

Virtual Instrument for a Multi-illumination Dome System

Mojtaba A. Khanesar¹, Hannah Corcoran², Lindsay MacDonald², Vijay Pawar², Stuart Robson², Samanta Piano¹

¹ Faculty of Engineering, University of Nottingham, NG8 1BB, Nottingham, UK, (ezzma5.ppzsp1)@exmail.nottingham.ac.uk

² Faculty of Engineering Science, University College London, WC1E 6BT, UK

Keywords: Multi-illumination dome system, Virtual Instrument, 3D digital object, Blender, Camera parameters, Photometric.

Abstract

A virtual instrument has been developed to study a multi-illumination dome system on digital objects using a camera mounted on top of the dome. The reliability of the blender environment is investigated for simulating the functions of components of this system. The high-precision three-dimensional (3D) mesh of an object measured with a laser tracker equipped with an AS1 scanner is used as a digital object within this virtual instrument. The camera is modelled by both intrinsic and extrinsic parameters and the virtual lights by power, beam angle and diffusion. The parameters are adjusted to match the real camera and light sources. The surface normals estimated using images taken of the 3D digital object using different illumination conditions are compared against actual surface normals for three artefacts with fully determined surface normals. Results show that the three sets of surface normals match each other and the mean value of the angles between the sets of surface normals for least squares with a threshold approach is limited to a few degrees for the case of objects of a few centimeters in size. Inspired by the results of surface normal estimation for the three digital artefacts, the same algorithms are then applied to a digital object scanned using a laser scanner.

1. Introduction

Reflectance transformation imaging (RTI) (Grzelec and Łojewski, 2023) is a technique for capturing and rendering enhanced details of an object's surface. This imaging method is often used for the visualisation of cultural heritage objects such as archaeological artefacts and ancient graffiti (Sammons, 2018). This method can also be used to relight archaeological artefacts for new lighting direction (Bosco et al., 2023). This paper provides a virtual environment to simulate an RTI environment. Simulating the setup using a virtual instrument enables the system to be optimised, ensuring that the surface normals calculated are as precise and accurate as possible by identifying factors that could potentially lead to systematic errors. The virtual instrument is developed in the Blender environment (Cartucho et al., 2021), incorporating light sources and camera models (Pottier et al., 2023a).

To test machine learning algorithms and perform optimisation for the real RTI system, its virtual equivalent is used. The virtual RTI environment investigated in this paper consists of an illumination dome and a single camera mounted at the top of the dome looking downwards at the object. Multiple light sources are mounted on the inner surface of the dome to provide directional lighting of the object. This configuration enables multiple images to be taken of the three-dimensional (3D) object with different illumination geometries.

The virtual photometric system is implemented by using Blender (Flavell, 2011). The advantages of selecting Blender over other simulation software to construct the virtual instrument are as follows:

1. Comprehensive and versatile: this software benefits from a wide range of optical components and 3D design capabilities. Furthermore, the 3D mesh provided by 3D scanning tools can easily be imported into this environment (Hosen et al., 2019).

2. Free and open-source: the software is publicly available and is supported by a community that continuously adds objects and features to facilitate its usage (Flavell, 2011).
3. Physics simulation capability: physical entities such as light, camera, and bidirectional reflectance distribution function can be simulated within this environment (Gschwandtner et al., 2011).
4. Integration with other software: Blender benefits from the Python API that allows this environment to import Python scripts and functionality within a Blender scene (Kent, 2015).

It has recently been shown (Pottier et al., 2023b, Eastwood et al., 2019) that Blender can be used reliably to implement a virtual instrument to perform multi-camera metrology. To investigate the capability of Blender for metrology applications, they have investigated metrology error within this virtual environment.

This paper investigates the capabilities of Blender to build a virtual RTI instrument. It is shown that images generated within this environment can be used instead of actual images taken from 3D objects in the real world for testing machine learning approaches and performing optimisation. The digital object to be studied in this virtual environment can be designed directly by CAD software or imported as the 3D digital scan of a real object. The latter relies on a 3D point cloud of the real object, obtained using an accurate laser tracker equipped with a handheld scanner, converted to a mesh format after resampling and surface reconstruction. Parameters for the camera's intrinsic and extrinsic characteristics and the power, beam angle, and diffusion of each light source are defined using tools in Blender, adjusted to match the physical imaging configuration. Multiple images are generated within the Blender virtual environment by turning on and off the light sources within the dome. The images taken from the digital object are then used to compute surface normals for comparison with the surface normals of the digital object. The results demonstrate the high performance

capability of Blender in simulating a virtual RTI environment. Images taken from the scanned 3D mesh within the Blender environment are visually compared with the images taken in the actual dome system as well.

The remainder of this paper is organised as follows. The methodology is presented in Section 2, results are reported in Section 3, and final remarks are presented in Section 4, followed by acknowledgments and references.

