symmetry

MDPI

Article

Gaze-Controlled Virtual Retrofitting of UAV-Scanned Point Cloud Data

Pavan Kumar B. N., Adithya B., Chethana B., Ashok Kumar Patil and Young Ho Chai *

Virtual Environments Lab, Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul 06974, Korea; pavanbn8@gmail.com (P.K.B.N.); adithyab@cau.ac.kr (A.B.); chethanab26@gmail.com (C.B.); ashokpatil03@hotmail.com (A.K.P.)
* Correspondence: yhchai@cau.ac.kr; Tel.: +82-2-820-5335

Received: 26 October 2018; Accepted: 28 November 2018; Published: 29 November 2018

Abstract: This study proposed a gaze-controlled method for visualization, navigation, and retrofitting of large point cloud data (PCD), produced by unmanned aerial vehicles (UAV) mounted with laser range-scanners. For this purpose, the estimated human gaze point was used to interact with a head-mounted display (HMD) to visualize the PCD and the computer-aided design (CAD) models. Virtual water treat plant pipeline models were considered for retrofitting against the PCD of the actual pipelines. In such an application, the objective was to use the gaze data to interact with the HMD so the virtual retrofitting process was performed by navigating with the eye gaze. It was inferred that the integration of eye gaze tracking for visualization and interaction with the HMD could improve both speed and functionality for human–computer interaction. A usability study was conducted to investigate the speed of the proposed method against the mouse interaction-based retrofitting. In addition, immersion, interface quality and accuracy was analyzed by adopting the appropriate questionnaire and user learning was tested by conducting experiments in iterations from participants. Finally, it was verified whether any negative psychological factors, such as cybersickness, general discomfort, fatigue, headache, eye strain and difficulty concentrating through the survey experiment.

Keywords: gaze interaction; head-mounted display; point cloud; virtual retrofitting

---

1. Introduction

Three-dimensional (3D) data, particularly PCD, and 3D models are used every day to represent environments and objects [1]. Modern PCD collection, processing, and visualization have considerable importance for many applications, including industrial design, virtual reality , augmented reality, and retrofitting. Laser range scanners have become increasingly used for sampling and creating 3D scenes, which has led to massive PCD.

Interactions with the large PCD have become challenging, such as viewing, retrofitting, and presenting the data. Interaction techniques in 3D for navigating viewpoints and models are now conventional and let the user choose the best possible viewpoint to analyze and retrofit the models. For example, redesigning route plans for transporting pipes and equipment in industrial plants, e.g., petrochemical or thermal plants, and hydromodification in water treatment facilities are particularly challenging [2] Hence, laborers involved in maintenance, reconstruction, and upgradation tasks face risks because of the unknown defects and/or unidentified complex objects in the plant. Maintenance and upgrading plant facilities frequently require components to be redesigned and/or added. Validating these upgrades (retrofitting) is time-consuming and tedious.

Therefore, virtual retrofitting applications are required that can analyze and optimize retrofit decisions, reducing the required time for currently typically lengthy projects. An accurate retrofit model of an existing pipeline plant in heavy industries would allow easier visualization and analysis to ensure that the proposed retrofit meets the requirements and provides the best value. Retrofitting has previously been achieved manually by professional staff using commercial software [3], and no previous virtual retrofitting studies have been documented.

For complicated engineering projects that include redesign tasks, 3D model retrofitting using PCD would require considerable staff time struggling with current interfaces. Therefore, intuitive interaction methodologies for virtual retrofitting of CAD models are urgently required to help validate complex projects.

This study adopted gaze-controlled interaction for virtual retrofitting using a HMD to increase speed and functionality. Eye gaze provides various promising information, including the individual's cognitive state [4]. In a controlled immersive virtual environment (VE), the natural quickness of eyeball movement complemented by human intuition provides excellent interaction input for controlling objects in the virtual world [5].

3D scanning has been widely employed across many industries for reverse engineering and part inspection for many years [6], and acquires the 3D shape with detailed geometry information. However, acquiring PCD in heavy industrial plants with numerous pipelines is manually strenuous. An UAV could provide spatial sensory information at much higher resolution by inspecting at a considerably closer range [7], and can access many environments where human access is restricted. Thus, a laser scanner mounted on a UAV could map the entire industrial environment, producing comprehensive PCD.

Therefore, we leveraged eye gaze to analyze and retrofit CAD models with PCD in a VE. The remainder of this paper is organized as follows. Section 2 discusses previous studies related to human gaze, eye tracking, PCD, and retrofitting. Section 3 details the proposed method, and Section 4 presents experimental results and analysis from applying the proposed system to a practical case study. Section 5 summarizes and concludes the paper.

2. Related Works

Various research on gaze tracking were conducted starting from video-based eye tracking study on a pilot-operated airplane [8]. Gaze tracking was enhanced with the objective of improving accuracy and reducing the constraints on the users [5]. Rapid advancements in computing speed, digital video processing, and low cost hardware have made gaze tracking equipment relatively accessible to users, with many current applications in gaming, web advertisements, and virtual reality [9].

Exploring PCD in immersive VE is an emerging research area. Various attempts have been made to enable human observers to explore PCD in VE [10], with mixed success. Exploration of 3D models and PCD in VE to securitize errors is considered effective [11], but developing new ways to interact with the virtual objects for modification and redevelopment is challenging [12].

Gaze based interaction with the VE has not been widely explored previously. The Pupil software [13] offers add-ons to HTC VIVE (HTC and Valve Corp., Bellevue, WA, USA) [14] that enables extracting the user's gaze, and various VE interaction models have been proposed [15]. Remote eye tracking has been recently introduced for TV panels to enable gaze-controlled functionality, such as switching channels and navigating menus [5]. Different gaze tracking devices, wearable and non-wearable with single or multiple cameras, have been studied [16–18]. Near-infrared cameras and illuminators have been employed for most non-wearable gaze tracking systems. Eye gaze interaction in VE has become a strong research and application focus [19], and Piumsomboon et al. [20] reviewed a number of promising eye gaze based techniques, including

1. duo-reticles,
2. radial pursuit,
3. nod and roll.

