

Article

A Novel Real-Time Match-Moving Method with HoloLens

Jaehyun Lee ¹, Jahanzeb Hafeez ¹ , Kwangjib Kim ¹, Seunghyun Lee ² and Soonchul Kwon ^{3,*}

¹ Department of Plasma Bio Display, Kwangwoon University, Seoul 01897, Korea

² Ingenium College, Kwangwoon University, Seoul 01897, Korea

³ Graduate School of Smart Convergence, Kwangwoon University, Seoul 01897, Korea

* Correspondence: ksc0226@kw.ac.kr; Tel.: +82-2-940-8637

Received: 11 June 2019; Accepted: 17 July 2019; Published: 19 July 2019



Abstract: With the advancement of media and computing technologies, video compositing techniques have improved to a great extent. These techniques have been used not only in the entertainment industry but also in advertisement and new media. Match-moving is a cinematic technology in virtual-real image synthesis that allows the insertion of computer graphics (virtual objects) into real world scenes. To make a realistic virtual-real image synthesis, it is important to obtain internal parameters (such as focal length) and external parameters (position and rotation) from an Red-Green-Blue(RGB) camera. Conventional methods recover these parameters by extracting feature points from recorded video frames to guide the virtual camera. These methods fail when there is occlusion or motion blur in the recorded scene. In this paper, we propose a novel method (system) for pre-visualization and virtual-real image synthesis that overcomes the limitations of conventional methods. This system uses the spatial understanding principle of Microsoft HoloLens to perform the match-moving of virtual-real video scenes. Experimental results demonstrate that our system is much more accurate and efficient than existing systems for video compositing.

Keywords: image synthesis; mixed reality; match-moving; pre-visualization; HoloLens

1. Introduction

Virtual-real synthesis techniques are used for video production in the media industry because they are able to express many images that cannot be expressed in the real world. These techniques can deliver a realistic experience as well as reduction in production cost. Therefore, they have attracted the attention of many researchers. In general, virtual-real synthesis refers to the insertion of computer graphics into a real world scene with precise position, scale, and rotation information. This technique is called match-moving. This technique works by extracting internal parameters (such as focal length) and external parameters (position and rotation) from an RGB camera used to capture a real scene [1,2]. From these parameters, we can retrieve the motion of the RGB camera, which is called camera tracking [3]. As the camera tracking process belongs to post-production and takes a long time to extract a camera trajectory from an RGB camera, consequently, it increases the production cost and time. Some factors, including motion blur caused by fast camera movement and occlusion by a person or object, can affect camera tracking results. These factors result in bad feature-matching between video frames, which results in poor or no virtual-real synthesis. Due to these errors, sometimes a scene must be shot again, which wastes time and money. Therefore, an accurate virtual-real synthesis system is required to reduce production time and cost.

In this study, we implemented a novel system consisting of Microsoft HoloLens and an RGB camera. This system works based on spatial understanding by scanning the environment in real time, which gives us the position and rotation information used for camera tracking. Our system differs

from conventional systems because it does not need to extract feature points, which are later used for camera tracking, from the recorded video. Additionally, we built a pre-visualization system that can verify the match-moving results while recording by displaying the real and virtual objects in the scene.

In Section 2, we present the existing match-moving method and HoloLens related works. In Section 3, we present the proposed system. Section 4 presents experimental results that compare the proposed match-moving system with existing methods and pre-visualization performance test results. Finally, Section 5 includes conclusions about the proposed match-moving system and future research plans.

2. Background Theory

2.1. Match-Moving

Match-moving uses numerical algorithms that can estimate position information by searching feature points from bright spots, dark spots, and corners. Typical examples are simultaneous localization and mapping (SLAM) [4–6] and structure-from-motion (SfM) [7,8]. These methods extract the external parameter of the camera. There are many commercialized programs in the film industry, including Boujou [9], PFtrack [10], and Match-Mover. Figure 1 shows the work-flow of Boujou, which is a feature point-based method and a common match-moving program.

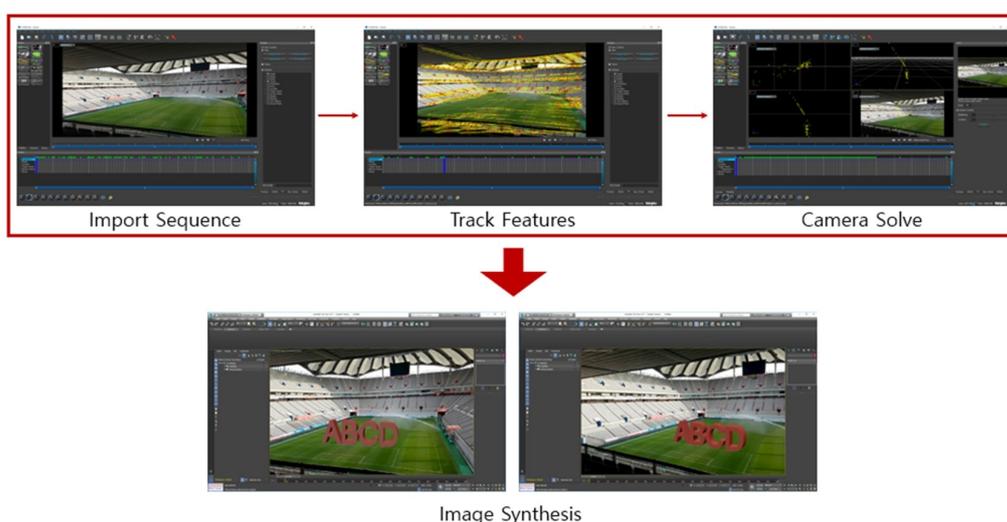


Figure 1. The match-moving process using the Boujou method creating image synthesis by extracting feature points and camera solve from video frames.

The match-moving programs mentioned above have two main processes for virtual-real image synthesis. First, there is “track features”, which is a process of extracting the feature points from recorded images. Secondly, “camera solve” is a process of tracking transformation information of the same feature points in every recorded frame to create the external parameter of the camera. Then, the obtained external parameter can be created as a virtual camera in a 3D program such as 3ds Max.

Most match-moving programs require stable camera movement or a stable environment (i.e., they require a cross pattern that is able to find many feature points more easily) in order to obtain accurate feature points. However, if motion blur by fast camera movement or occlusion by a large object is included in the recorded video like in Figure 2a,b, feature tracking will fail. Therefore, “camera solve” also cannot be generated properly. As a result, accurate virtual-real image synthesis is disturbed and much time and many human resources are wasted.