2. Methodology

To investigate the capability of Blender for 3D surface normal detection, the following steps are designed:

1. Generate the digital object using a CAD model or 3D mesh scanned by a laser tracker equipped with the appropriate scanner namely AS1 scanner (see Fig. 4).
2. Preprocess 3D data, including downsampling, creating meshes, and perform formatting conversion requirements.
3. Add reference targets to the scene (see Fig. 2). These reference objects are crucial for camera characterisation purposes.
4. Insert digital object into the Blender environment along with photometric elements, including camera and lighting objects. Adjust camera data including intrinsic and extrinsic parameters and light data to match the real environment.
5. Take images of the object using multiple illumination conditions. The light sources are turned on and off consecutively to take images.
6. Photometric methods are used to generate surface normals from the images. Mask the background for clarity.
7. Compare the surface normal results for the images taken in Blender with those computed from the real images. Analyse the distribution of surface normal angle errors between the two sets of corresponding surface normals.

Parameters of the virtual instrument developed in Blender refer to the camera, light, and object. For the camera these include intrinsic and extrinsic parameters, noise level, and filtering. The light source is defined by position, direction, diffusion, and colour. A transformation matrix is employed to define both the position and orientation of the object.

2.1 Experimental Setup

The experimental setup (Fig. 1) consists of:

1. The hemispherical dome structure, assembled from circular wire hoops, on which the LEDs are mounted.
2. 48 LED light sources to provide illumination to take images within the dome imaging system.
3. Solid state relays controlled by an Arduino to select and switch LEDs on and off.
4. Camera including connectivity to the PC to capture images. A trigger signal from the Arduino synchronises the camera to each light source on the dome.

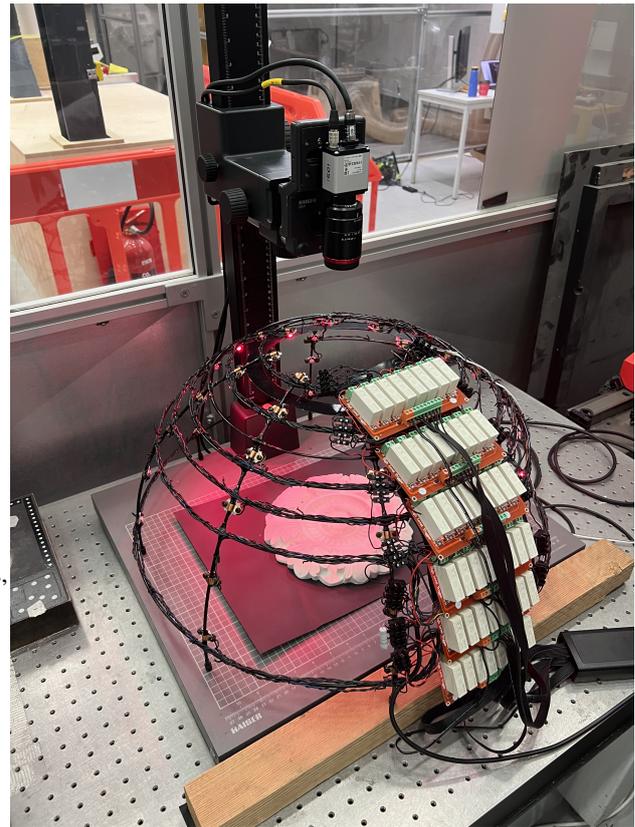


Figure 1. Experimental dome with 48 LEDs

5. Copy stand mounting of the camera, enabling its height to be adjusted.
6. Plaster test object to be studied within the dome system.

The overall image capture sequence begins with the command given by Matlab to the Arduino controller, selecting which relay to turn on to illuminate the required LED. A trigger signal is given to the camera to take an image of the object as soon as the LED has been turned on. The shutter speed for the camera is adjusted initially and then kept constant so that all images within the sequence are within the exposure range. Typical shutter speed is 50 msec.

2.2 Virtual Instrument in Blender

The virtual instrument implemented within the Blender environment includes a 3D digital object to be studied, a virtual camera to take images, and multiple virtual light sources that are illuminated consecutively to take images of the virtual object.

2.2.1 Digital Object through 3D Scanner To measure an object using the virtual instrument, it is required to import its digitised version within the virtual environment. For this purpose, a decorative plaster ceiling rose was placed in a mounting plate, as shown in Fig. 2. Small white spheres with diameters of $20\text{mm} \pm 5\mu\text{m}$ were added to the scene to provide photogrammetric references for the measurement system. The 3D data was acquired with the Hexagon Absolute Scanner AS1 as shown in Fig. 3, connected to an AT960 laser tracker. The software, SpatialAnalyzer, was used to collect the data enabling the point cloud to be viewed during the scanning process. SpatialAnalyzer can also be used to locate the instrument from known

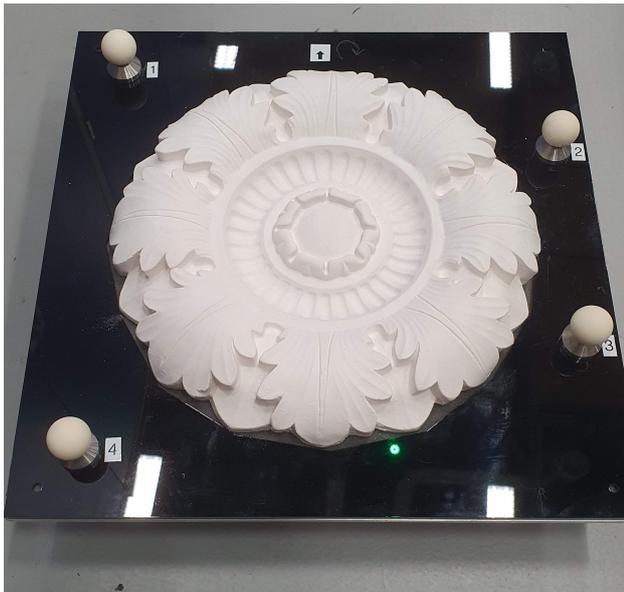


Figure 2. Cultural heritage test object to be studied.