Studies have also considered self-calibrating eye trackers embedded in an HMD to improve tracking and effective interaction [21].

Manipulating objects in a VE and related challenges have been studies for scenarios including remote collaboration [22], object manipulation on a table top VE [23], and object manipulation in immersive environments [24]. However, no previous virtual interaction model has employed eye gaze as an input to modify or interact with the virtual PCD.

PCD are a common input for geometric processing applications, and their acquisition and reconstruction are significant challenges for many applications, including virtual reality, digital industries, augmented reality, and retrofitting [25–27]. PCD have been acquired using a variety of laser scanners, and the availability of such 3D data is expected to pose new challenges to efficiently view, edit, and interact [28].

3. Proposed Method

The main objective for the current study was to develop a virtual retrofit application that provided affordable upgrades for complex engineering models in heavy industrial plants to support and help decision-making for retrofit projects. Currently, plant upgrades are high risk projects and traditional retrofit projects require engineers to make multiple site visits for field survey measurements and checking design aspects.

The proposed gaze-controlled virtual retrofitting method allows immersive retrofitting to be performed virtually and interactively. The proposed method has the potential to reduce errors and interferences that can occur with onsite construction work.

It provides for precise addition and deletion of CAD models to update existing models in the virtual system. We defined four gaze interactions as follows:

1. Inserting the CAD model into the VE.
2. Translating the CAD model along x- and y-axes to retrofit with the PCD.
3. Deleting the CAD model from the VE.
4. Zooming in and out within the VE:

$$S = \frac{(M_r^x - M_r^n)}{\left(M_{gN}^x - M_{gN}^n\right)}, \tag{1}$$

$$H_{gaze} = M_r^n + S(gN - M_{gN}). \tag{2}$$

The gaze data given by the eye tracker were in the normalized coordinate system. In contrast, the HTC VIVE rendered at a display resolution of 1080 ×1200 pixels per display. Values from the eye tracker were mapped to adapt to the HTC VIVE (In Figure 7) display resolution as shown in Equations (1) and (2), where S is the slope; and M_r^x and M_r^n are maximum and minimum HMD resolution, respectively;. M_{gN}^x and M_{gN}^n are the maximum and the minimum values of the gaze normal, respectively; and gN is the current gaze normal. Gaze normal is the gaze vector represented as a unit vector.

In immersive VE, the number of consecutive user blinks is lower than that in a non-immersive VE [29]. Hence, user eye blinks were considered to be a useful interaction. To maintain consistency between consecutive blinks, we set the blink interval 37–45 ms. We used eye blinks to insert predefined models for virtual retrofitting. Algorithm 1 shows the eye gaze interaction workflow.

Algorithm 1 Eye gaze interaction for retrofitting in a virtual environment (VE).

1: Retrieve $GazeNorms(x, y)$ and Eye Blink data from the IPC backbone messaging bus of the Pupil service.
2: Map the retrieved $GazeNorms(x, y)$ using the mapping function described in Equations (1) and (2), such that $H_{gaze} = f(GazeNorms(x, y))$.
3: Check for eye blinks. If true, move to step 4; else, go to step 7.
4: $getNoBlinks()$ from the IPC backbone messaging bus of the Pupil service.
5: If $37 < blink_{interval} < 45$, then go to step 6; else, ignore blinks.
6: Execute the corresponding interaction routine.

1. 2 blinks $\rightarrow$ zoom in.
2. 3 blinks $\rightarrow$ zoom out.
3. 4 blinks $\rightarrow$ toggle model (delete current model and insert a new model).

7: Execute switched routines $(getGazeTrajectory(H_{gaze}))$.
8: Interactions

1. Double gaze up: translate model in focus by 1 unit in the positive y direction.
2. Double gaze down: translate model in focus by 1 unit in the negative y direction.
3. Double gaze right: translate model in focus by 1 unit in the positive x direction.
4. Double gaze left: translate model in focus by 1 unit in the negative x direction.
5. Otherwise: Ignore gazeTrajectory.

9: Loopback from step 3 until the end of the session.

Figure 1 shows how the various modules and apparatus were integrated into the virtual retrofitting application.

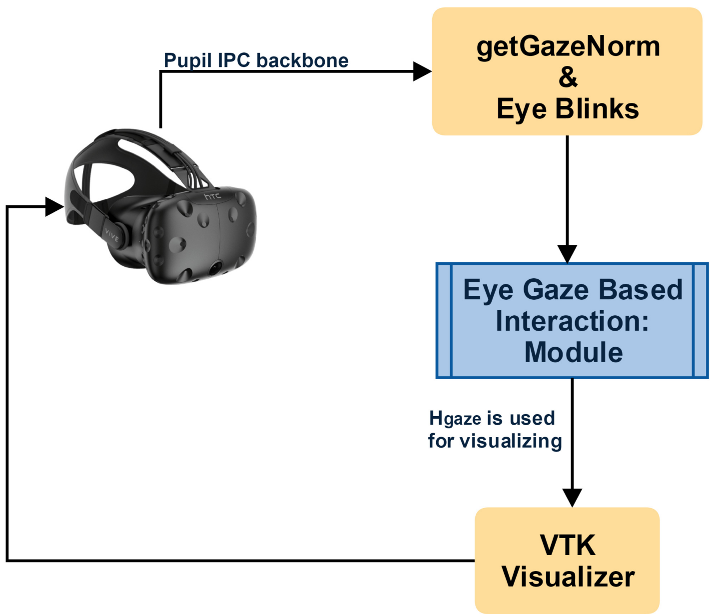

Figure 1. Module integration for the proposed virtual retrofitting application.

