International Journal of Geo-Information

MDPI

Article

Dynamic Fusion Technology of Mobile Video and 3D GIS: The Example of Smartphone Video

Ge Zhu, Huili Zhang, Yirui Jiang, Juan Lei, Linqing He and Hongwei Li *

The School of Geo-Science & Technology, Zhengzhou University, Zhengzhou 450000, China
* Correspondence: lhw29691518@zzu.edu.cn; Tel.: +86-136-7371-2015

Abstract: Mobile videos contain a large amount of data, where the information interesting to the user can either be discrete or distributed. This paper introduces a method for fusing 3D geographic information systems (GIS) and video image textures. For the dynamic fusion of video in 3DGIS where the position and pose angle of the filming device change moment by moment, it integrates GIS 3D visualization, pose resolution and motion interpolation, and proposes a projection texture mapping method for constructing a dynamic depth camera to achieve dynamic fusion. In this paper, the accuracy and time efficiency of different systems of gradient descent and complementary filtering algorithms are analyzed mainly by quantitative analysis method, and the effect of dynamic fusion is analyzed by the playback delay and rendering frame rate of video on 3DGIS as indicators. The experimental results show that the gradient descent method under the Aerial Attitude Reference System (AHRS) is more suitable for the solution of smartphone attitude, and can control the root mean square error of attitude solution within 2°; the delay of video playback on 3DGIS is within 29 ms, and the rendering frame rate is 34.9 fps, which meets the requirements of the minimum resolution of human eyes.

Keywords: virtual–real fusion; video GIS; smartphone video; 3D GIS; dynamic fusion

Citation: Zhu, G.; Zhang, H.; Jiang, Y.; Lei, J.; He, L.; Li, H. Dynamic Fusion Technology of Mobile Video and 3D GIS: The Example of Smartphone Video. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 125. https://doi.org/10.3390/ijgi12030125

Academic Editors: Wolfgang Kainz and Wei Huang

Received: 18 December 2022
Revised: 7 March 2023
Accepted: 9 March 2023
Published: 14 March 2023

1. Introduction

In the study of virtual–real fusion technology, 3D video fusion methods fusing virtual 3D GIS scenes and real video data to improve immersive visual experience have emerged due to the massive popularity of camera equipment and the booming 3DGIS industry [1–5]. Three-dimensional video fusion methods fuse video real-time registration into 3D virtual scenes [6] given the contextual information between videos [7] and create viewing scenes that conform to human cognitive habits [8], eliminating the "information noise" caused by obtaining multiple video information in a table or tree structure [9] and enabling immersive access to objects and semantic information within the video [10,11].

In recent years, the popularity of mobile filming devices with positioning function such as smartphones, drones and car cameras of the smart driving, and the rise of some short video software have generated many videos with mobile attributes. Because of their mobility, such videos contain richer object and semantic information. However, traditional fusion methods can only fuse videos with fixed positions and perspectives, which cannot meet the urgent demand for dynamic fusion of mobile videos in today's era. Achieving the fusion of mobile video with 3D GIS scenes complements the absence of mobile video sources in previous fusion, and theoretically realizes the fusion of all videos, which is of great practical significance in many fields such as ensuring public safety, improving the utilization of social and natural resources, and preventing natural and non-natural disasters.

In the study of 3D video fusion technology, researchers usually divide the fusion methods into two categories according to the different focus of fusion: augmented reality class with GIS-enhanced video [12] and augmented virtual class with video-enhanced GIS [13–16]. The augmented reality class with GIS-enhanced video often uses the video-labeled map approach [17–20] for fusion. This method combines video points with maps to

enable querying and guiding of videos from different locations [21]. However, video and video, and video and geographic scenes are still displayed separately without changes in location and perspective, and are not suitable for dynamic fusion.

The augmented virtual class with video-enhanced GIS also contains video image panoramic stitching methods, video and 3D scene overlay methods, and video and 3D scene fusion methods [22]. The video image stitching method [23] uses a spherical fixed virtual model to stitch the video streams in a panoramic view, but the viewpoint is only limited to the vicinity of the shooting viewpoint. Although both video overlay and fusion to 3D scenes methods [7,12,24] can meet the requirements of rendering when the viewpoint position and perspective change moment by moment, the video overlay method only allows the user to view the results of dynamic fusion on the transfer path of the camera viewpoint, and the fusion effect is not ideal. The video and 3D scene fusion method [13] registers the textures of the video images on the 3D model in real time, allowing roaming viewing at any location [6,25].

