Enhancing Spatial Image Datasets For Utilisation in A Simulator for Smart City Transport Navigation

Lepekola I. Lenkoe
Department of Electrical, Electronic and Computer Engineering
Central University of Technology, Free-State
Bloemfontein, South Africa
e-mail: pexonl3@gmail.com

Ben Kotze)
Department of Electrical, Electronic and Computer Engineering
Central University of Technology, Free-State
Bloemfontein, South Africa
e-mail: bkotze@cut.ac.za

***Abstract*—The introduction of Google Street Views has brought to surface a method for roof-mounted mobile cameras on vehicles. This method is regarded as one of the highly known and adopted methodologies for capturing street-level images. This article contributes to the development and implementation of Image-Based Rendering techniques by presenting a technique utilising a hexagon-based camera configuration model for image capturing. Upon the image capture stage, each segment camera is stored in a specific folder relative to the camera number (i.e., camera 1 = folder 1). Subsequently, the optimal image rendering process of each image blending takes place inside Blender3D software where image datasets are rendered for utilisation in a simulator. Utilising the Structure of Motion algorithm, dense point image, and its features, match detection is obtained. This article further contributes to the results process that allows for free movement within a 3D rendered scene by permitting for back and forward movement as compared to a slide show that only permits for forwarding motion.**

Keywords-Image-Based Rendering; Blender3D; Simulation; Datasets; Google Street Views; Smart Cities.

I. INTRODUCTION

Over the years, different kinds of techniques have been proposed for image data collection and image rendering. In the past few years, the Image-Based Rendering (IBR) technique has gained a lot of attention mainly in the image processing, computer vision, and computer graphics community. In addition to IBR's interest in communities, spatial knowledge which is regarded as an essential subject that makes use of geospatial statistics such as geoscience, geography has shown growth with regards to multi-functional ecosystems.

The street views have debilitated previous restrictions on the availability of data sources for evaluating streets [1][2]. Furthermore, Model-Based Rendering (MBR) is classified as an easy method for reconstructing virtual view from any arbitrary viewpoint by using explicit 3D geometric and model and texture information about the scene, while IBR is a method that constructs virtual view by using several images captured beforehand [3].

The captured images can provide valuable information about the incident, e.g., location. The location has the exact Global Positioning System (GPS) coordinates, which can also be an estimation of the location. Figure 1 presents the graphical representation of inquiring a query image to the reference database to find the match between a stored image and the query image [4].

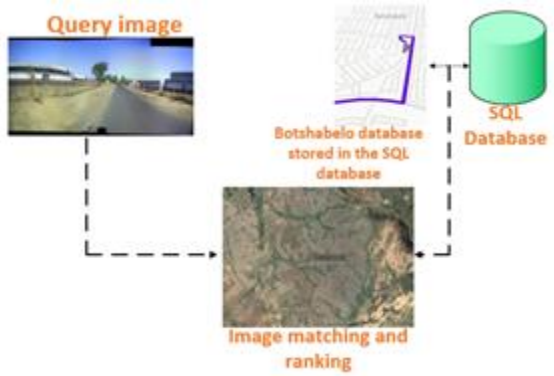

Figure 1. Street-view to overhead view image matching.

This article seeks to develop a 3D rendered model from a 2D captured model and allow for back and forward motion within a simulator. The 3D data acquisition process provides the probe position and orientation that remain in static order to produce accurate datasets. A single 3D capacity is not able to support the translational motion of the simulated probes, thus the need to develop a methodology for recording and capturing single 2D images and amalgamating the images into multiple 3D images within a single unit. Furthermore, the issue of emulating street-view images for multiple image transitions for application in geolocalisation for utilisation in a simulator needs to be investigated.

This paper is structured as follows: Section II describes the aim and objectives, and also outlines the contributions of this research. Additionally, Section III gives a brief description of the work done in this field, while Section IV describes the process and the development of the proposed model from the captured data up to data simulation. Section V discusses the results obtained from this study, while Section VI discusses the conclusion of the study.

 ISBN: 978-1-61208-863-1

A. *The Objective of This Study Is Therefore To:*

- To identify datasets with capabilities such as frame position, frame elevation, and frame indexing.
- To incorporate the system into the simulation system in real-time for increasing the reality of the simulation system in different geographical locations.