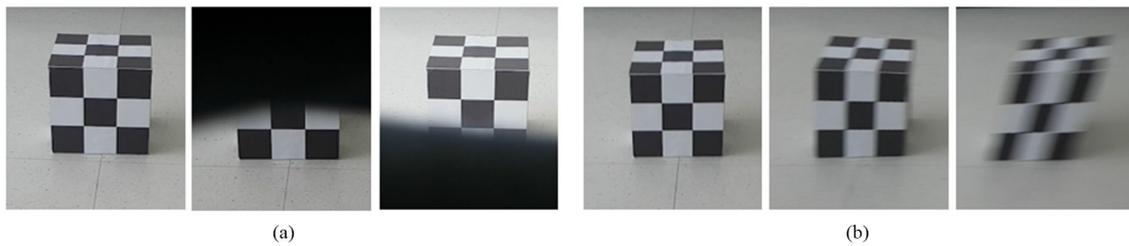


Figure 2. Common reasons for match-moving errors in traditional methods. (a) occlusion and (b) motion-blur cause problems in feature tracking.

2.2. HoloLens

HoloLens is an Head-Mounted Display(HMD) type mixed reality (MR) device developed by Microsoft in 2016 under the name Project Baraboo. It has its own CPU, GPU, and holographic processing unit, which eliminates the need for external computing devices. In addition, it has an inertial measurement unit (IMU), which includes an accelerometer, gyroscope, magnetometer, four environment cameras, a 2 MP RGB camera, and a Time Of Flight(ToF) camera. Therefore, it can obtain accurate data with spatial understanding and reconfiguration, head tracking, and limited gesture recognition. HoloLens understands and reconstitutes the space based on the ToF camera and the Kinect Fusion algorithm from the real environment [11]. It can also estimate the real-time head tracking value (position) using IMU and the iterative closest point (ICP) algorithm [12]. Therefore, a holographic model can be placed at the correct position by processing and calculating spatial information and the user's position information [13].

There have been many studies on the accuracy analysis of HoloLens. Vassallo presented an evaluation protocol for the accurate position maintenance of rendered holograms using optical measurement devices [14]. Liu compared HoloLens with OptiTrack to analyze its accuracy [12]. In this study, we used the real-time head tracking value of HoloLens as the external parameter value of the mounted RGB camera. In addition, we convert to Max script to make a virtual camera in 3ds Max. Finally, we compare our proposed method with that of Boujou.

2.3. Related Works

Similar to our study, several studies have been conducted to extract the external parameter of an RGB camera and use it for match-moving. One such method proposed by Lee et al. was used with a low-pass filter to smoothly correct the tracking data that generated the virtual camera's transform x , y , and z values of every frame from recorded images [15]. In addition, Kim et al. mounted the main camera and the secondary camera firmly using and calculated the relative position value between the two cameras using Kumar's correction method [16]. Thereafter, the two cameras record images to extract the external parameters, and that recorded images from each camera are optimized to generate a stable camera trajectory. When feature point loss occurs due to dynamical camera movement, it can connect the trajectory by calculating the cross-correlation between the feature point of the two camera images through the missing feature point trajectory connection algorithm [17].

Similarly, Yokochi et al. proposed a method of estimating the camera's external parameters using the feature point from the recorded images and the GPS position data as match-moving. It used the structure from motion (SfM) algorithm for feature point tracking and used GPS data-based correction to estimate errors caused by various factors [18].

Recently, Luo et al. proposed a match-moving method and real-time 3D preview visualization system using the Kinect Fusion algorithm. They used structure light Kinect1 and two RGB cameras to extract a depth map and track the camera pose and trajectory. However, this method was limited in terms of precision. When the surrounding environment or objects moved, noise was generated, which lowered tracking accuracy. Therefore, there is a disadvantage that the surrounding environment must be fixed or an additional noise removing step is required [19].

Similarly, our previous work reported on a real-time optimized mapping environment for real-time spatial mapping of HoloLens [20] and studied the building of a virtual studio system for broadcasting. Existing Microsoft Spectator View [21] causes network delay and frame breaks due to the number of polygons (or vertices) in the 3D modeling file. In order to use it as a broadcasting studio, we proposed a method of using only the head transform of HoloLens [22]. In this paper, we propose a stabilized match-moving system and a pre-visualization system using HoloLens based on these previous studies.

3. Proposed System

In this paper, we propose a novel system that uses Microsoft HoloLens to track the position precisely for match-moving techniques. The proposed system also includes a pre-visualized system that can be beneficial to preview the result of the virtual-real synthesis while recording the video. Figure 3 demonstrates the complete workflow of the proposed system.

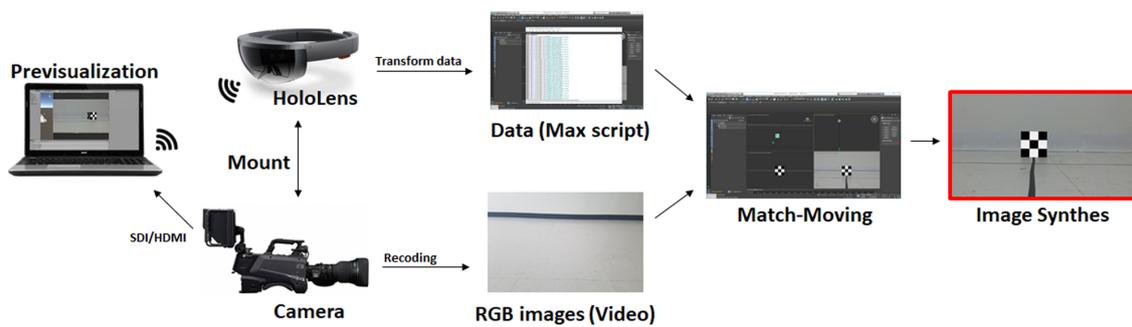


Figure 3. Novel system for real-time match-moving and pre-visualization with HoloLens.