Figure 3. Hexagon Absolute Scanner AS1

points if the registration of data within a particular coordinate system is required.

The point cloud of the object, processed with SpatialAnalyzer, is shown in Fig. 4. It includes more than 33 million points, encoded in the PTX file format. To insert the digital object into the Blender environment, its 3D mesh needs to be obtained. The screen Poisson surface reconstruction technique is used to create a 3D mesh out of an orientated point cloud (Kazhdan and Hoppe, 2013). Multiple complexity reducing algorithms within Meshlab software (Gabara and Sawicki, 2017) perform efficient and high quality surface reconstruction. The initial point cloud is down-sampled to one million points, then a screened Poisson reconstruction algorithm is applied. The resulting mesh can then be imported into the Blender environment in STL format.

2.2.2 Down-sampling and Surface Reconstruction Poisson-disk sampling is one of the point-cloud sampling approaches frequently used within computer graphics (Corsini et al., 2012). The aim is to select uniformly distributed points out of the point cloud such that the minimum distance between the points is larger than predefined $2r$ value. The method is known to have good *BLUE characteristics*, meaning that it avoids concentration of points in a region and maintains a consistent distance between points. Different variations of Poisson disk sampling exist in the literature. An efficient and flexible approach has been previously investigated in (Corsini et al., 2012) and is

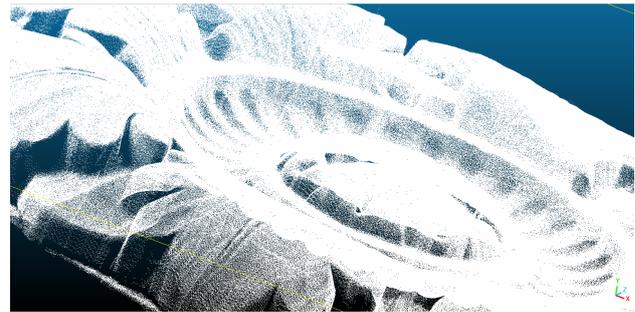


Figure 4. Point cloud obtained from the test object

implemented within an open-source point-cloud manipulation software, namely Meshlab (Cignoni et al., 2011). This is the preferred algorithm in this paper and it is modified to generate better results in terms of sampling distribution.

2.2.3 Digital Object CAD Model To assess the efficacy of the virtual instrument in identifying surface normals via Reflectance Transformation Imaging (RTI), various digital objects were examined. The surface normals of these objects are entirely known, allowing for a comprehensive evaluation of both the virtual instrument and the surface normal detection algorithm. The study focuses on analysis of three artefacts depicted in Figures 5, 6, and 7. Having the precise geometrical data of the three artefacts through their CAD model, the precise surface normal vectors associated with the two CAD models are calculated. A colouring convention is used to demonstrate the surface normals of a 3D object in 2D as follows.

$$\begin{aligned} R &= 255/2.(N_x + 1) \\ G &= 255/2.(N_y + 1) \\ B &= 255.N_z \end{aligned} \quad (1)$$

where $N_x \in [-1, 1]$, $N_y \in [-1, 1]$, and $N_z \in [0, 1]$ are the x , y , and z components of the surface normal and $R \in [0, 255]$, $G \in [0, 255]$, and $B \in [0, 255]$ are the corresponding color component of the pixel.

Using calculated values of the surface normals for the three given artefacts, it is possible to evaluate the performance of the RTI system within the virtual instrument.

2.2.4 Mathematical Model of Camera The mathematical representation of a camera is based on the notion of the camera functioning as a projection system from the 3D real world to a 2D image plane. This concept provides the mathematical analysis employed for distance measurement, surface textures, and dimensions using optical devices within a metrology application (Bottalico et al., 2023, Sturm, 2014). The overall projection process from the real world into a 2D image is formulated using camera parameters as follows:

$$s \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & \theta & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}}_{\text{intrinsic matrix}} \underbrace{T_w}_{\text{projection matrix}} \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix} \quad (2)$$

$$T_w = \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} & T_x \\ R_{yx} & R_{yy} & R_{yz} & T_y \\ R_{zx} & R_{zy} & R_{zz} & T_z \end{bmatrix}_w \quad (3)$$

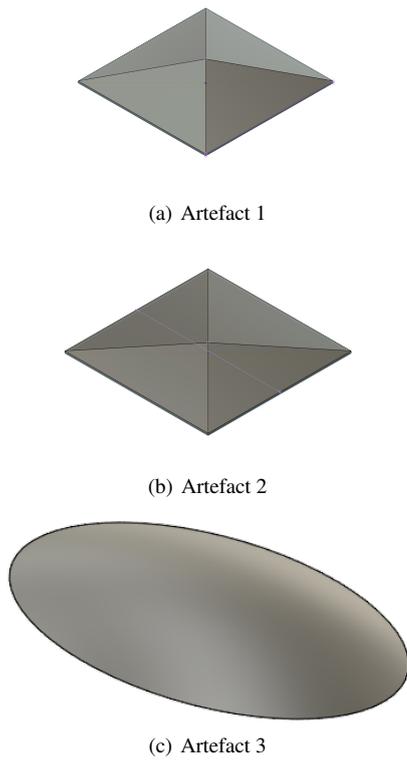


Figure 5. Two pyramid and spherical artefacts designed by CAD software with known surface normals.