B. *Original Contributions of This Research Article*

This research article has produced the following contribution in furthering the article the knowledge contribution in the field of computer vision as follows:

- 6 degree of freedom- where the user can move in any direction as opposed to the use of single slides show that allows for one-directional movement in a street view scene.
- The ability of the application to use multiple cameras between 3 to 6 inputs as opposed to the use of single omnidirectional camera feedback and still obtain the same output rendered panoramic and simulated results.

II. LITERATURE REVIEW

Quintessentially, massive amounts of image collections are presented as slideshows, which are arguably the practical way but with the current technological advancement, these methodological approaches are deemed not engaging. The change in scenery due to technological improvements has led to a wide study pool, which also cites the research conducted by Sivic et al. [5]. In that work, the authors highlight the connection of clustering visually similar images together to create a virtual space in which the users are free to change position from one image to another. This virtual space modelling can be obtained by utilising intuitive 3D control objects such as move left/right, zoom in/out and rotate. Sivic further supports his work by outlining that the displayed images in a correct geometric and photometric alignment concerning the current photo, results in a smooth transition between multiple images. In addition, Kopf et al. [6] present a method of combining images in the street view system by stitching the image side views. This approach means that standing on the street and looking in either the left or right direction of a certain street together to generate a long street slide for users to quickly browse this street is feasible.

Despite the excellent way of viewing the side scene of a certain street from Kopf`s methodology, however, the practicality of that method is not always the case while driving or walking. Kopf further presents the street slide methodology which combines the nature of bubbles provided by perspective stripe panoramas. Kopf further presents integrated annotations and a mini-map within the user interface to provide geographic information as well as the additional affordances for navigation.

Kopf`s work relates to Gortler et al. [7] due to their classic approaches of image-based rendering such as Lumigraph. Kopf's work is further supported by Agarwala et al. [8] by emphasising the utilising of the correlation alignment techniques for aligning adjacent vertical strips instead of modeling the full 3D geometric proxies. Subsequent to these approaches, rendering displacement in maps requires the surface to be adaptively retesselated [9].

The synthetic environments for the extraction of the depth information during the rendering process. The geometric construction relates to both implicit and explicit construction [10]. This is as a result of the view dependency, which means that the explicit geometric rendering much relies on the known approximate environment [11]. The use of the explicit rendering becomes much tricky in an informal environment/settlement due to the tiring exercise of data collection which is skewed since residents can build on the road and mountains.

III. METHODOLOGY

The simulator model design is developed for a driver or person riding a bicycle inside the simulator following a track in either forward or reverse direction. With this said, the initial thought design was to develop the Spatial Image Datasets (SID) based on the nonagon (9) camera configuration model. However, with the current technological capabilities, an omnidirectional camera could have been utilised to conduct this activity. The reason for not utilising the omnidirectional camera is because the rendering construction specifically for this research article requires individual camera feeds as opposed to one 360º feed. It is however, Yuah et al emphasized that the 6D object pose estimation is a challenging but important research direction in visual direction [12].

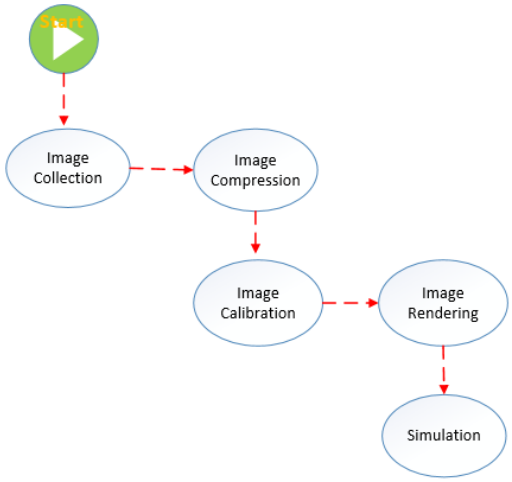

Figure 2. System model layout.

The system development was then initiated based on the model layout as indicated in Figure 2. Figure 2 depicts the technical approach for the design and development of the simulator utilising the hexagon camera configuration model.

Subsequently, Wang et al. [13] depict their method for integrating the IBR and NERF methods into s new learning method that generates a continuous multiple source view for rendering novel views. However, Wang's study does not clearly outline the dataset generation and the creation of

scenes from the dataset of an image that was not initially captured.