3.1. Real-Time Match-Moving with HoloLens

The proposed system uses HoloLens and an RGB camera mounted on a rig, which is connected to a PC. In this system, the HoloLens is configured such that it looks in the same direction as the RGB camera. The RGB camera is used to record the videos of the real scene, and, at the same time, HoloLens accumulates the external parameters of the RGB camera in 3D space. In order to ensure the precision of the RGB camera’s tracking by HoloLens, both devices are calibrated using Zhang’s calibration method [23]. For this purpose, images of a checkerboard pattern are captured by both the RGB and HoloLens cameras mounted on the rig and are further processed to extract the internal and external parameters as shown in Figure 4. These parameters are used to estimate the relative position information of both devices by comparing the images captured by each camera. These parameters are also used to estimate the internal parameter of the RGB camera for setting up the virtual camera later.



Figure 4. Camera calibration process for extracting RGB camera internal parameters and acquiring distance difference between RGB camera and HoloLens camera. (a) RGB camera picture; (b) HoloLens camera picture.

For better performance, it is necessary to scan the environment using a process called spatial mapping to accurately track the HoloLens position. At first, when the HoloLens turns on, it scans the environment automatically using its built-in ToF and environment cameras and stores its position information [20]. In order to improve the accuracy of the spatial mapping, the number of triangular meshes per cubic meter can be set in units of 100, 500, 750, and 2000. However, as the number of meshes increases, more data processing is required. In our system, we used 500 as the average value based on trial and error. Before starting the recording process, it is recommended to perform the spatial mapping by scanning the surrounding environment.

The external parameters obtained from HoloLens together with the RGB camera should be acquired in the 3ds Max file scripting language (.ms) to create a virtual camera in 3D Max. There are two methods to generate this file. In the first method, the 3D position information of the scene is acquired from HoloLens using a sharing server via a network connection [24]. However, this method can be affected by an unstable network connection which can cause errors in the position and rotation data acquired by HoloLens. In the second method, this information can be stored in the HoloLens locally, i.e., store the position and rotation information inside HoloLens and transfer to PC afterwards. In the proposed system, we have adopted the second method and acquired 3D position information as XYZ coordinates together with the RGB camera. Subsequently, the position information obtained from HoloLens was imported into 3ds Max to create a virtual camera for virtual-real image synthesis.

The RGB camera was set to record video at 30 frames per second (fps) and to synchronize the HoloLens with RGB camera, 0.033 s time was set in the HoloLens to save 3D position information for each frame. In addition, the measuring system of HoloLens is different to that of 3ds Max. Therefore, to synchronize the position information obtained from the HoloLens, 3ds Max's unit system is converted into the HoloLens unit system, where one unit is equal to one meter. When the Max script is loaded, it creates a virtual camera based on the internal parameters extracted from the RGB camera and external parameters from the HoloLens, as shown in Figure 5.

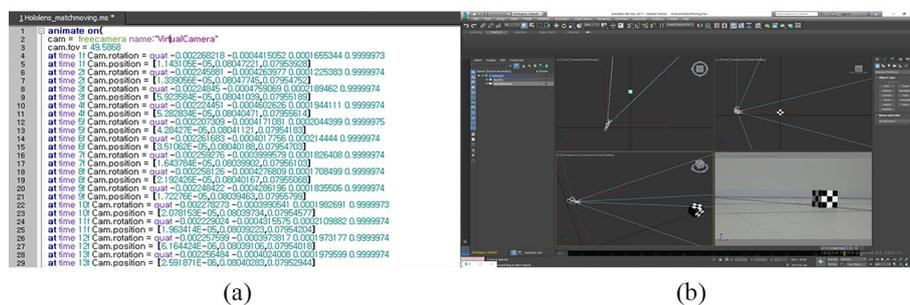


Figure 5. Max script generated from HoloLens and 3ds Max program. Synchronization of unit and coordinator system is required for using HoloLens data to 3ds Max. (a) Max script; (b) 3ds Max.

3.2. Pre-Visualization System

The pre-visualization in the proposed system is used to preview the virtual-real image synthesis while recording the video. It uses both the HoloLens and RGB camera to display the virtual objects in a real scene. For this purpose, we used Unity Game Engine, which synchronizes the scene information from both devices using the spatial understanding of HoloLens in real time. The 3D position information used to pre-visualize the scene is transferred to the PC via Transmission Control Protocol/Internet Protocol (TCP/IP) protocol wirelessly, which provide the 3D position information to virtual camera at 30 fps. Consequently, it helps to prevent the errors by pre-visualizing the image synthesis result and reduces the time required for video composition in a later process.

Currently, there are some systems that can be used for live streaming through HoloLens and an RGB camera. These systems include Spectator View and mixed reality view compositing (MRVC) [25] and can display virtual object (images) in a real scene in real time. However, these systems face

difficulties in live streaming when high polygon 3D models are uploaded from the server. In order to solve these problems, we previously proposed and developed Holo-Studio (Seoul, Korea) [22]. This system was built to use in a broadcasting virtual studio that was capable of loading high-polygon 3D models at higher frame rates and with lower network speeds. The main difference between MS SV and Holo-Studio is the exchange of object transformation information between HoloLens and the server. This information is exchanged in bidirectional and unidirectional modes in the former and latter case, respectively, as shown in Figure 6.

In this work, we propose a pre-visualization system similar to that of our previous system with some melioration. For pre-visualization only, in fact, we do not need to share and exchange the objects' transformation information with other HoloLens devices or the server. We only require the position information of the HoloLens mounted on the RGB camera. This position information is sufficient to display the virtual objects in a real scene. This melioration has two advantages. Firstly, we do not need to share the objects and their transformation information with other devices which were causing the delay in pre-visualization due higher number of polygons—for instance, in MS SV. Therefore, eliminating the process of object's transformation information exchanged with other devices would make it possible to use a virtual object of any number of polygons. Secondly, as the position information from the HoloLens is acquired in the form of XYZ coordinates, we can set the frame rate to be the same as that of the RGB camera. This prevents the issue of frame delay.

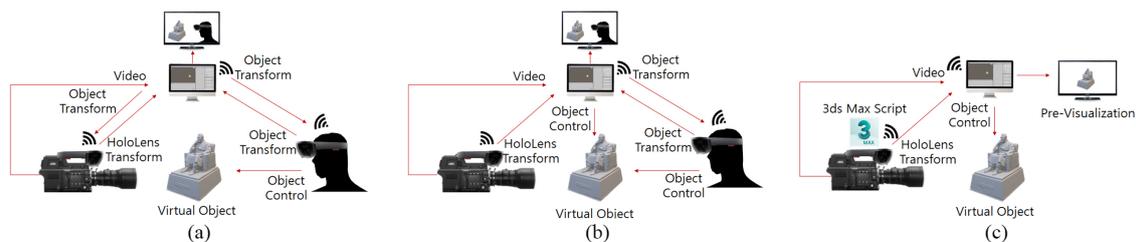


Figure 6. Comparison of HoloLens stream systems: (a) Spectator View [21]; (b) Holo-Studio [22]; (c) proposed method.