where (p_x, p_y, p_z) are the position coordinates of the point in real world, (p_u, p_v) are the corresponding coordinates of the point in the image in pixels, f_x and f_y are the camera intrinsic parameters such that $f_x = FS_x$, and $f_y = FS_y$, and S_x ($pixel / mm$), and S_y ($pixel / mm$) are the sensor's scale factors in x and y coordinates. The parameter F is the focal length in world units, typically expressed in millimetres. The parameter θ is the skew parameter. The parameter s is a scaling factor, which may be different than 1 (Hartley and Zisserman, 2003).

The parameters $R_{ij}, i, j = x, y, z$ define the rotation matrix and $T_i, i = x, y, z$ the translation matrix. The radial distortion can be modelled as a polynomial function of the distortion radius (Enebuse et al., 2021, Khanesar et al., n.d.):

$$x_u = p_u (1 + k_0 r_d^2 + k_1 r_d^4 + k_2 r_d^6 + k_3 r_d^8 + \dots)$$

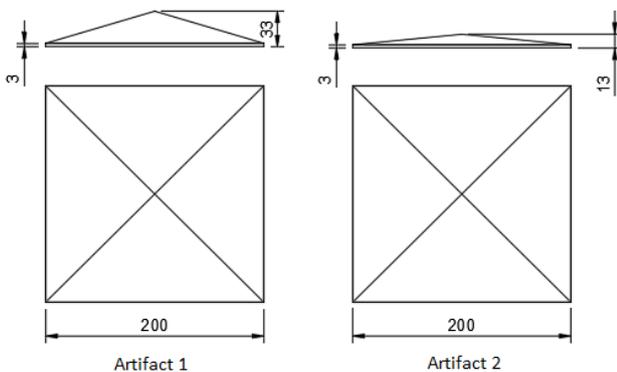


Figure 6. Dimensions of the two pyramid artefacts (all values are given in mm).

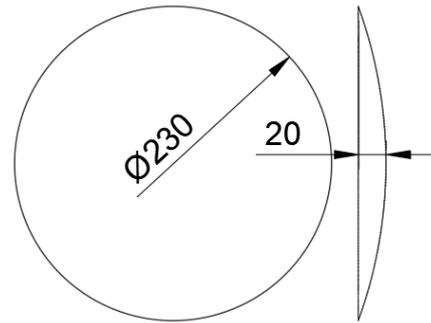


Figure 7. Dimensions of the spherical artefact (all values are given in mm).

$$y_u = p_v (1 + k_0 r_d^2 + k_1 r_d^4 + k_2 r_d^6 + k_3 r_d^8 + \dots) \quad (4)$$

where (x_u, y_u) are the coordinates of points in the image as distorted by the function given in (4).

2.2.5 Virtual Camera in Blender The camera is simulated in the Blender environment using a Python script to define the parameters. The focal length is set to $25mm$ to match the lens of the real camera. Likewise, the distortion parameters for the virtual camera match those of the real camera as a result of calibration procedure using Agisoft, as follows.

$$\begin{aligned} k_0 &= -0.0490, k_1 = -0.5679, \\ k_2 &= 1.6264, k_3 = 0 \end{aligned} \quad (5)$$

In the Blender environment, the dimension of the sensor is another essential factor when specifying a camera, because it influences the magnification, i.e. the real-world scale that corresponds to the pixels in the image. The sensor size for the virtual camera considered in this paper is $12.363mm \times 12.363mm$. For a complete list of camera parameters within the Blender environment, readers may refer to this URL: <https://docs.blender.org/api/current/bpy.types.Camera.html> (visited: 25/09/2024).

To calibrate the camera within the dome environment, images were taken of the artefact within the dome environment. Images were captured of the artefact in multiple positions on the plane, and at slight inclinations for high-performance calibration. Agisoft photogrammetry software was applied to the images taken within the dome and intrinsic and extrinsic parameters of the camera identified. The calibration data is exported in XML format from Agisoft software which are then parsed within Blender environment. These values were used within the virtual environment to set the parameters of the virtual camera as well as its position with respect to the centre of the image.