A. *Image Capturing and Collection*

The image capture section consists of the following apparatus; camera, images, and GPS coordinates. The six (6) mounted camera configuration model for image capture while the vehicle is in motion, and each image is treated as a frame. During the image capturing process, each camera image dataset is stored in its specific folder i.e., camera 1 = folder 1.

B. *Image Compression*

The scene depiction utilising multiple depth images in a dataset format is compressed. The image samples are then captured and obtained from the camera capture by delaying the camera switching algorithm between multiple cameras by 3ms (the delay period was based on the trial and error test conducted between 1ms – 5ms switching).

Furthermore, the Lossy compression algorithm is performed for the redundant processing of image information. Additionally, the image dataset is simulated without noise for better performance verification as outlined in Figure 3 utilising the Lossy compression algorithm.

Figure 3. image compression on a JPE file.

Figure 3 depicts the image types for the captured images. Subsequently showing the timestamp and the GPS coordinates for the image dataset created. Additionally, the image compression framework is used to obtain the results output outlined in Figure 4.

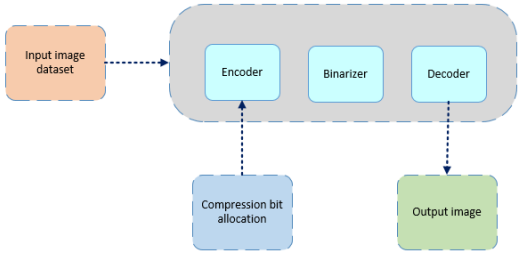

Figure 4. Compression structure.

Figure 4 depicts the compression structure that utilised the input image datasets for the network training sample. This is obtained by specifically setting the image dataset for image recognition where the compressed image datasets are compared against the raw image datasets. Additionally, the image compression bit allocation is then used to calculate the compression alterations. However, the alterations depend on the size of the dataset. Notably, each image in the dataset is compressed individual is to retain the image quality. However, this is deemed as a tedious process, especially for huge datasets.

C. *Image Calibration*

The data collection and image compression process, are determined by the reduction in the image size but still keeping the image resolutions intact. The image calibration model utilises the pinhole camera model that introduces some image distortions. These image distortions that are seen in this process are classified as radical image distortion. The radical image distortion was calculated as follows:

$$X_{distortion} = x(1 + k_1 r^2 + k_2 r^2 + k_3 r^6) \quad (1)$$

$$Y_{distortion} = y(1 + k_1 r^2 + k_2 r^2 + k_3 r^6) \quad (2)$$

Where: x = original; x location on the imager
y = original y location on the imager
k = radical distortion coefficient
r = radical distortion form Taylor series

The system setup approach makes use of the chess pattern for camera calibration setup. Some calibration methods in the literature rely on 3D objects. However, through the tests conducted, the flat chessboard pattern approach is deemed appropriate for this research article due to the method been less complex and easily understood even by non-technical individuals.

The camera calibration process is as follows:

- Capture 20 chessboard images from different poses;
- Find the chessboard corners;
- Find the intrinsic matrix, distortion coefficients, rotation vectors, and the translation vector;
- Store the .xml file.

Following the process completion, OpenCV for the python library is utilised to compute the results from the .xml file. This, therefore, allows for the reuse of the code for multiple cameras, which is relevant for this research paper which uses a hexagon camera configuration model with rotational image capture technique.

 ISBN: 978-1-61208-863-1

Figure 5. Black and white test match on a chessboard.

The black and white squared pattern match-finding is outlined as indicated in Figure 5 for the identification of features within a 2D image. The pattern match-finding is important in the extraction of image data from multiple image datasets that are the same.

D. Image Rendering Procedure

The image rendering technique in the context of this research article focuses on the known camera parameters and undistorted images for the rendering of the scenes. These images are reconstructed, and the texture is applied to their structure before rendering simulation can be executed.

E. Structure for Motion

The basic operation of the structure of motion in this research article follows the Detection of 2D features on every image. In this step, a 2D feature is detected using the Scale-Invariant Feature Transform (SIFT) algorithm as indicated in Figure 6. Figure 6 depicts the original image captured with one of the mounted cameras from the hexagon configuration model.