3.3. Experimental Environment

In order to measure the accuracy of proposed and traditional (Boujou) methods in terms of match-moving, we compared them against the ground truth. It is well-known fact that the match-moving results highly depend on the accuracy of camera trajectory [19]. In order to measure the accuracy of camera trajectory extracted from both the proposed and traditional methods, we compared them against the Oculus Rift's camera trajectory as ground truth. Previously, Borrego et al. [26] has evaluated the performance of Oculus Rift's head tracking accuracy (also known as camera trajectory) and reported with average values ranging from 0.6 to 1.2 cm. On the other hand, to measure the accuracy of virtual-real image synthesis, both the proposed and Boujou methods are evaluated against a real-world scene in two different scenarios.

3.4. Accuracy Evaluation of Camera Trajectory

Oculus Rift has high accuracy in terms of position trajectory because it performs tracking using two infrared sensors that work together to locate the position of VR headset within an area of 2 m² [26,27]. Due to its high accuracy within the sensor range, several studies have used the Oculus as ground truth [28–30]. In this experiment, we used Oculus Rift as ground truth to compare the accuracy of the proposed and Boujou method's camera trajectory. For this purpose, we recorded 27 s video with an RGB camera mounted together with HoloLens and Oculus Rift. While recording the video with the RGB camera, HoloLens and Oculus were also storing the camera trajectory information. First, half of the video was recorded with no occlusion as shown in Figure 7, whereas, the second half of the video

recording an object was used as an obstacle to create occlusion by moving it in front of the RGB camera as shown in Figure 7b.

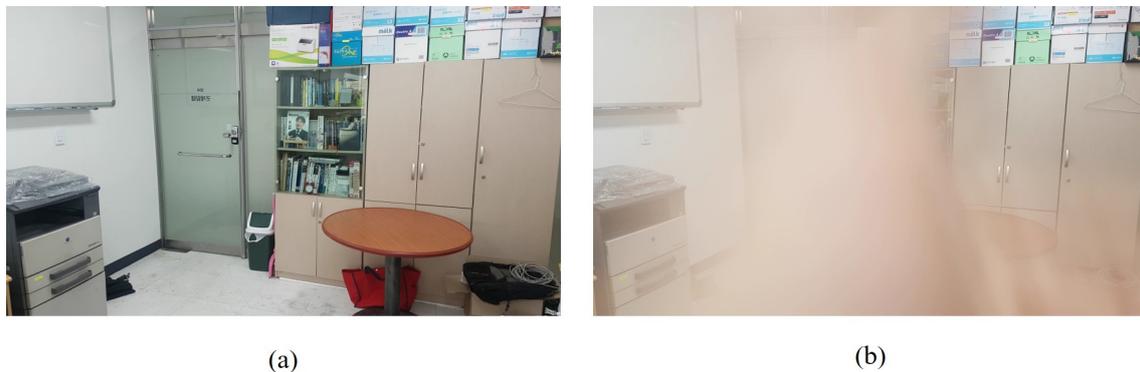


Figure 7. Experimental setup for accuracy evaluation of camera trajectory. (a) experimental space in normal condition; (b) experimental space with occlusion. Oculus was used to record the ground truth for comparing camera trajectory with HoloLens and RGB camera.

3.5. Evaluation of the Match-Moving Results

In order to evaluate the match-moving results in both the proposed and Boujou methods, we performed experiments by placing a cardboard cube marked with a checkerboard pattern in a 2 m wide window and 1.5 m away from the camera-HoloLens system. The scene was captured for a duration of 45 s at a rate of 30 fps. Then, we recoded the video of the cardboard cube in two scenarios with different backgrounds and motion blur using the RGB camera mounted with HoloLens. In the first scenario, the cardboard cube was captured by placing it in front of a white background with cross-patterns in a regular manner. The cross-patterns were pasted on the background wall to facilitate the feature point method, which requires a maximal number of features in the scene to work properly. In the second scenario, we recorded the video of the cardboard cube without cross patterns and by moving the camera fast in order to create some motion blur in the recorded video. Motion blur causes the failure of feature point identification from the recorded video frames, which affects the Boujou method. The camera was moved from the center to the right and then left in a specific motion. We paused and resumed the camera motion at four points as marked in Figure 8. The camera was moved smoothly in a regular manner between positions 1 and 2, and then, from position 2 to 3, it was moved swiftly to create motion blur in the recorded video frames.

The videos recorded in both scenarios were then processed using the Boujou method and the proposed method to synthesize the virtual-real images. First, the videos recorded by RGB camera were imported into Boujou software without any position information acquired from the HoloLens. The Boujou method uses a process to extract the feature points from the recorded video frames. In the next step, these feature points were processed to determine the “camera solve” or in other words the external parameters of the RGB camera, and this information was exported into 3ds Max to create a virtual camera. However, the proposed method can obtain the position information from HoloLens at the same time as recording the videos and store it in a Max script. This information from the HoloLens in the form of a Max script is used to synthesize the virtual-real images in 3ds Max. The synthesizing of virtual-real images was done by creating a virtual camera in 3ds Max. The path of the virtual camera was generated from the Max script obtained from the HoloLens.

In order to test the performance of the proposed system and Boujou under the circumstances of our experimental setup, we created a virtual scene exactly the same as the real scene in 3ds Max. The cardboard cube in the virtual scene is identical to that in the real scene. The point where the camera was placed is considered as the origin in both the real and virtual scenes. In addition, the distance between the camera and the object was exactly the same in both scenes. This compares the concordance rate between the real object and virtual object based on the mismatch pixels in a specific range using

different images. In addition, it compares the external parameter of camera position obtained from each match-moving method.