2.2.6 Virtual Light Virtual illumination is incorporated within a Blender scene to simulate various lighting conditions within this setting. Four options are available: point, sun, spot, and area. The point light source used in this paper benefits from features such as power, diffusion factor, colour, location, and transformation matrix. The power attribute designates its illumination strength measured in watts. The colour value defines the light colour in RGB format, while each of the other values is a real value from the interval $[0, 1]$. Finally, the transformation matrix of the light object, including its position and orientation, is assigned using the corresponding commands. In total, there

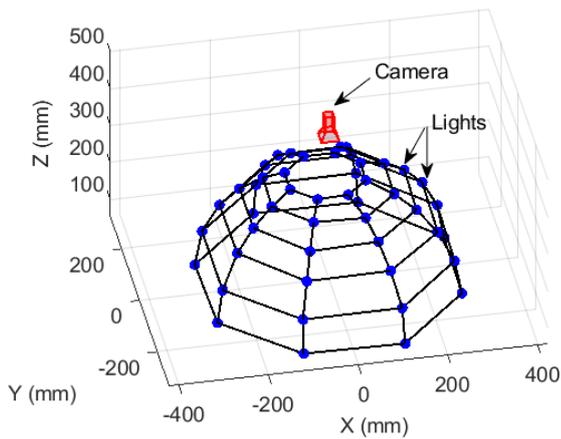


Figure 8. Dome illumination system using 48 light sources.

exist 48 light sources with the dome imaging system, and their positions are depicted as in Fig. 8.

Within the Blender environment, there exist multiple adjustable parameters for the light source. The diffusion and blend parameters are found to have a more crucial impact on the scene. If these parameters are close to zero, a sharp border between the shadow and bright regions is observed. However, if the value of these two parameters is increased, the border between the shadow and bright regions will be blurred. To make a realistic model of the light source, an experiment is designed in which the light source is turned on and off. The difference between the two images shows the influence of the light source on the object. The intensity values of the light source usually decrease from the central light ray. Intensity values can be drawn as a function of the angle of the light beam with respect to the direction of the surface normal. It is therefore possible to have a wider angle of measurement for a wider camera lens. The same experiment is repeated in the Blender environment to find the intensity diagram of the light source versus the angle of the beam with the surface normal. The experiment is repeated for a few 'diffusion' and 'blend' values. These parameters determine the distribution of the light intensities far from the centre of the light cone. The spot size parameter of the light source adjusts the overall beam angle of the light source. Throughout the Blender experiments, this parameter is set to 100° for both sides of the light source. The light source blend parameter is set to 1, and the diffusion parameter changes from 0 to 1. From the set of trials, it is concluded that the closest matched LED model to the manufacturers specifications occurs when the diffusion parameter is equal to 1. Therefore, this parameter is used within the virtual instrument to take photos of the object.

3. Photometric Approaches Used to Detect Surface Normals

3.1 Simple least squares solution

The Lambertian model for detecting surface normals assumes that the intensities within an image information model in a Lambertian scene depend on the angle between the light beam hitting the corresponding pixel and the surface normal and the diffuse albedo parameter (Woodham, 1979). Therefore, intensities in the image taken within this scene are expressed in terms of the dot product of the incoming light beam vector and surface

normal multiplied by the diffuse albedo parameter.

$$I = \rho n^T l, \quad (6)$$

where I is the intensity value of the corresponding point, $\rho \in R$ is the diffuse albedo, and $n \in R^3$ is the 3D vector corresponding to surface normal, $l \in R^3$ is the 3D vector corresponding to incoming beam angle. For n images taken with different light angles for each point and m number of intensity values within the image, it is possible to rewrite the image information model in matrix form as follows.

$$O = N^T L, \quad (7)$$

where $N = [\rho_1 n_1, \dots, \rho_m n_m] \in R^{m \times 3}$ is the overall surface normals in the scene, $L = [l_1, \dots, l_m] \in R^{m \times 3}$ is the overall light directions for each point in the image. A simple solution to (7) is the least squares solution to the problem (Wu et al., 2011).

$$N = O L^\dagger, \quad (8)$$

3.2 Least squares solution with Threshold Values

Shadow points and specularities inevitably exist in a scene, making the regression problem of (7) as follows.

$$O = N^T L + E, \quad (9)$$

where $E \in R^{m \times 3}$ represent uncertainties due to specularities and shadow in the image. Although shadow points are relatively easily detectable through a threshold value, specularities are more difficult to detect (Wu et al., 2011). More advanced estimation algorithms are used to solve the regression problem of (9).

The four different algorithms to detect shadows in images are: classification, segmentation, thresholding, and geometric modeling (Dare, 2005). Thresholding is a binary comparison between pixel values and a constant threshold value to identify the undesired features in the image. An appropriately selected threshold value can lead to avoiding a large number of miss classified pixels in the image.

4. Results

Figure 8 illustrates the multi-illumination dome system using 48 light sources, distributed evenly across the hemisphere to provide different lighting directions for the 3D object to be studied. The results presented in this paper include two main parts. In the first part, three virtual artefacts with slightly different surface normals are studied. In the second part, a physical moulded plaster object with a near-Lambertian surface is imaged enabling comparison between physical and virtual artefacts.

4.1 CAD model artefact

The three digital artefacts investigated in this paper (Figures 5, and 6) are two square pyramids with four planar surfaces and a spherical object (Figures 5, and 7), each covering a large area. The least-squares approach is applied to the registered pixel intensity values of the 48 images taken within the virtual dome environment to estimate the surface normal vectors associated with the object. Differences between the reference and measured surface normals were calculated and the statistics for these

Table 1. Surface normal angular errors from pyramidal artefact 1 (height=30mm).

Method	Error (degrees)			
	whole artefact		1cm disk	
	MAE	MSE	MAE	MSE
Least squares	16.59	20.94	1.87	2.16
Least squares with threshold	15.52	17.96	1.87	2.16

Table 2. Surface normal angular errors from pyramidal artefact 2 (height=10mm).

