# Controllable Light Stage

A compact capture system for object modelling and relighting

## Mid Term Report

Alp Bora Orgun

Department of Electronic Engineering

Faculty of Engineering and Physical Sciences

University of Surrey

# Abstract

This report introduces and goes over the Controllable Light Stage Project, which aims to develop a compact system to capture and reconstruct relightable objects in 3D. A hardware prototype was developed which can successfully capture a dataset of an object for 3D Reconstruction. An accompanying software will be developed which will use techniques discussed in this report to reconstruct the object in 3D. The report also goes over details of the construction of hardware prototype, the work plan for the project and any future work required to complete the project is discussed.

# Table of Contents

# Introduction

## A Brief Overview of Light Stages History and Use

Light stages are specialized stages used for capturing the appearance of real-world objects and people under a wide range of lighting conditions. They are used in the production of films, television series, and video games to create realistic and accurate digital representations of real-life objects and characters.

The use of light stages for capturing the appearance of real-world objects dates to the early 2000s, when Light Stage 1 was built by Paul Debevec and his team at University of California at Berkeley[1]. One of the earliest examples of a light stage being used was in a film production. Light Stage 2 which was also built at University of California at Berkeley and was used in the movie Spider-Man™ 2. The improved Light Stage 2 featured thirty bright strobe lights a ten-foot semi-circular arm compared to its predecessors one small spotlight. Light Stage 2 was used in numerous other movies as well[2].

Further development of the Light Stage technology was seen in research such as The Digital Emily Project which created the first photoreal digital actor. Using this technology, scenes were created which were accepted as indistinguishable from real footage of Emily[3].

The process has been used in many production movies such as the 2009 film "Avatar" which has won an Oscar award in visual effects[4], GI: Joe, Endhiran, TRON: Legacy and more[5].

Since their inception, light stages have become an important tool in the film, television and video game industries and have been used in the production of numerous blockbuster films and video games to create immersive simulations by realistically capturing and simulating objects and people's reflections under all possible lighting[6].

## Aims and objectives of this project

The aim of this project is to develop a compact light stage for capture and relightable reconstruction of objects in 3D. To achieve this a hardware and a software system is being developed. The main motive behind this project is to recreate real physical objects in a digital environment to be used in digital media such as video games. There are existing light stage solutions however most of these are big industrial devices that have high costs to operate. Controllable light stage being developed in the context of this project is compact and low cost.

Main objectives of the projects:

1. A hardware system that can light and capture an image from multiple angles
2. A user interface that allows for full control over all the lighting conditions the hardware system is capable of.
3. A software system that can reconstruct a relightable 3D Object from the captures made using the hardware system.

# Literature Review

Creating realistic computer-generated imagery has always been a goal in computer graphics dating back to one of the first ever 3D models, the Utah teapot which was created in 1975[7]. Since then, countless new techniques have been developed to reconstruct real-life in a virtual 3D environment.

There are many ways to capture images which allow an object to be reconstructed in 3D. A compact light stage is a system which allows for the capture of these images in a certain programmatic way. Light Stages can also be constructed in varying ways as it can be seen with the Light Stages 1, 2, 3 and others built at University of California at Berkeley[2]. If the images are captured in a way that fits into the technique that will be used to reconstruct the object, the two different steps are separate and can be abstracted away.

There are many ways to reconstruct objects in 3D. This report will be focusing on photogrammetry-based techniques which use images to acquire, depending on the technique, different properties of the image such as the surface normal, the radiance maps and more to generate 3D meshes, relightable models and synthesized views.

In this literature review a small overview of camera calibration is studied and different methods and algorithms for 3D object reconstruction were reviewed. Finally, a small summary of all the techniques is written.

As the scope of this project covers both the hardware (the light stage itself) and the software (used to reconstruct the 3D object) it is important to note that the literature review has been constrained to the theoretical techniques which will be used in the implementations of the software as the hardware design of the light stage has been pre-determined at the start of the project.

## Light Stage Setups

### Camera Calibration

Camera calibration is the process of estimating the internal properties such as the focal length, principal point, and distortion coefficients and the relative position and the pointing direction of the camera in the 3D scene. These are called the intrinsic and extrinsic parameters respectively[8]. These parameters are necessary for accurately estimating the 3D positions of points in the scene from 2D images taken by the camera[9].