4. Experimental Results and Analysis

4.1. UAV Setup

We chose the DJI Matrice 100 UAV (DJI, Shenzhen, Guangdong, China) with V.1.3.1.10 firmware (DJI, Shenzhen, Guangdong, China) and TB47D battery. This provided stabilized flight and 13 min of hovering time with 1000 g payload. The UAV supported dual battery, increasing flight time to 40 min and also included an advanced flight navigation system incorporating GPS, flight controller, and DJI lightbridge, allowing it to perform complex tasks and operate in all environment conditions. Table 1 provides DJI Matrice 100 technical specifications [30].

Table 1. Technical specifications for the DJI Matrice 100.

Parameters	Values
Drone type	Fixed wing with intelligent flight battery
Battery	5700 mAh LiPo 6S
Video output	USB, High-Definition Multimedia Interface-Mini
Flight specification	Ascent: 5 m/s (max) Descent: 4 m/s (max)
Operating temperature	−10 °C to 40 °C

4.2. UAV and Velodyne Sensor Integration

We used the Velodyne LiDAR Puck LITE (Velodyne LiDAR, San Jose, CA, USA) for partial PCD acquisition. This is a lightweight version specifically designed to meet relatively low UAV weight restrictions. The sensor was a 16-channel LiDAR scanning 360° in the horizontal field of view (FOV) and ±15° in the vertical FOV. The sensor had low power consumption, scanned the environment in 3D at up to 20 Hz, generating approximately 300,000 points per second, with a range up to 100 m, and weighed 590 g, making it ideal for mounting on a UAV [31]. The Matrice 100 offers a hardware interface to share its power supply with third party hardware, such as the Velodyne. We used a DROK voltage regulator (Droking, Hong Kong, China) to share UAV battery power to the Velodyne sensor, as shown in Figure 2. Power supply for the manifold was connected through a dedicated power port on the UAV and the Velodyne was connected to the manifold through a LAN cable.

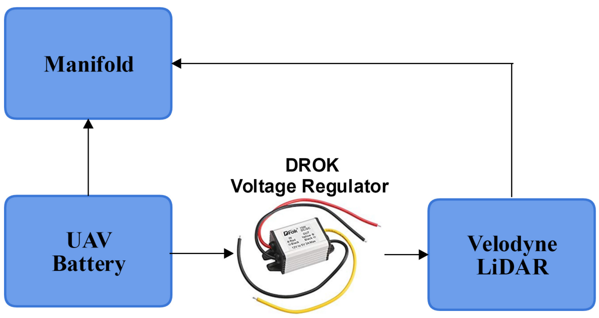

Figure 2. UAV and Velodyne sensor hardware integration.

We optimized the payload weight distribution and Matrice 100 configuration by trial and error, as shown in Table 2 and Figure 3.

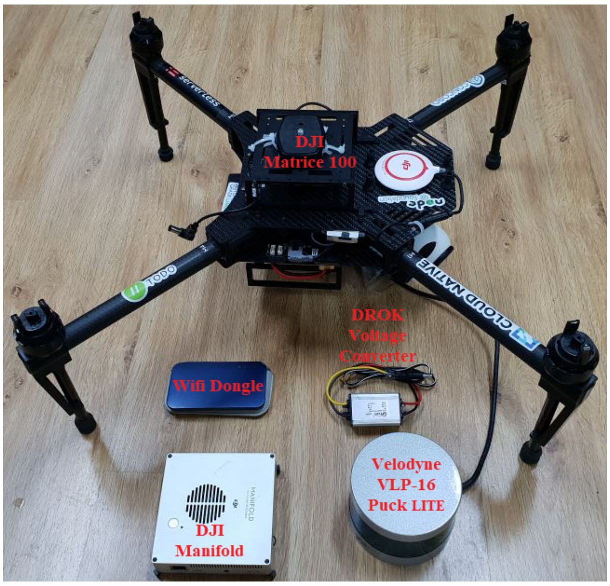

Figure 3. Trial and error optimized setup for data acquisition.

Table 2. Trial and error optimized payload distribution.

Device	**Weight (g)**
Wi-Fi dongle	80
DROK voltage regulator	9
DJI manifold	200
Velodyne LiDAR Puck Lite	590
Total	**879**

4.3. Recording Software Setup

We used the Indigo robot operating system (ROS-Indigo) [32,33] to record sensor data, running over Ubuntu 14.04. The Velodyne sensor was initially triggered, and then sensor data were stored as an ROS bag using the built-in rosbag node. The ROS bag was then converted to visualizable PCD format and stored on the onboard computer. PCD files were remotely transferred to another host machine over secure file transfer for subsequent processing.

4.4. Acquisition of 3D PCD

The Velodyne sensor was mounted on the chosen, as shown in Figure 4, and we manually calibrated the UAV to achieve stable flight, as shown in Figure 5. Scanning was triggered from a remotely connected computer to the onboard computer through the DJI manifold. The Matrice 100 quadcopter provided an onboard software development kit to simplify programming.

Figure 4. Velodyne Puck LITE sensor mounted on the chosen UAV DJI Matrice 100.

Figure 5. UAV manual calibration at the measurement site.

4.5. Preprocessed 3D PCD

The acquired PCD were processed to create a more detailed partial 3D point cloud models of the scanned environment. Alignment problems can arise depending on the application when a similar scene or environment for an area of interest was acquired multiple times from multiple views.

We used the commercial Trimble laser scanner (Trimble, Sunnyvale, CA, USA), with accuracy up to ±2 mm, to acquire the 3D PCD of the environment, as shown in Figure 6a. Figure 6b shows a typical partial PCD generated by Velodyne, which was used to check correct orientation with the PCD generated by the commercial Trimble. Many methods have been proposed for pairwise point cloud alignment [34].

We adopted the popular iterative closest point [35] registration algorithm variant called generalized iterative closest point [36] with an initial optimized step.

(**a**) (**b**)

Figure 6. Acquired 3D PCD: (**a**) preprocessed, and (**b**) partial 3D PCD.