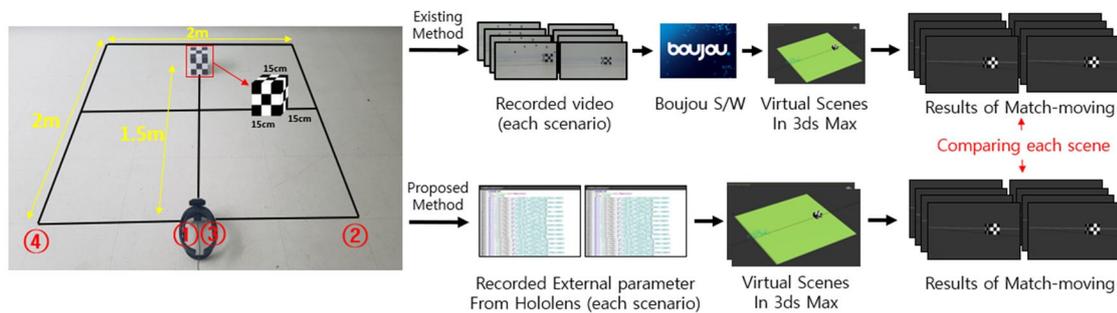


Figure 8. Experimental setup for comparing match-moving results.

4. Experimental Results

4.1. Camera Trajectory

Once the video recording process is completed, it is used to extract the camera trajectory of RGB camera, HoloLens and Oculus Rift. Figure 9a,c shows the camera trajectory extracted from HoloLens in comparison with the ground truth for the first and second half of the video, respectively, whereas Figure 9b,d illustrates the camera trajectory of the RGB camera for the first and second half of the video, respectively. The distance deviations calculated for both HoloLens and Boujou methods in comparison with the ground truth are indicated in Table 1. From Figure 9 and Table 1, it can be seen that the camera trajectory extracted from the HoloLens is almost perfectly aligned with the ground truth. On the other hand, the camera trajectory extracted from the RGB camera using the Boujou method is highly affected by the occlusion recorded in the second half of the video while the first half has only minor differences when compared with the ground truth.

From the experiments, it is clear that occlusion has a great effect in case of traditional methods such as Boujou, whereas the proposed method has no direct effect of occlusion because it only records/stores the position information of the RGB camera and not the video. It implies that the proposed method is better at finding the camera trajectory as compared to the Boujou method.

Table 1. Standard deviation of the comparison of ground truth, HoloLens and Boujou.

	Position X (cm)	Position Y (cm)	Position Z (cm)
Ground truth-HoloLens (1–400 Frame)	0.64	0.57	0.97
Ground truth-Boujou (1–400 Frame)	1.86	1.98	0.98
Ground truth-HoloLens (401–800 Frame)	0.63	0.28	1.03
Ground truth-Boujou (401–800 Frame)	6.85	12.46	49.26

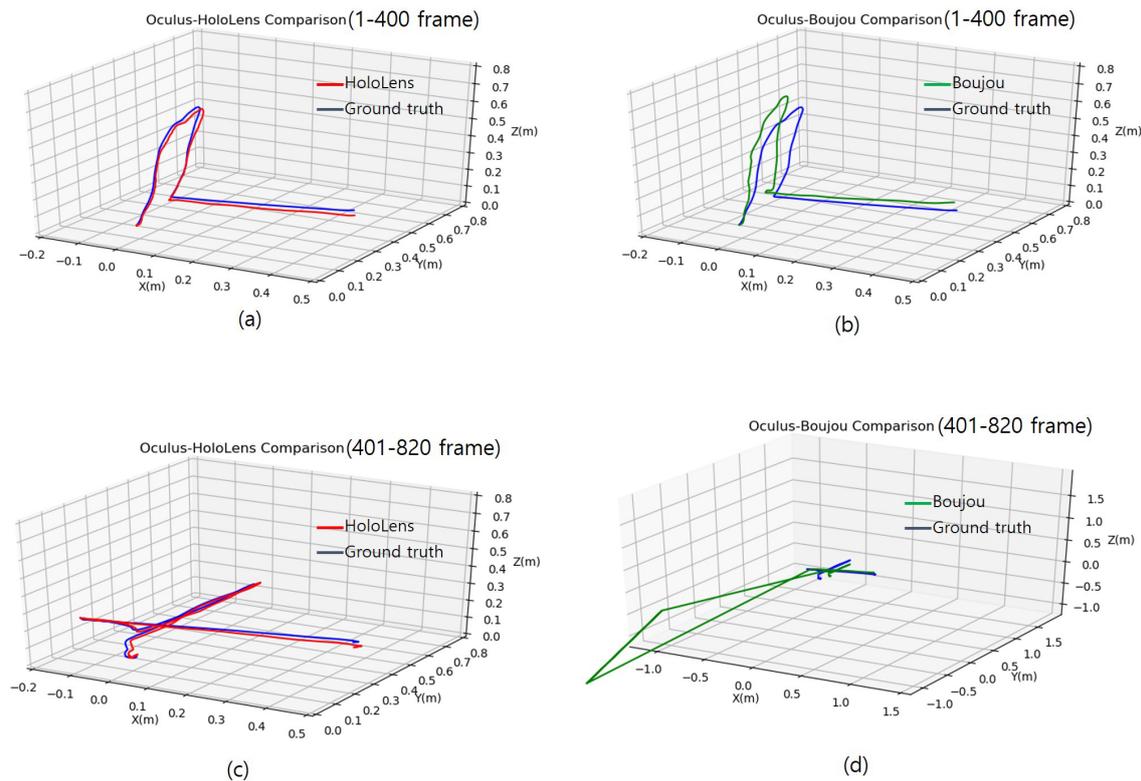


Figure 9. Accuracy evaluation of the camera trajectory extracted from HoloLens (a,c) and Boujou (b,d) method against ground truth (Oculus Rift).

4.2. Match-Moving with Cross Pattern

The videos recorded in both scenarios were synthesized with a virtual scene as explained in the previous section. Furthermore, in order to find the match rate, we found the difference of the image frames extracted from the RGB-camera and the virtual camera. For this purpose, we extracted four frames randomly at the points marked in Figure 8. The frames extracted from the real scene are considered as ground truth and the virtual-real synthesis of the images obtained from both methods are compared, and the difference between the ground truth and synthesized images is calculated. For evaluation purposes, we used the MATLAB software (2018b, MathWorks, MA, USA).