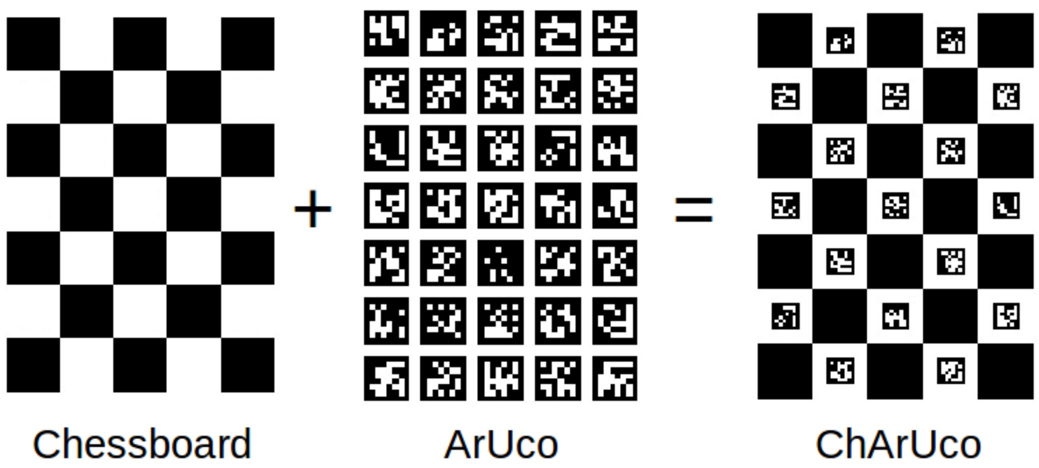


*Figure 1 - Chessboard Pattern and ArUco markers combined to create a ChArUco board used for camera calibration*[10]

Calibrating the camera is typically an iterative process that involves taking multiple images of a scene with known 3D points (such as a ChArUco board) from different viewpoints and using these images to estimate the intrinsic and extrinsic parameters of the camera[11]. Once the camera is calibrated, you can use the estimated parameters to accurately estimate the 3D positions of points in the scene from the 2D images.

There are many techniques which can be used to calibrate both the intrinsic and extrinsic parameters of the camera. Two widely used techniques are briefly explained which use different principles to calibrate the camera.

### Zhang Method

One technique to calibrate the camera is called the Zhang method. The technique uses the camera to capture at least two different orientations of a planar pattern with either the pattern or the camera freely moved[12].

### Tsai Calibration

Another technique which can be used to determine the intrinsic parameters of a camera is called the Tsai calibration. Using a pinhole camera model and a single image calibration is possible by solving projection equations[13].

## 3D Reconstruction – 3D Structure, Surface shape and Reflectance properties

### Photometric Stereo

Photometric Stereo is a technique used to recover the 3D shape of an object from images which were taken with incident illumination from different directions. The basic process is capturing images of a static object from a static viewpoint while varying the incident illumination direction. The set of images captured are then used to calculate the surface normals at each point of the object's surface[14].

The principal idea behind photometric stereo is that intensity of reflected light on a point on the target object's surface is dependent on two factors, the surface normal and the direction of the incoming light. Using this it is possible to estimate the surface normal at each point by looking at the intensity of the light reflected from the surface[14].

This information then allows for the reconstruction of the 3D model of the object.

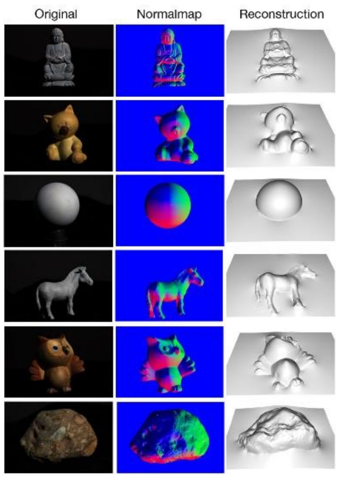


*Figure 2 - Example results from calibrated photometric stereo*[15]

This technique first introduced by Woodham[14] has been very successful at recovering shape and reflectance properties however it has some limitations.

Constraints and Assumptions of Photometric Stereo include:

- Surface assumed to be Lambertian (Diffuse)
- All light sources as point light sources at infinity
- One single viewpoint
- Require camera and light sources to be calibrated

However, in realistic conditions most surfaces tend to have specular properties, point light sources cannot be an ideal directionally light located at infinity. For certain use cases such as full 3D Reconstruction and relighting of an object a single viewpoint is not enough. In some cases, it is also not possible to have calibrated camera and light sources and hence calibration may be required.

Photometric Stereo has been around for more than 40 years hence has been studied very deeply. Multiple different articles have provided solutions for the different constraints of photometric stereo.

Using multi-view photometric stereo which extends photometric stereo method to allow for multiple view which allows for a full 3D Reconstruction of the object instead of from an only single viewpoint which usually resulted in a 2.5D Reconstruction rather than a full 3D one. This technique also uses silhouettes and a visual hull to calibrate camera and lighting positions allowing for a fully uncalibrated setup.[16]

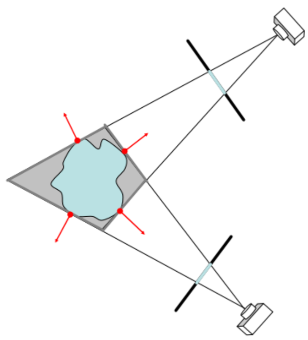

*Figure 3 – Visual Hall Illustration from Multi-view photometric stereo*[16]

Further improvement to this technique allows for surface normals of specular surfaces to also be obtained using a distributed light source[17].

For uncalibrated light sources there are a couple of techniques that allow for the calibration of the lights. One technique is to use the specular highlights observed on a captured sphere to determine the light directions. This technique can also be used to calibrate the camera as well[18].