4.6. Virtual Retrofitting and Efficient Visualization

The proposed gaze-controlled virtual retrofitting method allowed the decision maker to visualize and analyze the retrofit by interacting with the VE. The immersive visualization setup used an HTC Vive HMD, as shown in Figure 7, with a Pupil eye tracker. Estimated gaze values produced by the eye tracker were used as inputs to control PCD interactions. The Pupil eye tracker provided real-time data the IPC backbone messaging bus, which ran as a thread in the main process and allowed messages to push to it and could also subscribe to other actors' messages. Therefore, the IPC formed the backbone for all communication from, to, and within Pupil apps.

Figure 7. HTC Vive and Pupil eye tracker inside the HMD.

We selected the water treatment plant at the Korea Institute of Construction Technology for the experimental study, as shown in Figure 8a, with various pipe diameters as shown in Figure 8b. The water treatment plant can currently provide constant of water flow in the pipeline but needs to be upgraded to increase the water supply.

Figure 9 shows the predefined CAD Model 1 from AutoCAD for virtual retrofitting increased water flow efficiency, whereas Model 2 reduced pipe complexity and the time taken for the same total water flow, as shown in Figure 10.

Figure 11 shows the preprocessed PCD rendered view on the HMD, implemented using C++ and the visualization toolkit, an open source software system [37]. Figures 12 and 13 show the retrofitted PCD with Models 1 and 2, respectively.

(**a**)

(**b**)

Figure 8. Experimental setup: (**a**) water treatment plant at Korea Institute of Construction Technology; (**b**) pipe diameters considered for retrofitting.

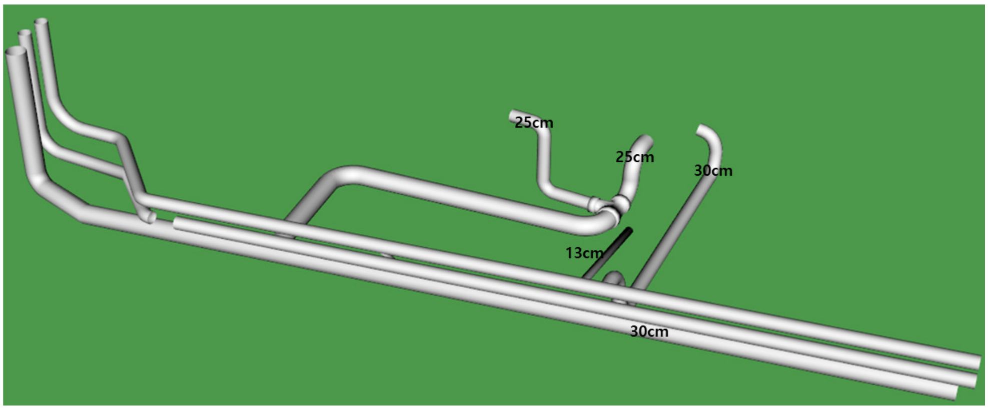

Figure 9. Predefined CAD model 1 with T joint to increase efficiency.

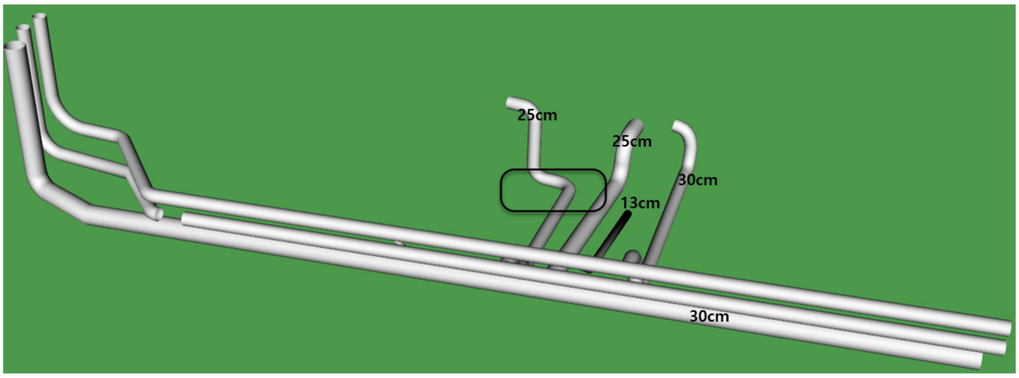

Figure 10. Predefined CAD model 2 with L joint to reduce pipe complexity.

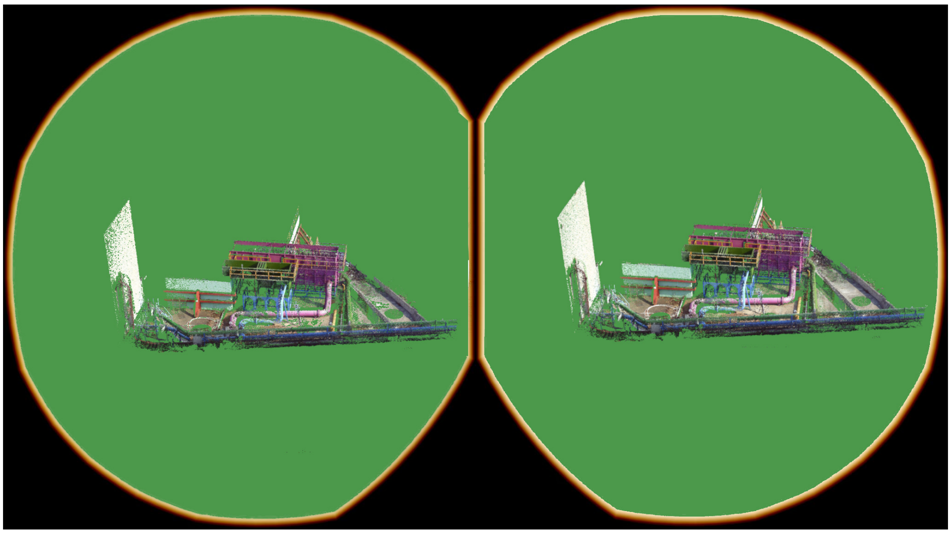

Figure 11. PCD view in the HMD.

