisco, 2006.

[40] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec. 08 - frequency domain and gradient domain tone reproduction. In Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, editors, *High Dynamic Range Imaging*, The Morgan Kaufmann Series in Computer Graphics, pages 325 – 365. Morgan Kaufmann, San Francisco, 2006.

[41] T. Ho and M. Budagavi. Dual-fisheye lens stitching for 360-degree imaging. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2172–2176, March 2017.

[42] Kevin Matzen, Michael F. Cohen, Bryce Evans, Johannes Kopf, and Richard Szeliski. Low-cost 360 stereo photography and video capture. *ACM Trans. Graph.*, 36(4):148:1–148:12, July 2017.

[43] Panono 360 Camera. https://www.panono.com/. Accessed: 18/05/2018.

[44] Theta 360 Camera. https://greyscalegorilla.com/theta360/. Accessed: 18/05/2018.

[45] Photoshop Image Stack Mode. https://helpx.adobe.com/photoshop/using/image-stacks.html. Accessed: 18/05/2018.

[46] Jing Wang, Ke Lu, Daru Pan, Ning He, and Bing kun Bao. Robust object removal with an exemplar-based image inpainting approach. *Neurocomputing*, 123:150 – 155, 2014. Contains Special issue articles: Advances in Pattern Recognition Applications and Methods.

[47] Seyed Saeid Mirkamali and P. Nagabhushan. Object removal by depth-wise image inpainting. *Signal, Image and Video Processing*, 9(8):1785–1794, Nov 2015.

[48] J.P. Lewis. Fast template matching. *Vis. Interface*, 95, 11 1994.

[49] Erum Arif Khan, Erik Reinhard, Roland W. Fleming, and Heinrich H. Bülthoff. Image-based material editing. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 654–663, New York, NY, USA, 2006. ACM.

[50] Paul Debevec. A median cut algorithm for light probe sampling. In *ACM SIGGRAPH 2008 Classes*, SIGGRAPH '08, pages 33:1–33:3, New York, NY, USA, 2008. ACM.

[51] Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. All-frequency shadows using non-linear wavelet lighting approximation. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, pages 376–381, New York, NY, USA, 2003. ACM.

[52] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 497–500, New York, NY, USA, 2001. ACM.

[53] T. Rhee, L. Petikam, B. Allen, and A. Chalmers. Mr360: Mixed reality rendering for 360 panoramic videos. *IEEE Transactions on Visualization and Computer Graphics*, 23(4):1379–1388, April 2017.

[54] Thomas Iorns and Taehyun Rhee. Real-time image based lighting for 360-degree panoramic video. In Fay Huang and Akihiro Sugimoto, editors, *Image and Video Technology – PSIVT 2015 Workshops*, pages 139–151, Cham, 2016. Springer International Publishing.

[55] Saghi Hajisharif. Real-time image based lighting with streaming hdr-light probe sequences. page 37, 2012.

[56] Joel Kronander, Francesco Banterle, Andrew Gardner, Ehsan Miandji, and Jonas Unger. Photorealistic rendering of mixed reality scenes. *Comput. Graph. Forum*, 34(2):643–665, May 2015.

[57] Christian Bloch. *The HDRI Handbook: High Dynamic Range Imaging for Photographers and CG Artists +DVD*. Rocky Nook, 2007.

[58] Dacid Coffin. dcraw - Decoding raw digital photos. https://www.cybercom.net/~dcoffin/dcraw/. Accessed: 27/08/2018.

[59] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A flexible technique for accurate omnidirectional camera calibration and structure from motion. *Proceedings of IEEE International Conference of Vision Systems (ICVS'06)*, January 2006.

[60] Davide Scaramuzza. *Omnidirectional Vision. From Calibration to Root Motion Estimation.* PhD thesis, ETH Zurich, 2007.

[61] Absolute Orientation - Horn's Method. https://uk.mathworks.com/matlabcentral/fileexchange/26186-absolute-orientation-horn-s-method. Accessed: 27/08/2018.

[62] Berthold KP Horn, Hugh M Hilden, and Shahriar Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *JOSA A*, 5(7):1127–1135, 1988.

[63] A. R. Robertson. Historical development of cie recommended color difference equations. *Color Research & Application*, 15(3):167–170.

Appendices

Appendix A

Calibration Images

A.1 Upper Lens

A.2 Lower Lens