## NeRF: Neural Radiance Fields

NeRF is a state-of-the-art technique for view synthesis. Based on the idea that a 3D scene can be represented as a continuous volumetric scene function that maps 3D positions to the radiance (intensity and colour) of light emanating from the positions using a sparse set of input images.

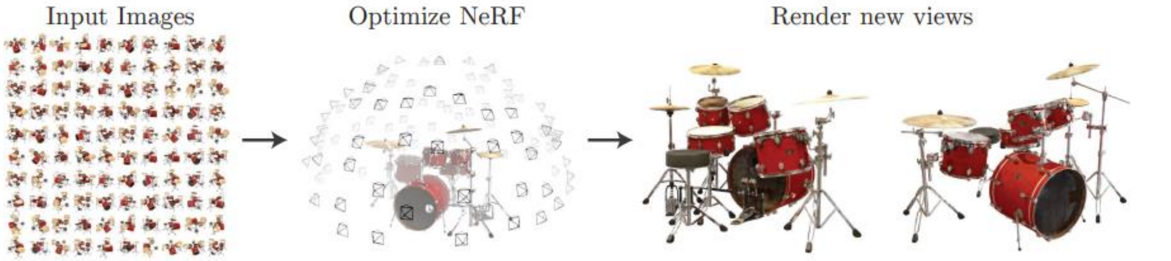


*Figure 4 - Example of input images and rendered output of NeRF*[19]

The continuous volumetric function is learned using a neural network which has been trained on a dataset of input images and ground truth radiance values. Using the trained network new views can be synthesised by sampling the continuous radiance function.

Views are synthesized by querying a 5D coordinate along the camera ray and classic volume rendering techniques are used to project the output colours and densities into an image.

It is also possible to convert NeRF into a mesh using the marching cubes technique[20].

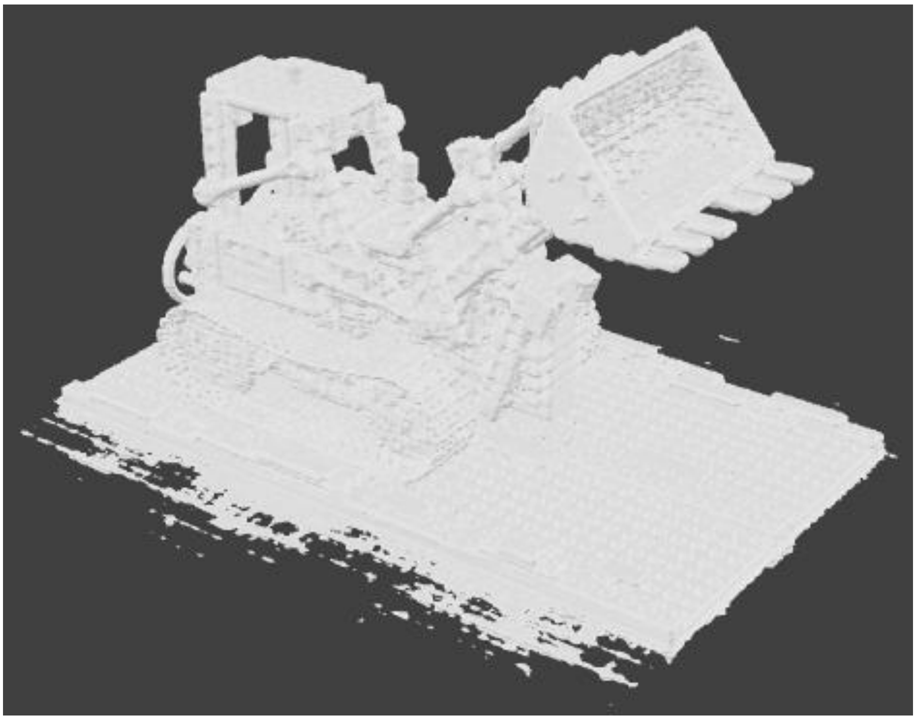

*Figure 5 - Mesh acquired from NeRF using marching cubes*[21]

Advantages of NeRF are that NeRF can synthesis views with high geometric and photometric complexity such as objects with non-Lambertian reflectance, transparency and can handle occlusion and depth discontinuities.

However, there are several disadvantages to NeRF. NeRF requires a large dataset of input values and corresponding grand truth radiance values to train the neural network. Acquiring the dataset can be time-consuming and expensive. It is also computationally expensive to synthesize new views using NeRF as the neural network must be sampled at multiple positions.

There are further studies and extension on NeRF to improve its performance and capabilities.

One study which allows for real-time view synthesize is Baking Neural Radiance Fields for Real-Time View Synthesis. Using Sparse Neural Radiance Grids (SNeRG) which are generated by precomputing the continuous function and storing (baking) it into a discrete SNeRG it is possible to achieve real-time performance for view synthesis[22].

Another extension to NeRF called NeRV (Neural Reflectance and Visibility Fields) allow for the creation of relightable views[23].

# Development to Date

Currently a working controllable light stage has been assembled. 3D printed parts made from PLA plastic have been produced to act as the various structures for the light stage to mound the LED Strip and to act as the display turntable surface.