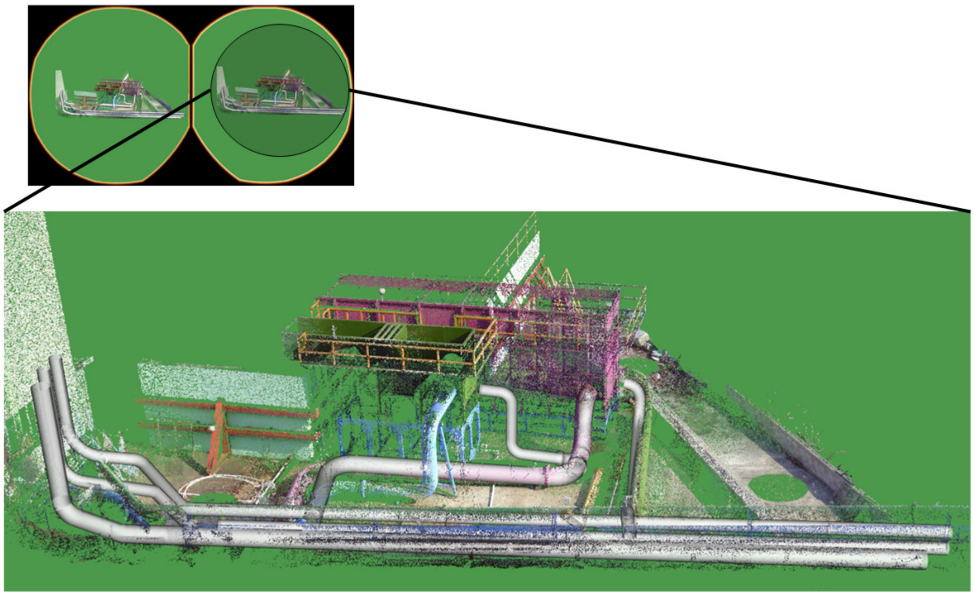

Figure 12. HMD view of retrofitted PCD with Model 1.

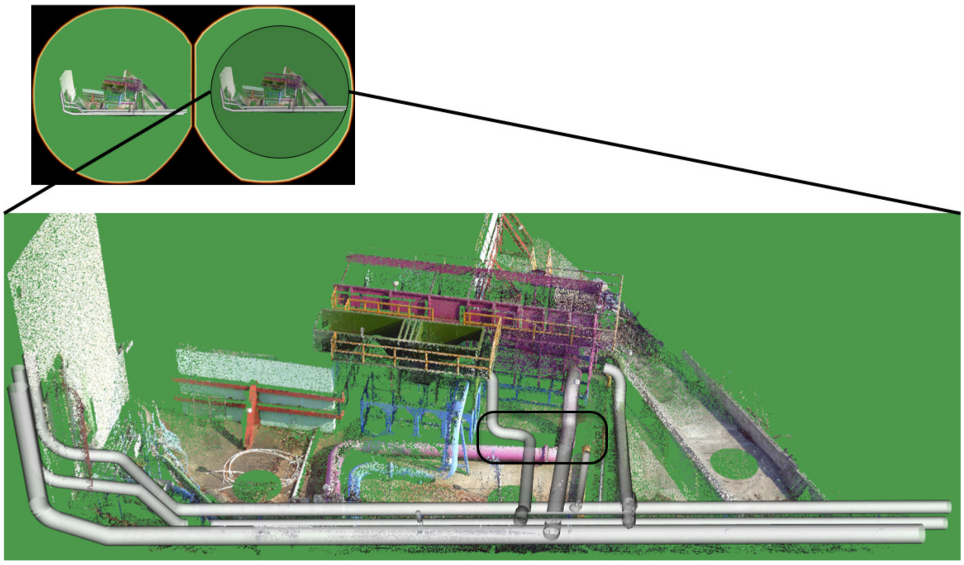

Figure 13. HMD view of retrofitted PCD with Model 2.

4.7. Usability Study

To investigate the speed and the usability of the proposed system, we conducted a user study with five participants (two female and three male) aged between 25 and 30 years with corrected and normal vision. The CAD model was placed 27.35 cm from the rendered PCD, randomly in the x, y plane. Participants were asked to move the CAD model and retrofit with the PCD using the mouse and gaze-controlled interactions in separate experiments.

The Pupil eye tracker was calibrated individually for each participant prior to commencing the experiment, in order to compensate for any participant's myopia. Ten iterations of mouse and gaze-controlled interaction retrofitting were conducted for each participant, providing a total of 100 experiments. Table 3 shows that speed and ease of interaction were significantly improved (25%) using the gaze-controlled interaction compared with mouse interaction retrofitting.

Table 3. Speed and usability results.

Method	Average Time Taken for Retrofitting in Milliseconds				
	P1	P2	P3	P4	P5
Mouse interaction-based	1970	2260	2190	2020	2560
Gaze-controlled	1490	1790	1480	1560	1820
Difference in speed	480	470	710	460	740
Average increase in speed			**572**		

4.8. VE Immersion

We adapted an appropriate questionnaire from Witmer et al. [38] to analyze immersion, interface quality, and accuracy for the proposed system, comprising the following questions where participants rated their responses from 1 (very bad) to 10 (very good).

1. Immersion for the proposed system.
   - How involved were you in the virtual environment experience?
   - Were you involved in the experimental task to the extent that you lost track of time?
   - Were there moments during the virtual environment experience when you felt completely focused on the task or environment?
2. Interface quality.
   - How much were you able to control events?
   - How helpful was the gaze based interfere in performing the assigned tasks?
3. Accuracy.
   - How well could you move or manipulate objects in the virtual environment?
   - Were you able to anticipate what would happen next in response to the actions you performed?