The feature point detection process can obtain stable results from images with edges and corners with high frequency changes. We present the results of match-moving (virtual-real synthesis images) between Boujou and the proposed method under given circumstances for the feature point detection process, as shown in Figure 10. In addition, to observe the match rate, we calculated the image difference between the ground truth and virtual-real synthesized images. As most parts of the compared images are identical, a binary mask was created to separate the region of interest from the background. For our purposes, the region of interest was only the cube in the recorded image frames with a resolution 200×300 pixels. The mismatched pixels between the ground truth and virtual-real synthesized images are illustrated in white and indicated in Table 2.

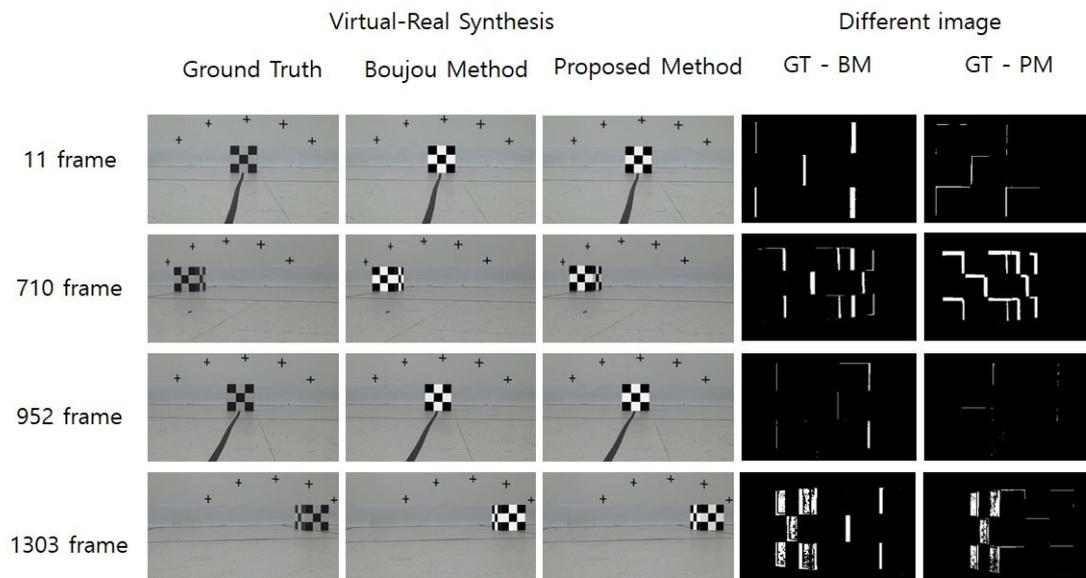


Figure 10. Comparison of different images in match-moving with cross pattern.

The lowest match rate at different frames for the first scenario was 96.77% and 94.38% for the feature point method and proposed method (PM), respectively. If we consider the uncertainty in size and distance between real and virtual cubes from the camera rig, the calculated difference between the ground truth (GT) and the synthesized images can be neglected. Therefore, in the first scenario, both the feature point (BM) and proposed method showed similar performance.

Table 2. Match rate of different images in match-moving with cross pattern.

	Method	Number of Unmatched Pixels	Specified Rate Resolution	Match Rate (%)
11 frame	GT-BM	938	60,000	98.4
	GT-PM	500	60,000	99.1
710 frame	GT-BM	1199	60,000	98.0
	GT-PM	3371	60,000	94.4
952 frame	GT-BM	577	60,000	99.0
	GT-PM	155	60,000	99.7
1303 frame	GT-BM	1936	60,000	96.8
	GT-PM	773	60,000	98.7

4.3. Match-Moving with Motion Blur

As mentioned earlier, factors such as motion blur and occlusion can affect the camera tracking results in the Boujou method. These factors affect the feature point detection process and consequently lose the camera path. Therefore, the virtual-real image synthesis cannot be accurate. This limitation of the Boujou method can be seen in the video recorded in the second scenario when there were no cross patterns in the background and there was motion blur. As the checkerboard pattern is printed on the surface of the cardboard cube, the Boujou method can find feature points from the captured images even when there are no cross patterns in the background. However, the amount of error is increased when compared to the first scenario when cross patterns were added in the background. The difference in the accuracy of virtual-real image synthesis can be seen in Tables 2 and 3.

Due to the abrupt motion of the camera in the second scenario, the video frames between 479 and 542 have motion blur. As a result, the lowest match rate of the Boujou method was 46.41% and 52.41% at 702 frames and 975 frames, respectively, which is indicated in Table 3. This shows that the

match rate dropped significantly when comparing with the results in Table 2. As seen in Figure 11, the virtual-real image synthesis from 703 frames onwards has completely lost the camera path. On the other hand, the proposed method has a match rate of over 94% for the whole video, as shown in Table 3. This means that the proposed method is not affected by any errors caused by motion blur.

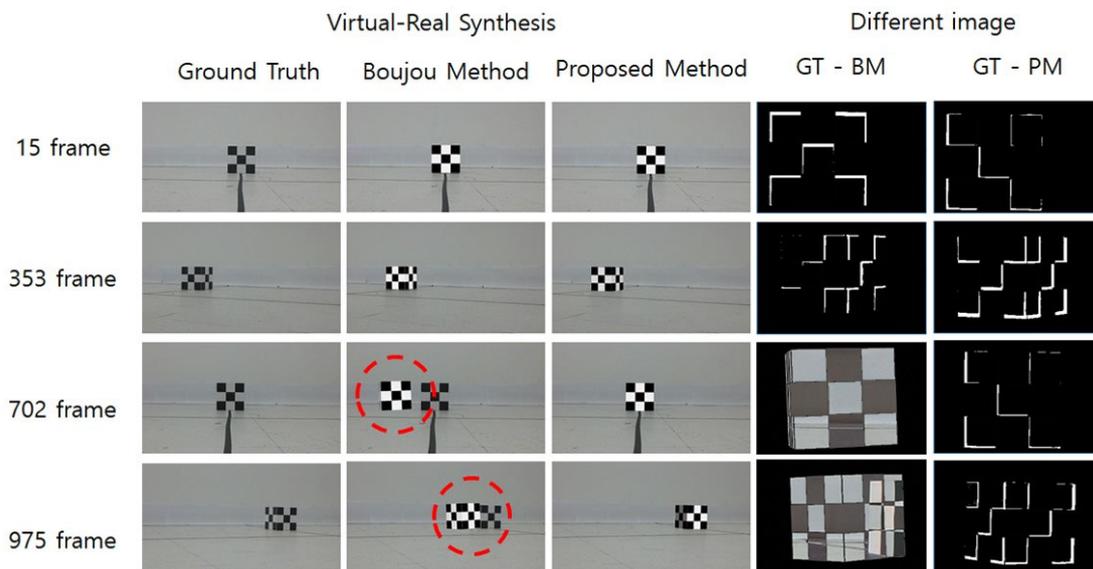


Figure 11. Comparison of different images in match-moving with motion blur.