A Raspberry Pi 3B+ is used to control the system controlling an Arduino UNO to control the LED Strip which act as point-like light sources, a stepper motor driver which controls a stepper motor to control the display turntable and a DSLR Camera to capture images.

## Overview

Systems level diagram is provided in Figure 7. Showing how all the different parts of the hardware are connected. The Arduino UNO is controlled by the Raspberry Pi using USB Serial through the provided USB ports on the Raspberry Pi and the Arduino UNO. The stepper motor driver is connected to the Raspberry Pi through the GPIO Pins, and the camera is controlled through USB Interface. LED Strip is powered and controlled by the Arduino UNO through GPIO Pins.

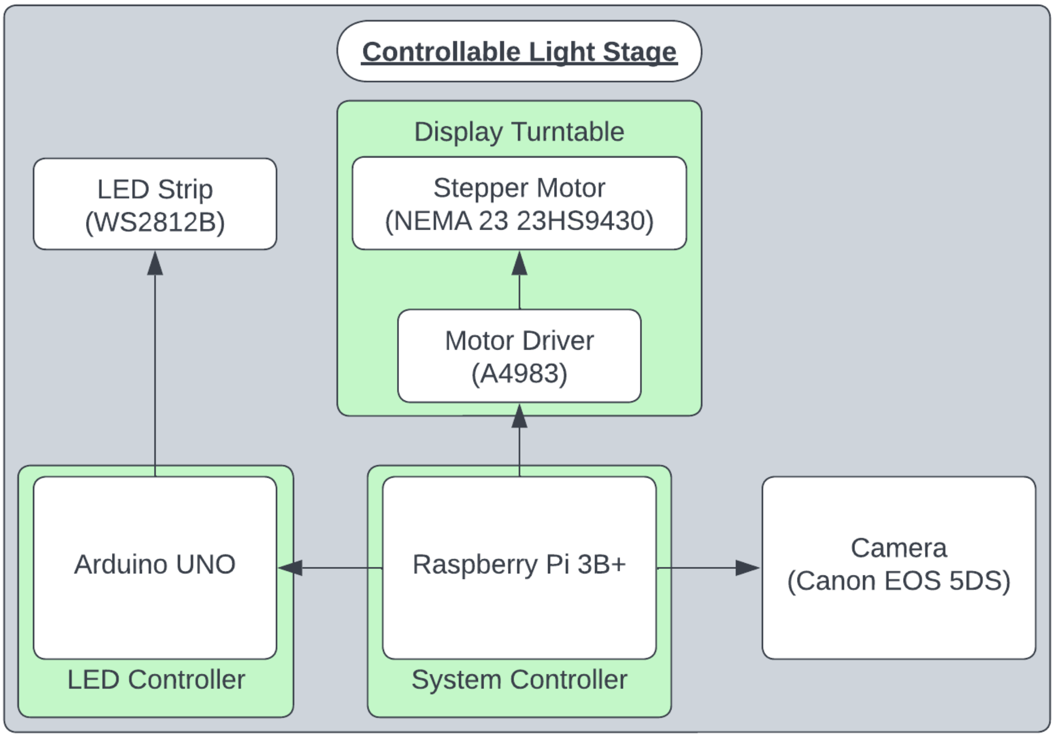


*Figure 6 System Level Diagram*

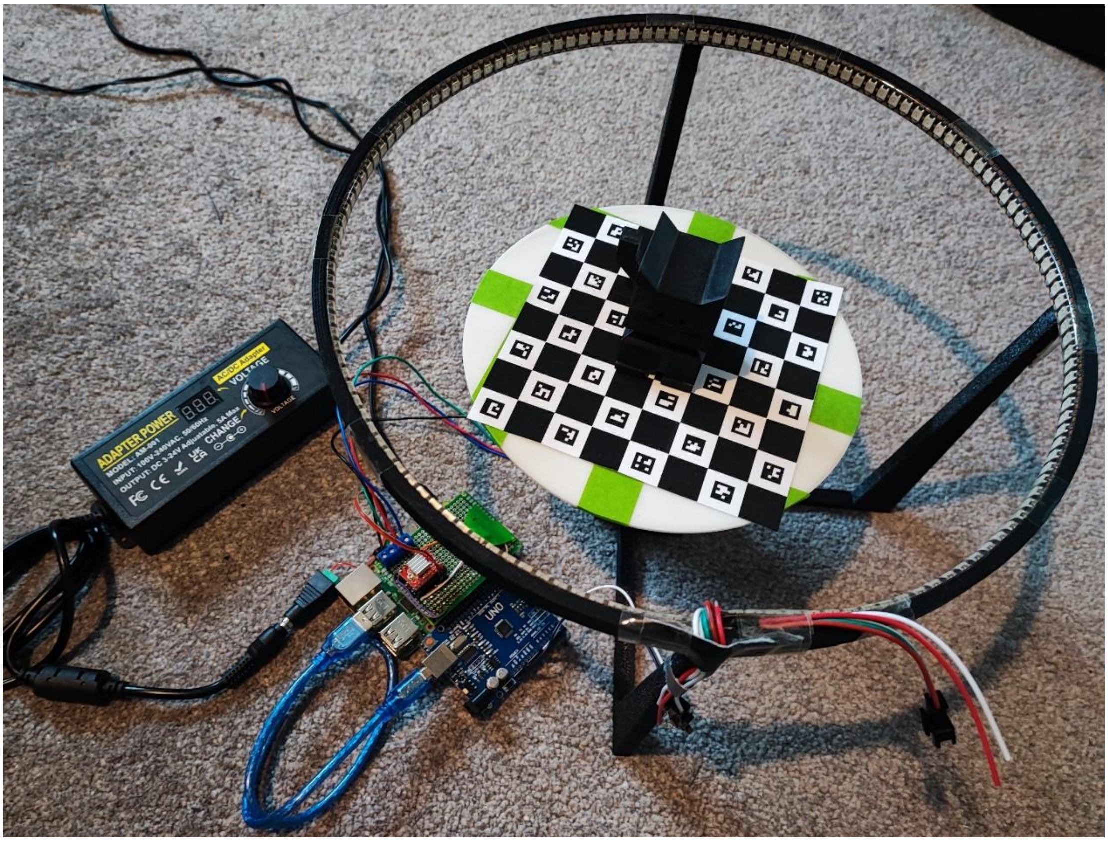


*Figure 7 Current Prototype*

## Hardware

### Structural Elements

Three separate parts were 3D printed using PLA plastic as parts of the light stage or as mounting points for other hardware. All parts were designed using Autodesk Fusion 360 CAD Software[24].

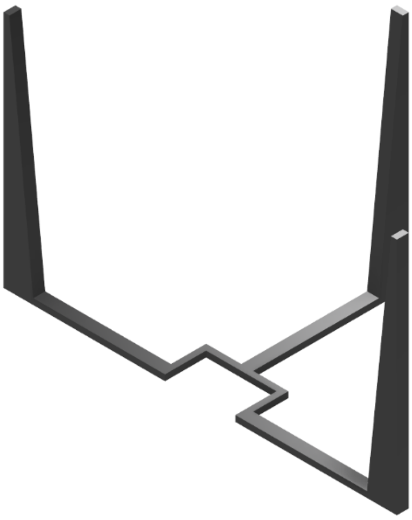

*Figure 8 - LED Halo Stand*