Figure 14 shows the questionnaire responses. There was a linear correlation between participant immersion within the VE and accuracy of the tasks performed.

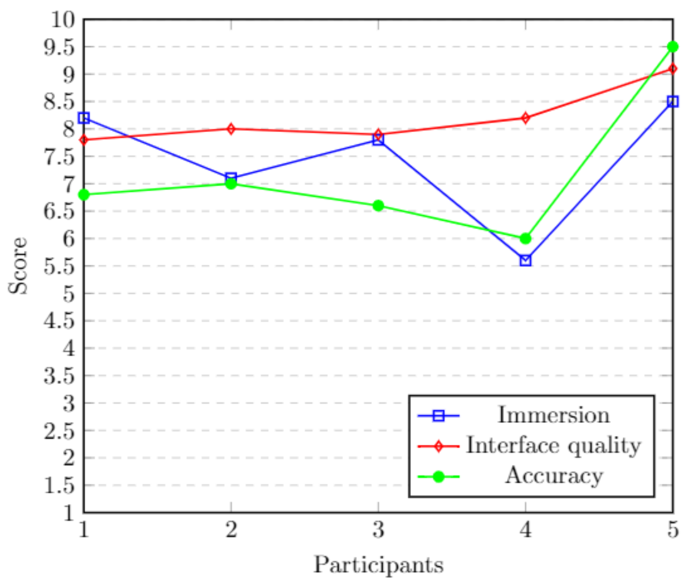

Figure 14. First experiment participant questionnaire results.

4.9. User Learning

The proposed system was tested with a different five participants (four female and one male) aged between 25 and 30 years. These participants were trying the proposed system for the first time, but had information regarding how the system worked. They were asked to perform five retrofitting iterations, and then performed the same questionnaire as in the previous experiment. Figure 15 shows that participant retrofitting accuracy increased, time taken decreased, and immersion level decreased over the iterations. This outcome was to be expected as the participants became more acquainted with the system with every iteration. The time taken to retrofit could be considered a direct measure of participant learning, as shown in Figure 16.

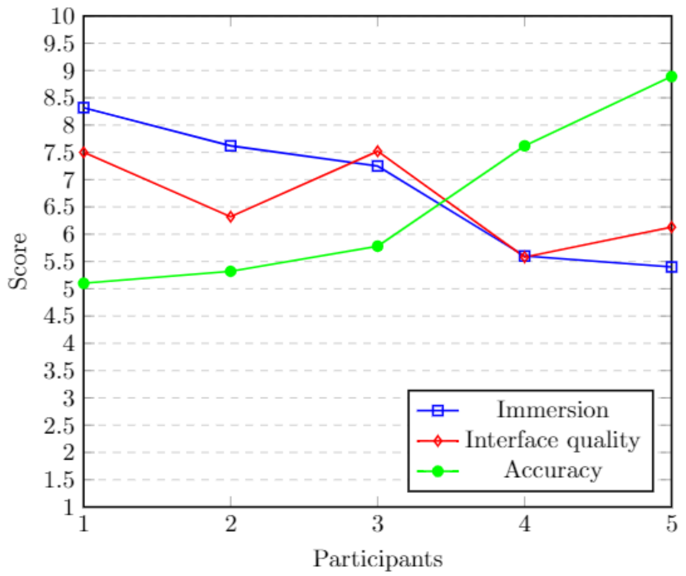

Figure 15. Second experiment participant questionnaire results.

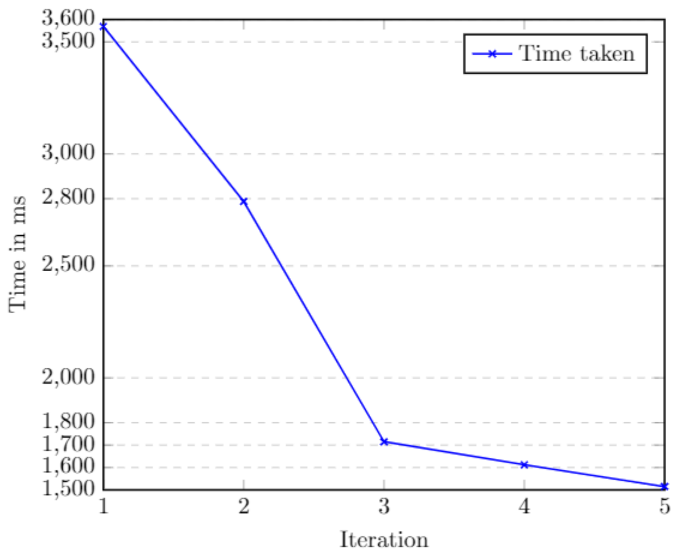

Figure 16. User learning.

4.10. User Cybersickness

We used an appropriate questionnaire from Kennedy et al. [39] to measure participant cybersickness regarding the proposed system, including general discomfort, fatigue, headache, eye strain, and difficulty concentrating. Participants were asked to respond as none, slight, moderate, and severe. Figure 17 shows the questionnaire results.

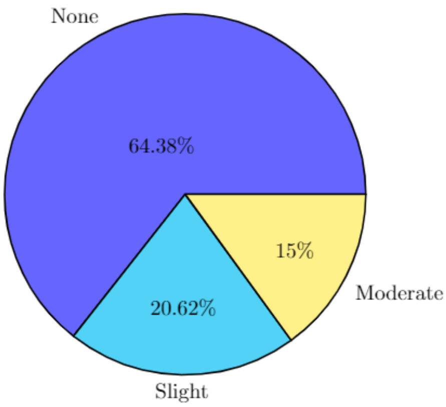

Figure 17. Experiment 2 participant cybersickness.

5. Conclusions

Retrofitting existing pipelines in plants is challenging due to critical defects in unidentified complex objects, which are factors of risk for field working operators. This paper proposed a framework for virtual retrofitting of industrial pipeline plants using eye trackers to estimate the user's gaze for interaction with a VE. The gaze-controlled interaction efficiently assisted with modification and upgradation of existing facilities.