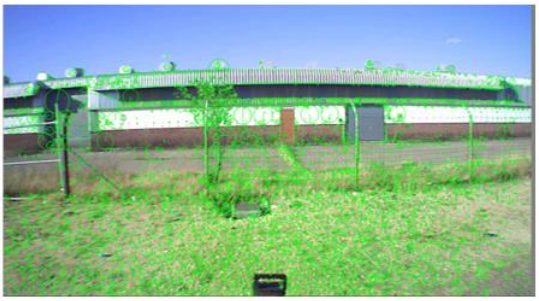

Figure 6. Detecting the image feature using SIFT.

The camera parameters are captured, and the patch-based stereo and semi-global matching are used to generate point tracks, depth-maps as well as the points cloud. Upon successful generation of these variables, a mesh of the scene is created as indicated in Figure 7. Finally, all the refined depth maps are merged to get the final reconstruction.

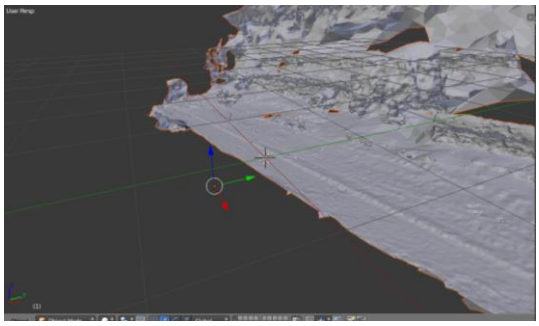

Figure 7. Mesh output from the multiview stereo.

The multi-view stereo algorithm is further used as a Semi-Global Matching algorithm (SGM), where it consists of calculation, aggregated costs, disparity computation, and the extension for multi-baseline matching.

F. Texturing

Patches are formed onto the faces of the model, and the texture patches colours are adjusted. This is achieved by adjusting colour between adjacent patches. This results in seamless texture across the model.

G. Data Simulation

The model with texture consists of the path/road and environment (trees and buildings). Lighting, Camera, and Collides are added. Lighting is added to illuminate the model to simulate the light from the sun. The movement of the camera simulates a vehicle moving through the path/road created with the model. The movement gets its inputs from the keyboard. Colliders are Blender3D objects that provide physics attributes to the model, and they are added to prevent the user from moving beyond the required space within the simulator.

IV. RESULTS

The results outlined in this section depict the image rendering framework for the 364 .JPEG images that were captured on each camera at a total dataset worth 2184 JPEG images at a high resolution of 1280X720 pixels at a total size of 39.8MB. The system required a dynamic scene with 6 cameras arranged a 2D arc at a spanning of about 600cm apart from each other. Additionally, each camera frame comprised of 364 JPEG images were captured from the real scene, and only then the process of image matching and texturing is applied using depth maps resulting with the output textured PNG image of 25.2 MB of 8192X9192 pixels resolution.

 ISBN: 978-1-61208-863-1

A. *Rendering Simulation Outcomes*

Figure 8. Denser point of cloud with Multiview stereo.

Following the creation of the mesh output, the texture was then generated by taking models, images, and the camera position (this was achieved through the use of the GPS coordinates). This was accomplished in the meshroom function by selecting 8192 texture slides and by unwrapping this method as indicated in Figure 8.

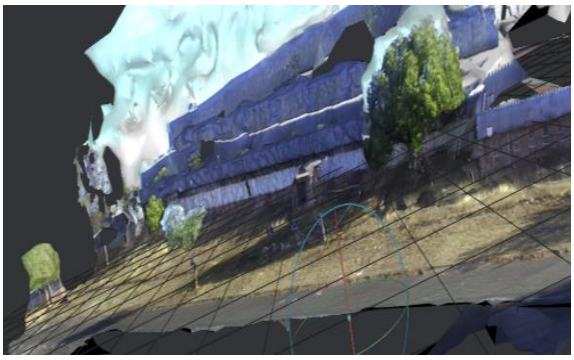

Figure 9. Generated texture.

Figure 9 depicts the depth map restoration and colour images were paired up to create the application of texture to the 3D mesh model and warp to the new viewpoint. Furthermore, the extraction of the depth map model from the 3D mesh, and the shortcomings were observed to be the delay by which the mesh update duration is long and as a result, affects the depth map extraction.

The final output that is represented in Figure 10 depicts the rendered view of the panoramic street views. The panoramic street view can be viewed in multiple viewing angles as indicated in Figure 10.