First part is the LED halo stand which can be seen in figure 9, serves two main purposes. One is to hold the LED Halo so that it's at a level which can directly illuminate the object on the turntable surface and the other is to align the LED Halo using the central half square cut-out in the middle of the part which goes around the stepper motor ensuring the turntable surface is at the central position of the LED Halo.

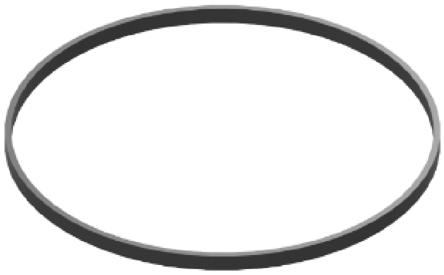

*Figure 9 - LED Halo*

*Figure 10 - Turntable Surface*

Second part is the LED halo which holds the LED Strip that houses the 144 individually addressable LEDs used as point lights to illuminate the object being captured. The LED Halo is a circle with a diameter of 32cm.

Third part is the turntable surface which is mounted on the stepper motor and is the surface the target object sits on. A ChArUco marker is pasted onto it for camera calibration.

## Electronics

An electronics system has been built to have a great amount of control over all aspects of the light stage. The circuit diagram for this is given in figure 12. A Raspberry pi acts as the main controller of the system controlling to a A4988 Motor Driver which controls a NEMA 23 23HS9430 Stepper Motor. The Raspberry pi also controls an Arduino Uno through a USB Serial connection which then controls an individually addressable LED Strip with 144 individual LEDs.