Method	Error (degrees)			
	whole artefact		1cm disk	
	MAE	MSE	MAE	MSE
Least squares	17.36	21.87	2.78	3.59
Least squares with threshold	16.32	19.07	2.78	3.59

results are given in Tables 1, 2, and 3 in terms of mean absolute error (MAE), and mean squared error (MSE) values.

Two different algorithms to estimate surface normals are investigated. A basic least-squares approach where all pixel intensity observations are considered similarly weighted, and a threshold process whereby pixels corresponding to shadow regions are rejected prior to the least-squares process. In both cases, Lambertian reflectance of the test surface is assumed. Using a minimum threshold value within the least-squares algorithm can filter out pixels that are in shadow. For example, given a low illumination angle emanating from one corner of the pyramidal artefact, the furthest two facets from the light source will be in shadow. As demonstrated in Tables 1, 2, and 3 the error value associated with the entirety of the object exhibits a larger value than those for a smaller area. Furthermore, the least squares algorithm augmented with a threshold value outperforms the standard least squares algorithm in the case of the two pyramid artefacts, as it removes those intensities that are outliers. However, when comparing the errors for the pyramid and disk the threshold appears to increase the errors. This could possibly be due to there being no self shadowing on the disk unlike the pyramid. This could then result in incorrect pixel intensities being removed from the calculations.

The mean surface normal error values across the disk centred within the image, with a radius varying from 0.01m to 0.1m, are illustrated in Fig. 11. For a disk radius of 0.01m, the average absolute angle error of the surface normal is 2.87°. This mean error rises to 13.7° as the radius extends to 0.1m. A similar effect is observed for the two other artefacts as well. This could be caused by a decrease in light intensity further away from the light source, resulting in larger errors occurring away from the centre of the image. Studies are being carried out by the authors to understand this further.

Table 3. Surface normal angular errors from artefact 3.

Method	Error (degrees)			
	whole artefact		1cm disk	
	MAE	MSE	MAE	MSE
Least squares	22.1826	36.9130	3.2240	3.6385
Least squares with threshold	22.1793	36.9109	3.2240	3.6385

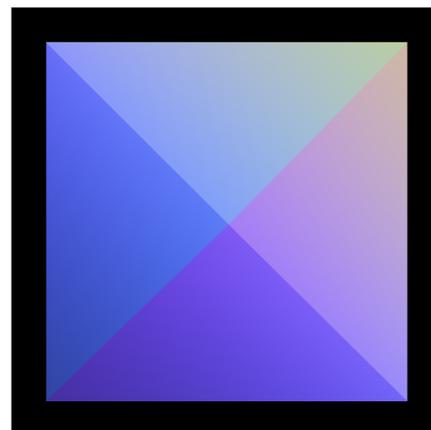


Figure 9. Estimated surface normals for artefact 1.

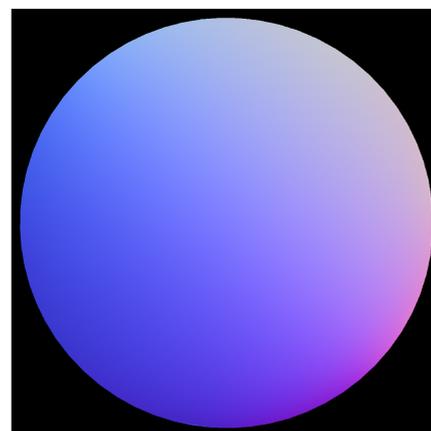


Figure 10. Estimated surface normals for artefact 3.

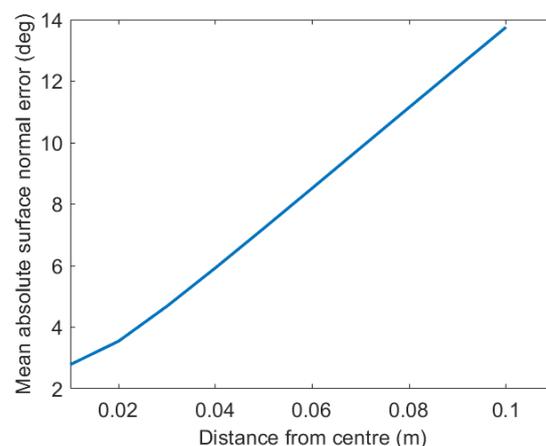


Figure 11. Surface normal errors within disk surrounding centre for artefact 1.

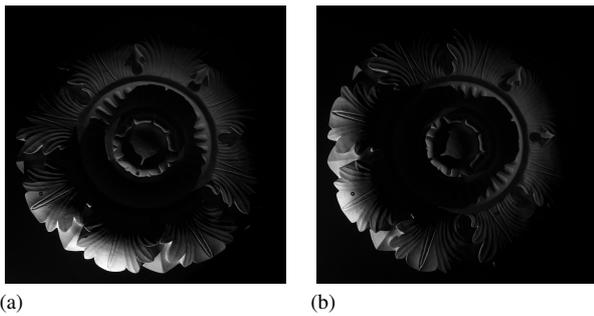


Figure 12. Real image taken with LED (a) number 1 (b) number 2 are turned on.