Table 3. Match rate of different images in match-moving with motion blur.

	Method	Number of Unmatched Pixels	Specified Rate Resolution	Match Rate (%)
15 frame	GT-BM	2289	60,000	96.2
	GT-PM	1263	60,000	97.9
353 frame	GT-BM	1475	60,000	97.5
	GT-PM	3079	60,000	94.9
702 frame	GT-BM	32,383	60,000	46.4
	GT-PM	1213	60,000	98.0
975 frame	GT-BM	28,549	60,000	52.4
	GT-PM	2379	60,000	96.0

As a result, when there is motion blur, the Boujou method loses the feature points, which leads to the pausing of camera solve until the feature points are found again. At the point that it finds the feature points again, it generates a new camera solve. As a result, the camera position is lost during that time. Consequently, the new camera solve has a different position and rotation values compared to the existing (first) camera solve, so the virtual-real image synthesis cannot be accurate.

4.4. Pre-Visualization

The performance of the pre-visualization system was compared with that of Spectator View based on the number of polygons, as shown in Figure 12. As mentioned before, HoloLens cannot handle a larger number of polygons (>100,000) due to the lag it causes in frame rate and network speed. In Figure 12a, when the number of polygons was increased to more than 65,000, the frame rate dropped from 60 fps to 30 fps. Additionally, the network speed dropped from 1000 KB/s to 768 KB/s, as shown in Figure 12b. This problem was caused by the HoloLens’s so-called “holographic processing unit”, which can handle only a limited number of polygons at one time, which greatly reduces the frame rate and network speed of the HoloLens.

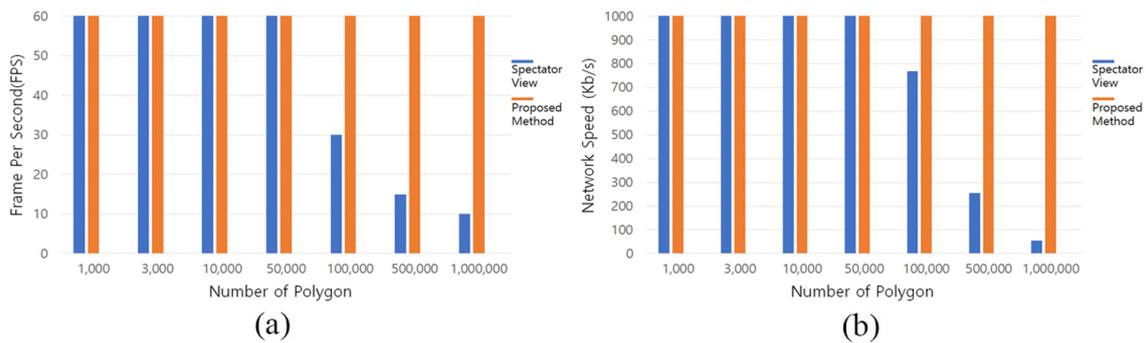


Figure 12. Performance comparison: (a) Spectator View; (b) proposed method.

On the other hand, the frame rate and network speed remained stable for the proposed method. This is because the proposed system does not require graphics information to be transmitted to HoloLens. In general, since high resolution 3D models are used in the film and entertainment industry, real-time virtual-real image synthesis cannot be obtained by conventional methods such as Spectator View. Consequently, it is difficult to obtain the accurate position information required for pre-visualization. Therefore, we employed pre-visualization used in the proposed system, which can obtain accurate position information from HoloLens in real time.

5. Conclusions

In this paper, we proposed a novel match-moving method for producing a stable virtual-real image synthesis for use in film production. The proposed system consists of two sub-systems. First, MS HoloLens is used to obtain the camera trajectory by obtaining position information in real time. This information was further used to create a virtual camera for match-moving. Secondly, an efficient pre-visualization system was implemented to preview the result of match-moving during recording. In this study, we exploited a system that overcomes the limitations of existing commercial software such as Boujou, which uses the feature point method. Our experimental results show that the proposed system is not affected by the external factors that were causing errors in conventional methods. The proposed method performed significantly better (98%) than the Boujou method (46%) especially when it comes to the motion-blur and occlusion in RGB camera video. However, the proposed system's limitation is that it does not have an optical zoom system. In the future, the authors of this paper are motivated to implement an optical zoom control system by installing a device such as focus gear ring that would be able to send necessary zoom information to the PC for further processing. In addition, Microsoft has released HoloLens 2 with a wider field-of-view that can be useful for spatial understanding required in the proposed system and needs further research.

It can be predicted that, with the advancement of the film industry, the demand for match-moving will also increase and therefore a more efficient and cost-effective system would be required. Boujou is a commercial software that costs around \$11,195 that is very expensive as compared to the proposed system which uses MS HoloLens. We expect that the system presented in this paper will be used for effective match-moving of the virtual-real image synthesis in low-budget production companies.

Author Contributions: Conceptualization, J.L. and J.H.; Data curation, J.L.; Formal analysis, S.K.; Funding acquisition, S.L.; Investigation, J.L. and J.H.; Methodology, J.L. and J.H.; Project administration, S.K.; Resources, K.K.; Software, K.K.; Supervision, S.L.; Validation, S.K.; Visualization, S.L.; Writing—original draft, J.H.; Writing—review and editing, S.K. and J.H.

Funding: This research received no external funding.

Acknowledgments: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2018-2015-0-00448) supervised by the IITP (Institute for Information & Communications & Technology Promotion).