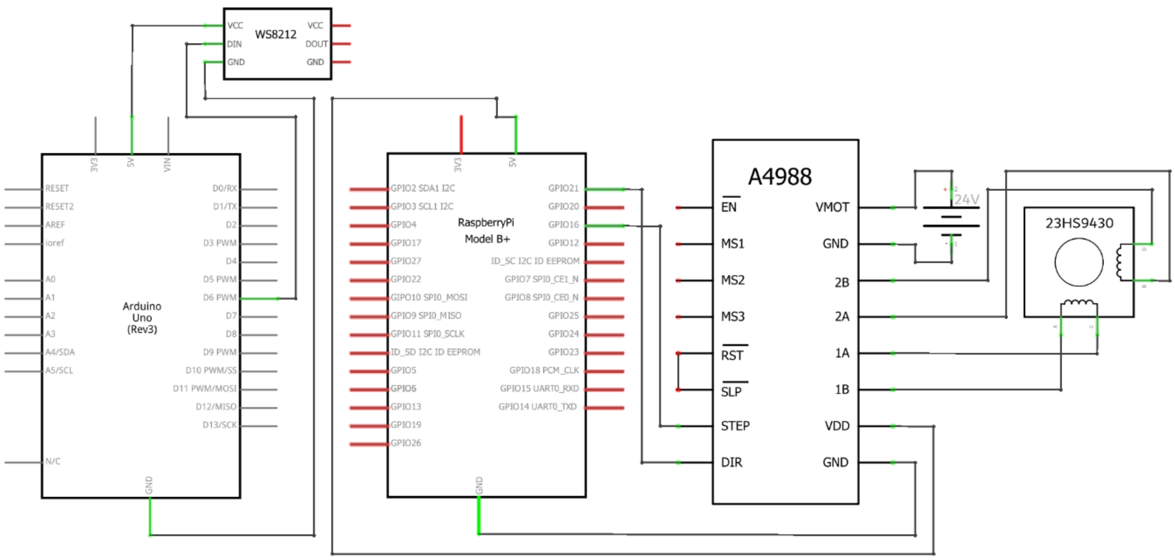


*Figure 11 – Circuit Diagram for the Light Stage (Arduino and the Raspberry Pi are connected through USB)*

## Software

A software and a firmware system has been developed in order to control the light stage. Three Different low-level APIs were written in C++ to allow for control over the stepper motor, the serial data transfer between the Arduino and the Raspberry Pi and the LED Strip. Currently no configurable user interface exists so all configurations must be manually hardcoded. To communicate with the DSLR Camera libgphoto2 is used, a C library for accessing digital cameras[25]. This allows for complete control over which LEDs to light, how much to rotate the turntable and when and when to capture an image. This has allowed the light stage prototype to capture multiple datasets with success and some without success which has allowed for the discovery of minor bugs in the software that has been fixed.

*Figure 12 - 36 Example Captures from a dataset showing an object lit from 36 different directions*

## Planned Deliverables

To be able to achieve the aim and objectives of this project a set of deliverables should be completed. These are categorised as hardware, software and documentation and are as follows:

### Hardware

A compact light stage system that can illuminate and capture and object from multiple directions.

The system should:

- Reproduce consistent captures and in case of small inconsistencies these should be fixed thoroughly software.
- Be compact, mobile and if necessary modular.
- Relatively affordable.

### Software

One singular or multiple software packages/scripts which should be able to work with the dataset provided by the hardware capture system.

The software should be able to:

- Fully control the hardware system to the requirements of the user within its capabilities.
- Calibrate the Camera
- Generate object models.
- Generate materials that allow for the object to be relit.

### Documentation

Both the software and hardware should be documented enough so that it allows the user to fully operate the systems, or the system should have a user interface that allows the user to understand how to operate it. All work and testing should be recorded for it to be reproducible by others.

### Current Status

Hardware system is currently fully operational and meets all the requirements for it to satisfy the project objectives.

Software to operate the hardware is also currently operation but cannot control the light stage in a way that would allow it to utilise all hardware functionalities for certain 3D Reconstruction techniques. Currently a user interface for easy configuration is missing.

There is also no software for 3D reconstruction or camera calibration but according to the project plan this part of the project is scheduled for semester 2.

# Future Work

## Hardware

Hardware prototype for the project is mostly complete and is fully functional and will not receive any further upgrades unless allowed in the case where there is excess time available. There are still many optimisations available such as multiple LED Halos which house more LED Strips allowing for lighting from more angles which should allow for the capture of data set that allows better reconstruction and relighting of the 3D object.

## Software

Software will be focused on in the second semester of the academic year. Hardware control software is mostly complete but still requires some API implementations to allow the user full control over the hardware for it to be able to fulfil captures for specific techniques. A user interface might also be implemented if given enough time for easy configuration.

Software to process captures is still in the planning stages and will be built next. 3D Reconstruction techniques will be implemented into this software to allow for object modelling and relighting from the captures provided by the hardware.

# Project Plan

## Overview

A basic Gantt chart and a more detailed Kanban board was made to make sure project progresses accordingly and to be able to track all completed tasks. Both the Gantt chart and the work Kanban plan is updated and expanded as the project progresses. A second Gantt chart for Semester two will be constructed at the beginning of the semester to outline the time allocation for each of the tasks remaining. This was not made before hand for it to be more easily adapted to the state of the project at the beginning of semester two.