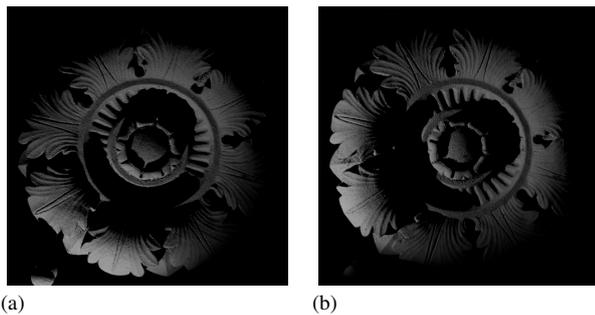


Figure 13. Virtual image when LED number (a) number 1 (b) number 2 are turned on

4.2 Ceiling Artefact

A white moulded plaster object of diameter 229 mm was scanned with a hand-held Hexagon AS1 triangulation scanner tracked with a Hexagon AT960 (AG, 2025) laser tracker to provide a reference point cloud. This laser tracker is a high precision metrology equipment (Khanesar et al., 2023b, Khanesar et al., 2023a). A down-sampled point cloud with 1,000,000 points was created using Meshlab software. The algorithm used for downsampling is explained in Section 2.2.2. The downsampled cloud was triangulated to a mesh surface which was then studied within the virtual Blender environment. The images taken within the virtual instrument are depicted in Fig. 13 and are compared with the images taken in the real world in the dome system in Fig. 12.

Using the 48 images taken from the ceiling object in the real world and the virtual environment, the surface normals of the real and virtual object are generated using the least squares method with threshold (see Fig. 14 and Fig. 15). Visually, the images generated from the virtual environment appear similar to the image generated from the real ones. The range of intensities values in the images appears to be greater for the Blender image. Future work is required to understand the quantitative differences between the two methods.

5. Conclusions

A virtual instrument has been designed within the Blender environment to mimic a practical multi-illumination dome imaging system. The virtual environment gives us the opportunity to test illumination and imaging geometries against different virtual objects in order to understand the metric qualities of different photometric approaches. To perform a simulation within the virtual environment, the light sources were initially

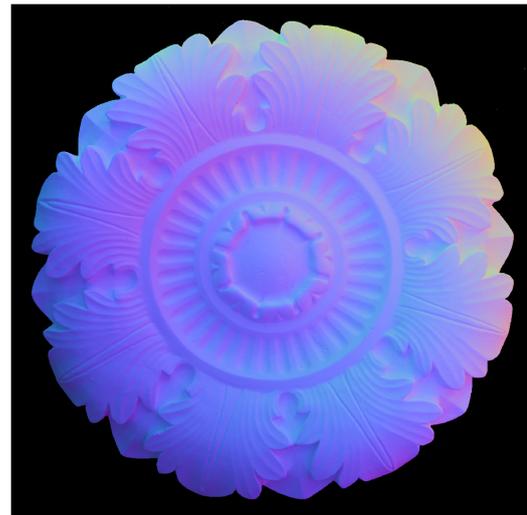


Figure 14. Surface normal estimated from the real images.

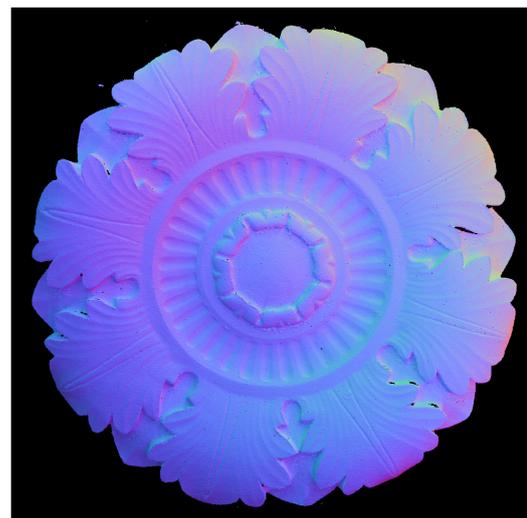


Figure 15. Surface normal estimated from the images taken in Blender.

modelled by adjusting their diffusion and blend parameters to simulate the light sources in the real dome as closely as possible.

We studied a 3D digital artefact with known surface normal angles illuminated sequentially from 48 lighting locations. Two algorithms, one using a least-squares method and the other integrating a threshold value into the least-squares approach, were employed for photometric analysis on images taken by the virtual instrument. The performance of these algorithms was evaluated by measuring the errors in the angles of the surface normal vectors. Subsequently, we investigated surface normal analysis of a real object within the virtual instrument. To obtain the object surface normal, the 3D point cloud was generated using a metrology-grade laser scanner. The surface normal vectors computed for each pixel of the object are subsequently compared between the virtual instrument and the actual environment to showcase the virtual instrument's performance.

Acknowledgements

This work is primarily funded and supported by the Engineering and Physical Sciences Research Council (EPSRC) under grant number: EP/X024059/1—imaging for high performance manufactured aerostructures (Robodome).