Alignment of the pre-processed partial PCD direct from the from LiDAR provided accurate global coordinate system positioning, which ensured precise 3D CAD model retrofitting. The HMD allowed efficient visualization to retrofit the physical plant in the VE before onsite implementation.

The Pupil eye tracker employed for this study has some limitations. Although it had good accuracy immediately after calibration ($\approx$1.5°), calibration was required every time the tracker was used, making it difficult to test. The eye cameras often heated up, causing jitters in the gaze data. We did not attempt to measure this error, rather we just re-calibrated the Pupil every time this occurred. Since the application employs gaze direction and eye blinks for interaction, precise gaze point convergence wasn't so critical and we were able to achieve acceptable results (to the users) with the Pupil eye tracker, which was the cheapest and best solution.

The PCD density depends on the depth of the scanning environment, and the PCD must be refreshed at regular intervals in the renderer. Thus, a graphics processing unit able to handle large PCDs is essential. Environmental factors, including wind speed and temperature, affect UAV stability while acquiring the PCD. These effects could be improved by incorporating inertial measurement sensors for better orientation accuracy. The current study performed retrofitting offline procedure using the preprocessed PCD and predefined CAD models. Future work will extend this process to real time.

We also intend to investigate and implement gaze based user interaction methods to increase interaction speed and reduce user cybersickness. A parallel auditory system could also be added to increase immersion. Closer investigation of the interaction method will identify areas that could improve user satisfaction, immersion, and reduce cybersickness.

The proposed system could be implemented alongside other VE applications, such as industrial design, interior design, and gaming.

Author Contributions: Conceptualization and Methodology: P.K.B.N. and A.B.; Software, Validation, Formal Analysis, and Investigation: P.K.B.N., A.B., C.B., and A.K.P.; Writing—Original Draft Preparation: P.K.B.N. and A.B.; Writing—Review and Editing: P.K.B.N., and A.B.; Visualization: C.B. and A.K.P.; Data Curation: P.K.B.N., A.B., and A.K.P.; Supervision, Project Administration: A.K.P. and Y.H.C.; Funding Acquisition: Y.H.C.

Funding: This work was supported by the Technology Advancement Research Program funded by the Ministry of Land, Infrastructure and Transport of Korean government and the Ministry of Science, ICT and Future Planning of Korea, under the Software Star Lab. program (IITP-2018-0-00599) supervised by the Institute for Information and communications Technology Promotion (grant 18C TAP-C 132982-02).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations were used in this paper.

PCD	Point cloud data
UAV	Unmanned aerial vehicle
HMD	Head-mounted display
VE	Virtual environment
3D	Three-dimensional

References

1. Houshiar, H.; Winkler, S. Pointo—A low cost solution to point cloud processing. *Int. Arch. Photogramm. Remote Sens. Sp. Inf. Sci.* **2017**, *42*, 111–117. [CrossRef]
2. Patil, A.K.; Kumar, G.A.; Kim, T.H.; Chai, Y.H. Hybrid approach for alignment of a pre-processed three-dimensional point cloud, video, and CAD model using partial point cloud in retrofitting applications. *Int. J. Distrib. Sens. Netw.* **2018**, *14*. [CrossRef]
3. Applications for Laser Scanning. Available online: http://www.vicosoftware.com/trimble-buildings/laser-scanning-for-construction/4-applications-for-laser-scanning (accessed on 31 January 2018).
4. Macrae, C.N.; Hood, B.M.; Milne, A.B.; Rowe, A.C.; Mason, M.F. Are you looking at me? Eye gaze and person perception. *Psychol. Sci.* **2002**, *13*, 460–464. [CrossRef] [PubMed]
5. Sibert, L.E.; Jacob, R.J. Evaluation of eye gaze interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, The Hague, The Netherlands, 1–6 April 2000; pp. 281–288.
6. Bernardini, F.; Rushmeier, H. The 3D model acquisition pipeline. *Comput. Graph. Forum* **2002**, *21*, 149–172. [CrossRef]
7. Green, D.R.; Gomez, C. Small-scale airborne platforms for oil and gas pipeline monitoring and mapping. In Proceedings of the 5th Marine and Coastal Environments Conference, San Diego, CA, USA, 5–7 October 1998.
8. Mavely, A.G.; Judith, J.E.; Sahal, P.A.; Kuruvilla, S.A. Eye gaze tracking based driver monitoring system. In Proceedings of the 2017 IEEE International Conference on Circuits and Systems (ICCS), Thiruvananthapuram, India, 20–21 December 2017; pp. 364–367.
9. Morimoto, C.H.; Mimica, M.R. Eye gaze tracking techniques for interactive applications. *Comput. Vis. Image Understand.* **2005**, *98*, 4–24. [CrossRef]
10. Bruder, G.; Steinicke, F.; Nuchter, A. Poster: Immersive point cloud virtual environments. In Proceedings of the 2014 IEEE Symposium on 3D User Interfaces (3DUI), Minneapolis, MN, USA, 29–30 March 2014; pp. 161–162.
11. Burwell, C.; Jarvis, C.; Tansey, K. The potential for using 3D visualization for data exploration, error correction and analysis of LiDAR point clouds. *Remote Sens. Lett.* **2012**, *3*, 481–490. [CrossRef]
12. MacEachren, A.M.; Edsall, R.; Haug, D.; Baxter, R.; Otto, G.; Masters, R.; Fuhrmann, S.; Qian, L. Virtual environments for geographic visualization: Potential and challenges. In Proceedings of the 1999 Workshop on New Paradigms in Information Visualization and Manipulation in Conjunction with the Eighth ACM Internation Conference on Information and Knowledge Management, Kansas City, MO, USA, 2–6 November 1999; pp. 35–40.
13. Kassner, M.; Patera, W.; Bulling, A. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, Seattle, WA, USA, 13–17 September 2014; pp. 1151–1160.
14. Dempsey, P. The teardown: HTC Vive VR headset. *Eng. Technol.* **2016**, *11*, 80–81.
15. Mine, M.R. *Virtual Environment Interaction Techniques*; Technical Report; University of North Carolina: Chapel Hill, NC, USA, 1995.