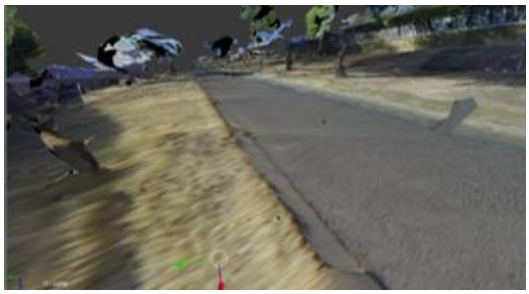

Figure 10. Rendered image horizontal view.

Figure 10 outlines the achieved and projected rendered 3D image from a horizontal street view in an omnidirectional manner. These results are outlined to provide the Freedom of Movement (FoM) and the views which were not captured by the camera but through rendering the uncaptured scene can be viewed.

V. CONCLUSION

The feature detection and matching technique was observed as the best technique in detecting and matching the images from multiple image datasets. As a result, the use of the image-based rendering technique utilising hexagon camera configuration was proposed as an ideal method in this paper.

As compared against the literature, this paper has additionally contributed to the end-to-end process from the development of an image capturing model to rendering a scene with and being able to move forth and back and view the scene that was not originally captured.

Furthermore, the paper has demonstrated the feasibility and efficiency for the integration of the IBR model from a hexagon camera configuration model, and the simulator as indicated:

- The incorporation of the system into the simulation system in real-time for increasing the reality of the simulation system in different geographical locations;
- To simulate a rendering technique for improvement of visual, spatial, and quality of the panoramic images for location identification.

ACKNOWLEDGMENT

Acknowledgment for financial assistance and academic assistance from the Central University of Technology, Free-State (CUT, FS) and Mr. T.G Kukuni for his academic expertise and unending support.

REFERENCES

[1] L. Yin, Q. Cheng, Z. Wandy and Z. Shao, "Big data for pedestrian volume: Exploring the Sasol," Exploring the use of Google street view images for pedestrian counts, no. 63, pp. 337-345, 2015.

[2] N. Runge, P. Samsonov, D. Degraen and J. Schoning, "No more autobahn: Scenic route generation using Googles Street View," In Proceedings of the I, 7-10 March 2016.

[3] R. Sato, S. Ono, H. Kawasaki and K. Ikeuchi, "Photo-Realistic Driving simulator using Eigen Texture and Real-Time Restoration Techniques by GPU," vol. 6, no. 2, p. 87, 2 December 2008.

[4] N. N. Vo and J. Hays, "Localizing and Orienting Street View Using Overhead Imagery," p. 2, 2016.

[5] J. Sivic, B. Kaneva, A. Torralba, S. Avidan and W. T. Freeman, "Creating and Exploring a Large Photorealistic Virtual Space," Proc. IEEE Workshop on Internet Vision, 2008.

 ISBN: 978-1-61208-863-1

[6] J. Kopf, B. Chen, R. Szeliski and M. Cohen, "Street slide: Browsing Street Level Imagery," ACM Trans. Graphics, vol. 29, no. 4, 2010.

[7] S. J. Gortler, R. Grzeszczuk, R. Szeliski and M. F. Cohen, "The Lumigraph. In Computer Graphics Proceedings, Annual Conference Series," in ACM SIGGRAPH, Proc. SIG-GRAPH 96, New Orleans, 1996.

[8] A. Agarwala, M. Agrawala, M. Cohen, D. Salesin and R. Szeliski, "Photographing long scenes with multi-viewpoint panoramas," ACM Transactions on Graphics 25, pp. 853-861, 3 August 2006.

[9] K. Moule and M. McCool, "Efficient bounded adaptive tesselation of displacement maps," In proc. of GI, pp. 171-180, 2002.

[10] H. Y. Shum and S. B. Kang, "A review of image-based rendering techniques," p. 1.

[11] P. E. Debevec, C. J. Taylor and J. Malik, "Modeling and Rendering Architecture from photos: A Hybrid geometry- and image-based approach," In ACM SIGGRAPH, pp. 11-20, 1996.

[12] H. Yuan and R. Veltkamp, "A 3D photo-realistic environment simulator for mobile robots," IEEE Robotics and Automation, vol. 6, no. 2, p. 143, 2021.

[13] Z. Wang, Q. Wang, K. Genora, P. Srinivasan, H. Zhou, J. Barron, R. Brualla, N. Snavely and T. Funkhouser, "IBRNet: Learning multi-view Image-based Rendering," p. 1, February 2021.

 ISBN: 978-1-61208-863-1