The advantages of video and 3D scene fusion methods have led researchers to continuously optimize the method. The authors of literature [26] first proposed a video image-based fusion technique to meet the requirements of virtual model accuracy in the process of virtual–real fusion and overcome the problem of depth mismatch brought by the video projection itself. The authors of literature [27] proposed a method based on multi-threading technology and quad-tree indexing video scheduling to achieve efficient scheduling and display of video images in large-scale 3D scenes. The authors of literature [28] designed an algorithm to solve the phenomenon of occlusion penetration generated by insufficient image depth information. The authors of literature [29] proposed a topology network-based method for fusing multi-channel video with 3DGIS scenes, which does not require traversing each object in the field of view in turn to determine whether to reject the object. The authors of literature [30] studied the method of linkage video projection of virtual and real scenes for PTZ (Pan/Tilt/Zoom) equipment, which improves the display performance of multi-channel video integration in 3D scenes to a certain extent. The previous studies focused on the integration of fixed-angle surveillance video with 3DGIS, ignoring mobile video sources.

In this paper, we combine the advantages of video and 3D scene fusion methods with the pose solving system [31] and GPS positioning system. For the problem of dynamic fusion of video in 3DGIS scene with momentary changes in viewpoint position and shooting device pose angle, taking cell phone video as an example, we use gradient descent method to solve the pose of cell phone when shooting video according to its MEMS sensor, obtain GPS position information of cell phone, use motion interpolation method to conduct temporal correspondence between video frame and cell phone position pose, and then propose a method to build dynamic depth camera projection texture mapping method, dynamically calculate the mapping relationship between video texture and 3D scene model texture, and realize the dynamic fusion of mobile video, which can be viewed in any position of 3DGIS roaming. The fusion flow chart is shown in Figure 1.

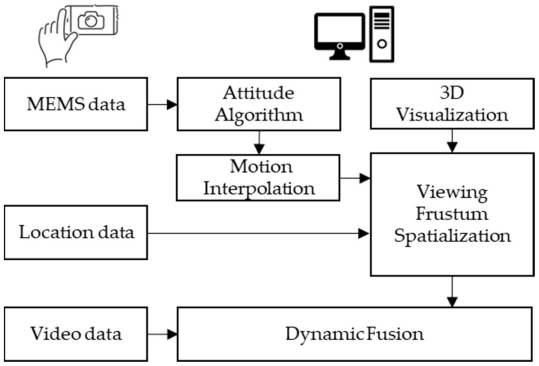

Figure 1. Dynamic fusion flowchart with mobile video as an example.

2. Materials and Methods

2.1. Technology and Method in Data Preparation Stage

2.1.1. GIS 3D Visualization

3DGIS is established for fixing the essential defect of 2DGIS losing the amount of spatial information (especially elevation information and 3D topological spatial information), and attempts to understand and express the real-world features, geographic phenomena and their spatial relationships directly from the perspective of 3D space. WebGL (Web Graphics Library) provides a rich variety of 3D model API [32], and STK [33] introduced Cesium based on the research and development of WebGL. Cesium supports direct drawing of 3D models using gltf, glb, and 3Dtiles files, and also allows the use of the Cesium Lab tool to convert common 3D model data into 3Dtiles format by processing slices in a tree structure using the Cesium Lab tool. The path of the organization file (json file) of 3Dtiles data is stored in the created Cesium3DTileset object, drawn and added in the created scene 3D scene using primitive API. The specific process is shown in Figure 2, and the visualization of 3DGIS scene can be realized.