16. Franchak, J.M.; Kretch, K.S.; Soska, K.C.; Adolph, K.E. Head-mounted eye tracking: A new method to describe infant looking. *Child Dev.* **2011**, *82*, 1738–1750. [CrossRef] [PubMed]
17. Morimoto, C.H.; Amir, A.; Flickner, M. Detecting eye position and gaze from a single camera and 2 light sources. In Proceedings of 16th International Conference on Pattern Recognition (ICPR), Quebec City, QC, Canada, 11–15 August 2002; p. 40314.
18. Corcoran, P.M.; Nanu, F.; Petrescu, S.; Bigioi, P. Real-time eye gaze tracking for gaming design and consumer electronics systems. *IEEE Trans. Consum. Electron.* **2012**, *58*, 347–355. [CrossRef]
19. Tanriverdi, V.; Jacob, R.J. Interacting with eye movements in virtual environments. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, The Hague, The Netherlands, 1–6 April 2000, pp. 265–272.
20. Piumsomboon, T.; Lee, G.; Lindeman, R.W.; Billinghurst, M. Exploring natural eye-gaze-based interaction for immersive virtual reality. In Proceedings of the 2017 IEEE Symposium on 3D User Interfaces (3DUI), Los Angeles, CA, USA , 18–19 March 2017; pp. 36–39.
21. Tripathi, S.; Guenter, B. A statistical approach to continuous self-calibrating eye gaze tracking for head-mounted virtual reality systems. In Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, USA, 24–31 March 2017; pp. 862–870.
22. Higuch, K.; Yonetani, R.; Sato, Y. Can Eye Help You?: Effects of Visualizing Eye Fixations on Remote Collaboration Scenarios for Physical Tasks. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; pp. 5180–5190.
23. Kato, H.; Billinghurst, M.; Poupyrev, I.; Imamoto, K.; Tachibana, K. Virtual object manipulation on a table-top AR environment. In Proceedings of the IEEE and ACM International Symposium on Augmented Reality (ISAR 2000), Munich, Germany, 5–6 October 2000; pp. 111–119.
24. Bowman, D.A.; Hodges, L.F. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In Proceedings of the 1997 Symposium on Interactive 3D Graphics, Providence, RI, USA , 27–30 April 1997.
25. Akinci, B.; Boukamp, F.; Gordon, C.; Huber, D.; Lyons, C.; Park, K. A formalism for utilization of sensor systems and integrated project models for active construction quality control. *Automat. Constr.* **2006**, *15*, 124–138. [CrossRef]
26. Woo, J.H.; Menassa, C. Virtual retrofit model for aging commercial buildings in a smart grid environment. *Energy Build.* **2014**, *80*, 424–435. [CrossRef]
27. Thomson, C.; Boehm, J. Automatic geometry generation from point clouds for BIM. *Remote Sens.* **2015**, *7*, 11753–11775. [CrossRef]
28. Bergé, L.P.; Aouf, N.; Duval, T.; Coppin, G. Generation and VR visualization of 3D point clouds for drone target validation assisted by an operator. In Proceedings of the 8th Computer Science and Electronic Engineering (CEEC), Colchester, UK, 28–30 September 2016; pp. 66–70.
29. Kim, J.; Sunil Kumar, Y.; Yoo, J.; Kwon, S. Change of Blink Rate in Viewing Virtual Reality with HMD. *Symmetry* **2018**, *10*, 400. [CrossRef]
30. Matrice, D.J.I. 100, 2016. Available Online: https://store.dji.com/product/matrice-100 (accessed on 28 November 2018).
31. Velodyne Puck LITE, VLP 16. Available Online: https://velodynelidar.com/vlp-16-lite.html (accessed on 28 November 2018).
32. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source Robot Operating System. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; p. 5.
33. Carvalho, J.P.; Jucá, M.A.; Menezes, A.; Olivi, L.R.; Marcato, A.L.M.; dos Santos, A.B. Autonomous UAV outdoor flight controlled by an embedded system using Odroid and ROS. In *CONTROLO 2016*; Garrido, P., Soares, F., Moreira, A., Eds.; Springer: Cham, Germany, 2017; pp. 423–437.
34. Tam, G.K.; Cheng, Z.Q.; Lai, Y.K.; Langbein, F.C.; Liu, Y.; Marshall, D.; Martin, R.R.; Sun, X.F.; Rosin, P.L. Registration of 3D point clouds and meshes: A survey from rigid to nonrigid. *IEEE Trans. Vis. Comput. Gr.* **2013**, *19*, 1199–1217. [CrossRef] [PubMed]
35. Besl, P.J.; McKay, N.D. Method for registration of 3-D shapes. In Proceedings of the Sensor Fusion IV: Control Paradigms and Data Structures, Boston, MA, USA, 12–15 November 1991; pp. 586–607.
36. Segal, A.; Haehnel, D.; Thrun, S. Generalized-ICP. *Robot. Sci. Syst.* **2009**, *2*, 435.

37. SSchroeder, W.J.; Lorensen, B.; Martin, K. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, 3rd ed.; Kitware: Clifton Park, NY, USA, 2004.
38. Witmer, B.G.; Jerome, C.J.; Singer, M.J. The factor structure of the presence questionnaire. *Presence Teleoper. Virtual Environ.* **2005**, *14*, 298–312. [CrossRef]
39. Kennedy, R.S.; Lane, N.E.; Berbaum, K.S.; Lilienthal, M.G. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *Int. J. Aviat. Psychol.* **1993**, *3*, 203–220. [CrossRef]