References

- AG, L. G., 2025. *Leica AT930/AT960 user manual*. Leica Geosystems, Unterentfelden, Switzerland.
- Bosco, A., Minucci, E., De Luca, D. et al., 2023. Virtual RTI application on 3D model for documentation of ancient graffiti: proposal of a methodology for complex archaeological sites. *Archeologia e Calcolatori*, 34, 59–68.
- Bottalico, F., Niezrecki, C., Jerath, K., Luo, Y., Sabato, A., 2023. Sensor-Based Calibration of Camera's Extrinsic Parameters for Stereophotogrammetry. *IEEE Sensors Journal*, 23(7), 7776–7785.
- Cartucho, J., Tukra, S., Li, Y., S. Elson, D., Giannarou, S., 2021. VisionBlender: a tool to efficiently generate computer vision datasets for robotic surgery. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 9(4), 331–338.
- Cignoni, P., Ranzuglia, G., Callieri, M., Corsini, M., Ganovelli, F., Pietroni, N., Tarini, M. et al., 2011. MeshLab.
- Corsini, M., Cignoni, P., Scopigno, R., 2012. Efficient and flexible sampling with blue noise properties of triangular meshes. *IEEE transactions on visualization and computer graphics*, 18(6), 914–924.
- Dare, P. M., 2005. Shadow analysis in high-resolution satellite imagery of urban areas. *Photogrammetric Engineering & Remote Sensing*, 71(2), 169–177.
- Eastwood, J., Sims-Waterhouse, D., Weir, R., Piano, S., Leach, R., 2019. Autonomous close-range photogrammetry using machine learning. *Proc. ISMTII2019*.
- Enebuse, I., Foo, M., Ibrahim, B. S. K. K., Ahmed, H., Supmak, F., Eyobu, O. S., 2021. A comparative review of hand-eye calibration techniques for vision guided robots. *IEEE Access*, 9, 113143–113155.
- Flavell, L., 2011. *Beginning blender: open source 3d modeling, animation, and game design*. Apress.
- Gabara, G., Sawicki, P., 2017. Study on 3d point clouds accuracy of elongated object reconstruction in close range—comparison of different software. *Environmental Engineering. Proceedings of the International Conference on Environmental Engineering. ICEE*, 10, 1–7.
- Grzelec, M., Łojewski, T., 2023. Applications of Reflectance Transformation Imaging (RTI) for Books and Objects on Paper. *Journal of Paper Conservation*, 24(1), 16–30.
- Gschwandtner, M., Kwitt, R., Uhl, A., Pree, W., 2011. Blesor: Blender sensor simulation toolbox. *Advances in Visual Computing: 7th International Symposium, ISVC 2011, Las Vegas, NV, USA, September 26-28, 2011. Proceedings, Part II 7*, Springer, 199–208.
- Hartley, R., Zisserman, A., 2003. *Multiple view geometry in computer vision*. Cambridge university press.
- Hosen, M. S., Ahmmed, S., Dekkati, S., 2019. Mastering 3D Modeling in Blender: From Novice to Pro. *ABC Research Alert*, 7(3), 169–180.
- Kazhdan, M., Hoppe, H., 2013. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3), 1–13.
- Kent, B. R., 2015. *3D scientific visualization with Blender®*. Morgan & Claypool Publishers.
- Khanesar, M. A., Todhunter, L., Pawar, V., MacDonald, H. C. L., Robson, S., Piano, S., n.d. euspen's 24th International Conference & Exhibition, Dublin, IE, June 2024.
- Khanesar, M. A., Yan, M., Isa, M., Piano, S., Ayoubi, M. A., Branson, D. T., 2023a. Enhancing Positional Accuracy of the XY-Linear Stage Using Laser Tracker Feedback and IT2FLS. *Machines*, 11(4), 497.
- Khanesar, M. A., Yan, M., Isa, M., Piano, S., Branson, D. T., 2023b. Precision denavit–hartenberg parameter calibration for industrial robots using a laser tracker system and intelligent optimization approaches. *Sensors*, 23(12), 5368.
- Pottier, C., Petzing, J., Eghtedari, F., Lohse, N., Kinnell, P., 2023a. Developing digital twins of multi-camera metrology systems in Blender. *Measurement Science and Technology*, 34(7), 075001.
- Pottier, C., Petzing, J., Eghtedari, F., Lohse, N., Kinnell, P., 2023b. Developing digital twins of multi-camera metrology systems in Blender. *Measurement Science and Technology*, 34(7), 075001.
- Sammons, J. F. D., 2018. Application of Reflectance Transformation Imaging (RTI) to the study of ancient graffiti from Herculaneum, Italy. *Journal of Archaeological Science: Reports*, 17, 184–194.
- Sturm, P., 2014. Pinhole camera model. K. Ikeuchi (ed.), *Computer Vision: A Reference Guide*, Springer US, Boston, MA, 610–613.
- Woodham, R. J., 1979. Photometric stereo: A reflectance map technique for determining surface orientation from image intensity. *Image understanding systems and industrial applications I*, 155, SPIE, 136–143.
- Wu, L., Ganesh, A., Shi, B., Matsushita, Y., Wang, Y., Ma, Y., 2011. Robust photometric stereo via low-rank matrix completion and recovery. *Computer Vision—ACCV 2010: 10th Asian Conference on Computer Vision, Queenstown, New Zealand, November 8-12, 2010, Revised Selected Papers, Part III 10*, Springer, 703–717.