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

References

1. Erica, H. *The Art and Technique of Matchmoving: Solutions for the VFX Artist*, 1st ed.; Elsevier: New York, NY, USA, 2010; pp. 1–14, ISBN 9780080961132.
2. Dobbert, T. The Matchmoving Process. In *Matchmoving: The Invisible Art of Camera Tracking*, 1st ed.; Sybex: San Francisco, CA, USA, 2005; pp. 5–10, ISBN 0782144039.
3. Pollefeys, M.; Van, G.L.; Vergauwen, M.; Verbiest, F.; Cornelis, K.; Tops, J. Visual Modeling with a Hand-held Camera. *Int. J. Comput. Vis.* **2004**, *59*, 207–232. [[CrossRef](#)]
4. Davison, A.J. Real-Time Simultaneous Localisation and Mapping with a Single Camera. In Proceedings of the Ninth IEEE International Conference on Computer Vision, Nice, France, 13–16 October 2003. [[CrossRef](#)]
5. Klein, G.; Murray, D. Parallel Tracking and Mapping for Small AR Workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234. [[CrossRef](#)]
6. Davison, A.J. SLAM++: Simultaneous Localisation In addition, Mapping at the Level of Objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1352–1359. [[CrossRef](#)]
7. Bao, S.Y.; Savarese, S. Semantic structure from motion. In Proceedings of the IEEE CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; Volume 8. [[CrossRef](#)]
8. Hafeez, J.; Jeon, H.J.; Hamacher, A.; Kwon, S.C.; Lee, S.H. The Effect of Patterns on Image-Based Modelling of Texture-less Objects. *Metrol. Meas. Syst.* **2018**, *25*, 755–767. [[CrossRef](#)]
9. Boujou. Available online: <https://www.vicon.com/products/software/boujou> (accessed on 21 January 2019).
10. Pftrack. Available online: <https://www.thepixelfarm.co.uk/pftrack/> (accessed on 23 January 2019).
11. Newcombe, R.; Izadi, S.; Hilliges, O.; Molyneaux, D.; Kim, A.; Davison, P. KINECTFUSION: Real-time Dense Surface Mapping and Tracking. In Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality, Basel, Switzerland, 26–29 October 2011; pp. 127–136. [[CrossRef](#)]
12. Liu, Y.; Dong, H.; Zhang, L.; Abdulmotaleb, E.S. Technical Evaluation of HoloLens for Multimedia: A First, Look. *IEEE Multimed.* **2018**, *25*, 8–18. [[CrossRef](#)]
13. Yingnan, J. Footstep Detection with HoloLens. *Am. J. Eng. Res. (AJER)* **2018**, *7*, 223–233.
14. Vassallo, R.; Adam, R.; Elvis, C.S.C.; Terry, M.P. Hologram stability evaluation for Microsoft HoloLens. *SPIE Proc. Med. Imaging* **2017**, *10136*. [[CrossRef](#)]
15. Lee, J.S.; Lee, I.G. A Study on Correcting Virtual Camera Tracking Data for Digital Compositing. *J. Korea Soc. Comput. Inf.* **2012**, *17*, 39–46. [[CrossRef](#)]
16. Kuamr, R.K.; Ilie, A.; Frahm, J.; Pollefeys, M. Simple Calibration of Non-overlapping Cameras with a Mirror. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–7. [[CrossRef](#)]
17. Kim, M.H.; Yu, J.J.; Hong, H.K. Using Camera Tracking and Image Composition Technique In Visual Effect Imaginary Production. *J. Korea Contents Assoc.* **2011**, *11*, 135–143. [[CrossRef](#)]
18. Yokochi, Y.; Ikeda, S.; Sat, T.; Yokoya, N.; Un, W.K. Extrinsic Camera Parameter Estimation Based-on Feature Tracking and GPS Data. In *Asian Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 369–378. [[CrossRef](#)]
19. Luo, A.C.; Chen, S.W.; Tseng, K.L. A Real-time Camera Match-Moving Method for Virtual-Real Synthesis Image Composition Using Temporal Depth Fusion. In Proceedings of the 2016 International Conference on Optoelectronics and Image Processing, (ICOIP), Warsaw, Poland, 10–12 June 2016. [[CrossRef](#)]
20. Hwang, L.H.; Lee, J.H.; Jahanzeb, H.; Kang, J.W.; Lee, S.H.; Kwon, S.C. A Study on Optimized Mapping Environment for Real-time Spatial Mapping of HoloLens. *Int. J. Internet Broadcast. Commun.* **2017**, *9*, 1–8. [[CrossRef](#)]
21. Spectator View for HoloLens. Available online: <http://docs.microsoft.com/en-us/windows/mixed-reality/spectator-view> (accessed on 5 February 2019).
22. Lee, J.H.; Kim, S.H.; Kim, L.H.; Kang, J.W.; Lee, S.H.; Kwon, S.C. A Study on Virtual Studio Application using Microsoft HoloLens. *Int. J. Adv. Smart Converg.* **2017**, *6*, 80–87. [[CrossRef](#)]
23. Zhang, Z.A. Flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [[CrossRef](#)]

24. Mathieu, G.; Pierre-Olivier, B.; Jean-Philippe, D.; Luc, B.; Jean-Francois, L. Real-Time High-Resolution 3D Data on the HoloLens. In Proceedings of the 2016 IEEE International Symposium on Mixed and Augmented Reality, Merida, Mexico, 19–23 September 2016; pp. 189–191. [[CrossRef](#)]
25. Mixed Reality Toolkit-Unity. Available online: github.com/Microsoft/HoloToolkit-Unity (accessed on 1 February 2019).
26. Andrian, B.; Jorge, L.; Mariano, A.; Roberto, L. Comparison of Oculus Rift and HTC Vive: Feasibility for Virtual Reality-Based Exploration, Navigation, Exergaming, and Rehabilitation. *Game Health J.* **2018**, *7*. [[CrossRef](#)]
27. Steven, M.L.; Anna, Y.; Max, K.; Michael, A. Head tracking for the Oculus Rift. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014. [[CrossRef](#)]
28. Manikanta, K.; Sachin, K. Position Tracking for Virtual Reality Using Commodity Wifi. In Proceedings of the 10th on Wireless of the Students, by the Students, and for the Students Workshop, New Delhi, India, 2 November 2018; pp. 15–17. [[CrossRef](#)]
29. Xu, X.; Karen, B.C.; Iia-Hua, L.; Robert, G.R. The Accuracy of the Oculus Rift Virtual Reality Head-Mounted Display during Cervical Spine Mobility Measurement. *J. Biomech.* **2015**, *48*, 721–724. [[CrossRef](#)] [[PubMed](#)]
30. Marcin, M.; Marek, P.; Mateusz, P. Testing the SteamVR trackers operation correctness with the OptiTrack system. *SPIE Conf. Integr. Opt.* **2018**, *10830*. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).