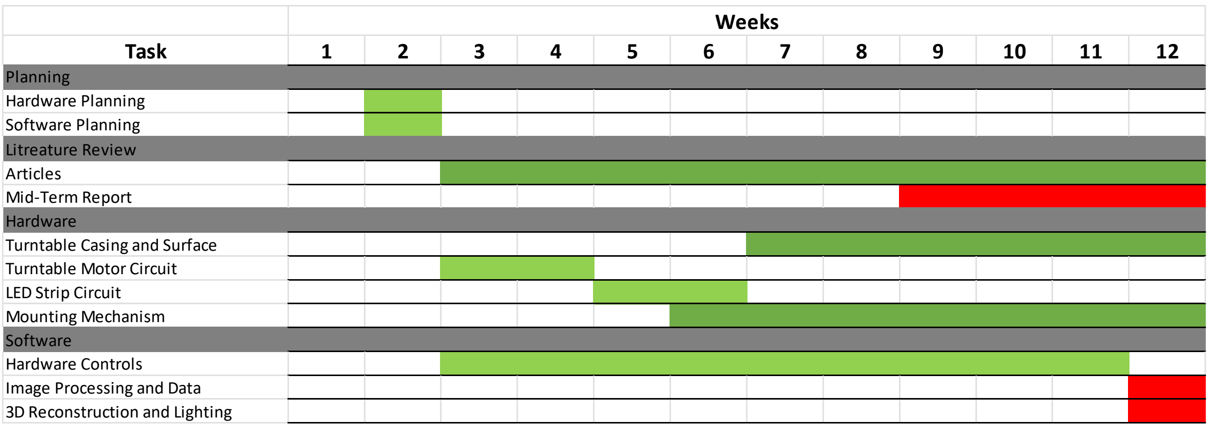


*Figure 13 - Project Gantt Chart*

| Backlog | To-Do | In Progress | Done |
|---|---|---|---|
| Game Engine Showcase<br>Process: 0% | Image Processing Code<br>Process: 0% | Camera Calibration Code<br>Process: 0% | Stepper Motor Circuit for Turntable<br>Process: 100% |
| Hemisphere Lighting Setup<br>Process: 0% | Data Management Code<br>Process: 0% | | LED Controller Circuit<br>Process: 100% |
| Multi-Camera Setup<br>Process: 0% | 3D Reconstruction and Lighting Code<br>Process: 0% | | RPI - Arduino Serial Connection Code<br>Process: 100% |
| 3D Scanning optimisation<br>Process: 0% | | | RPI Stepper Motor Controller Code<br>Process: 100% |
| | | | RPI Led Controller Code<br>Process: 100% |
| | | | Arduino LED Controller Code<br>Process: 100% |
| | | | Literature Review<br>Process: 100% |
| | | | LED Strip Mounting Setup<br>Process: 100% |
| | | | Camera Controller Circuit<br>Process: 100% |
| | | | Camera Controller Code<br>Process: 100% |

*Figure 14 - Project Kanban Board*

## Risk Analysis

As the hardware prototype for the project is complete there aren't many risks left which might be encountered. A Risk assessments matrix was made in order to identify potential risks and how likely they are to occur. This way it possible to foresee any issues that may arise and prepare precautions beforehand.

| | | Severity | | | |
|---|---|---|---|---|---|
| | | **Catastrophic** | **Critical** | **Marginal** | **Negligible** |
| **Probability** | **Frequent** | No Risk | No Risk | Not able do captures often enough do not having access to the DSLR Camera constantly | Software and Hardware Bugs |
| | **Probable** | No Risk | No Risk | No Risk | No Risk |
| | **Remote** | No Risk | Limitation on how much data can be acquired due to long capture times | No Risk | No Risk |
| | **Improbable** | Complete failure of both hardware and software | No Risk | No Risk | No Risk |

*Figure 15 – Risk Assessment Matrix*

# Estimated Cost

Estimated cost of all the different parts of the current prototype. Many of the parts were already available at zero cost from the Electronic Engineering Undergraduate Labs.

| Part | Cost | Source |
|---|---|---|
| Raspberry Pi 3B+ | £40 | EEUG labs (Makerspace) |
| Arduino Uno | £22.50 | EEUG labs (Makerspace) |
| A4983 Stepper Motor Driver | £1.85 | EEUG labs (Makerspace) |
| WS2812B LED Strip | £14 | Already Available |
| NEMA Stepper Motor | £25.47 | EEUG labs (Makerspace) |
| 3D Printed Parts (PLA Plastic) | ~£6.70 | Already Available |

*Figure 16 – Table of components*

Total cost of the project currently comes out to £110.52 which is over the given budget by £10.52 but as these parts were sourced through means which did not require any of the budget to be spent. The camera was not included in the project cost as it is loaned from the supervising academic and not a permanent part of the project.

## Final Report Structure

A final report structure based on this report has been produced. Including mostly the same headings with some minor changes. This might be adjusted and improved as needed during the writing of the final report. Below is an outline of the final report structure

- Title Page
- Abstract
- Introduction
- Literature Review
    - Further review on different techniques
- Development and Experiments
    - Further subheadings explaining development and experiments
- Results and Analysis
    - What was achieved
    - Comparison of different Results
- Showcase
    - Showcase of the results
- Summary
- Conclusion
- Reflection and Future Outlook
- Acknowledgements
- References
- Appendices

## Summary

This report introduced the controllable light stage project with it's aims and objects and a short history of light stage development.

A literature review was produced which goes over techniques which could be used to process the dataset captured using the light stage prototype and how different techniques work and their results are discussed.

Current development progress of the controllable light stage is showcased with explanations of the current design and how it works.

Planned deliverables are discussed which outline what is required of this project for it to be completed and any required future work is discussed.

Project Planning is showcased with a Gantt chart and a Kanban board. A risk assessment is conducted to identify potential risks.

Estimated cost is calculated with a table of components which list their price and how the components were acquired.

A draft of the final report structure is explained with planned headings for the final report given.

## Conclusion

In conclusion a working controllable light stage prototype has been built which can successfully capture datasets. This work was documented and explained. Existing Literature for how to process the capture datasets were discussed and planned work was also documented. The project is on track to be completed in the given timeframe with further work and software implementations to process the capture datasets to be implemented in the second semester of the academic year.

## Acknowledgments

This project and report would not have been possible if not for the supervising academic Dr Jean-Yves Guillemaut. Further acknowledgments should be given to Paul Debevec for great contributions in the creation of light stages and improving the technology for it to be possible to replicate and improve upon to this extent and finally The University of Surrey should be acknowledged for making this project possible.

## References


[1] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the Reflectance Field of a Human Face," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, 2000, pp. 145–156. doi: 10.1145/344779.344855.

[2] University of Southern California Institute for Creative Technologies., "The Light Stages at UC Berkeley and USC ICT." https://vgl.ict.usc.edu/LightStages/ (accessed Jan. 01, 2023).

[3] O. Alexander *et al.*, "The Digital Emily Project: Achieving a Photorealistic Digital Actor," *IEEE Comput Graph Appl*, vol. 30, no. 4, pp. 20–31, Jul. 2010, doi: 10.1109/MCG.2010.65.

[4] Academy of Motion Pictures Arts and Sciences., "82nd Academy Awards." https://www.oscars.org/oscars/ceremonies/2010 (accessed Jan. 01, 2023).

[5] P. E. Debevec, "The Light Stages and Their Applications to Photoreal Digital Actors," in *International Conference on Computer Graphics and Interactive Techniques*, 2012.

[6] University of Southern California Stevens Center for Innovation, "Light Stage - Success Stories - Stevens center for Innovation." https://stevens.usc.edu/success-stories/light-stage/ (accessed Jan. 01, 2023).

[7] Computer History Museum, "The Utah Teapot." https://www.computerhistory.org/revolution/computer-graphics-music-and-art/15/206 (accessed Jan. 01, 2023).

[8] A. Fetić, D. Jurić, and D. Osmanković, "The procedure of a camera calibration using Camera Calibration Toolbox for MATLAB," in *2012 Proceedings of the 35th International Convention MIPRO*, 2012, pp. 1752–1757.

[9] Mathworks, "Camera Calibration." https://uk.mathworks.com/help/vision/ug/camera-calibration.html (accessed Dec. 31, 2022).

[10] OpenCV, "Detection of ChArUco Boards." https://docs.opencv.org/4.x/df/d4a/tutorial_charuco_detection.html (accessed Dec. 31, 2022).

[11] OpenCV, "Calibration with ArUco and ChArUco." https://docs.opencv.org/3.4/da/d13/tutorial_aruco_calibration.html (accessed Dec. 31, 2022).

[12] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans Pattern Anal Mach Intell*, vol. 22, no. 11, pp. 1330–1334, 2000, doi: 10.1109/34.888718.

[13] R. Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," in *Computer Vision and Pattern Recognition*, 1986.

[14] R. J. Woodham, "Photometric Method For Determining Surface Orientation From Multiple Images," *Optical Engineering*, vol. 19, no. 1, Feb. 1980, doi: 10.1117/12.7972479.

[15] NewProggie, "Photometric-Stereo." https://github.com/NewProggie/Photometric-Stereo (accessed Jan. 03, 2023).

[16] C. Hernandez Esteban, G. Vogiatzis, and R. Cipolla, “Multiview Photometric Stereo,” *IEEE Trans Pattern Anal Mach Intell*, vol. 30, no. 3, pp. 548–554, Mar. 2008, doi: 10.1109/TPAMI.2007.70820.

[17] K. Ikeuchi, “Determining Surface Orientations of Specular Surfaces by Using the Photometric Stereo Method,” *IEEE Trans Pattern Anal Mach Intell*, vol. PAMI-3, no. 6, pp. 661–669, Nov. 1981, doi: 10.1109/TPAMI.1981.4767167.

[18] D. Schnieders and K.-Y. K. Wong, “Camera and light calibration from reflections on a sphere,” *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1536–1547, Oct. 2013, doi: 10.1016/j.cviu.2013.06.004.

[19] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis,” Mar. 2020.

[20] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3D surface construction algorithm,” *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 163–169, Aug. 1987, doi: 10.1145/37402.37422.

[21] ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng, “NeRF.” https://www.matthewtancik.com/nerf (accessed Jan. 03, 2023).

[22] P. Hedman, P. P. Srinivasan, B. Mildenhall, J. T. Barron, and P. Debevec, “Baking Neural Radiance Fields for Real-Time View Synthesis,” Mar. 2021.

[23] P. P. Srinivasan, B. Deng, X. Zhang, M. Tancik, B. Mildenhall, and J. T. Barron, “NeRV: Neural Reflectance and Visibility Fields for Relighting and View Synthesis,” Dec. 2020.

[24] Autodesk, “Overview - Fusion 360.” www.autodesk.com/products/fusion-360/overview (accessed Dec. 31, 2022).

[25] gPhoto, “Projects::libgphoto2.” http://www.gphoto.org/proj/libgphoto2/ (accessed Jan. 01, 2023).