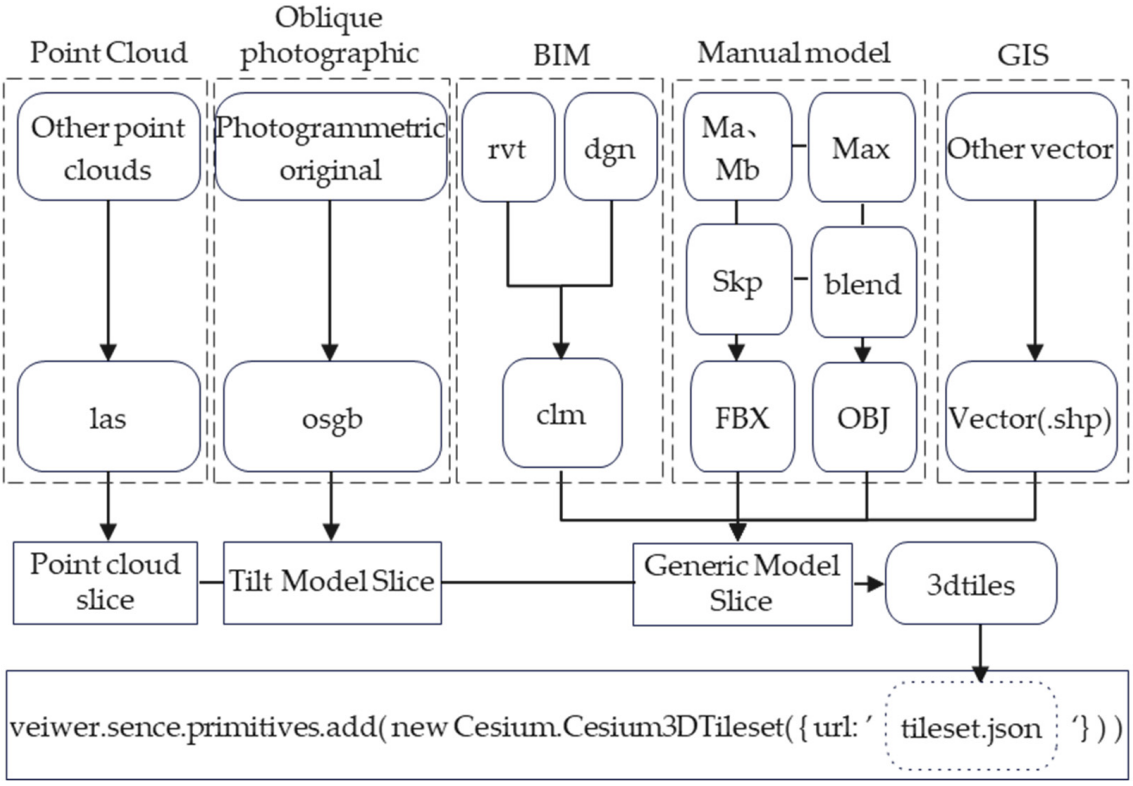

Figure 2. Cesium loads various 3D models.

2.1.2. Attitude Algorithm

Attitude solving for smartphones is to solve the attitude of the phone at the moment of shooting based on the output data of accelerometer, magnetometer, and gyroscope to obtain the yaw, pitch, and roll angles of the phone when shooting video. Commonly used pose solving methods are complementary filtering algorithm, extended Kalman filtering, and gradient descent algorithm [34–37]. The gradient descent method uses quaternion to solve the attitude. In the quaternion formulation of the attitude transformation, the transformation of the navigation coordinate system to the carrier coordinate system is achieved by a single rotation of α angle around a vector u defined in the reference coordinate system [38]. Based on the principle of attitude transformation, the mathematical formulation of the

rotation of the vector v from the M-system (model coordinate system) to the N-system (navigation coordinate system) using quaternion is implemented as shown in Equation (1),

$$^{N}v = {}^{M}_{N}Q_t \otimes {}^{M}v \otimes {}^{M}_{N}Q_t^{\ *}, \tag{1}$$

where ^{M}v denotes the vector in the M-system, ^{N}v denotes the vector in the N-system, ${}^{M}_{N}Q_t$ denotes the quaternion of the rotation of the M-system with respect to the N-system at time t, and ${}^{M}_{N}Q_t^{\ *}$ denotes the conjugate hyper-complex of ${}^{M}_{N}Q_t$.

The gyroscope has integration error due to zero-point drift noise and other effects. The accelerometer has poor high frequency performance and poor accuracy in high-speed motion. Therefore, literature [34] used accelerometer and magnetometer to construct the error equation and Jacobi equation and employed the gradient descent method to detect the gradient quaternion $Q_{\nabla,t}$, converge with the gyroscopic quaternion ${}^{M}_{N}Q_{w,t}$ at the same speed and fuse linearly to solve the optimal rotational quaternion ${}^{M}_{N}Q_t$, as shown in Equation (2), where ε_t is the weight of linear fusion of two rotating quaternions at moment t:

$${}^{M}_{N}Q_t = \varepsilon_t {}^{M}_{N}Q_{\nabla,t} + (1-\varepsilon_t){}^{M}_{N}Q_{w,t} \qquad 0 \le \varepsilon_t \le 1. \tag{2}$$

The gradient descent method is applied to the heading attitude reference system (AHRS) and the inertial measurement unit (IMU). AHRS has one more magnetometer data than IMU, which can achieve the effect of bias correction. However, if the magnetometer data is not accurate enough, the correction will cause more deviation instead. The corrective effect of the magnetometer will be discussed in detail in the experimental results.

2.1.3. Interpolation of Spatio-Temporal Trajectories Based on Cubic Polynomials

The sampling frequency of MARG (magnetic, angular rate and gravity) sensor and GPS sensor is approximately 20–25 times/second, while the frame rate of video image is 27.9 frames per second. In order to position the pose of video frames and to meet the local time system of video playback and the global time system of cell phone when the position pose changes, it is necessary to interpolate the geodetic coordinate data of GPS and the Euler angles obtained from the attitude decomposition are interpolated in time and space.

The triple interpolation method can obtain the parameters a_i, (i = 0, 1, 2, 3), according to the time, position, and velocity of the starting and ending points, so that $\varphi(t)$ approximates the real motion function $f(t)$ and realizes the motion interpolation between two points [39–41], as shown in Equation (3):

$$\varphi(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3. \tag{3}$$

Direct interpolation is performed for the Euler angle. In the interpolation of geodetic coordinates (BLH), the velocity vector v is decomposed based on the cell phone movement velocity v obtained from GPS, and the north-east-up (NEU) coordinate system is established with the cell phone position as the origin at time t. The x-axis component v_x is the direction of linear velocity change of latitude L perpendicular to the Prime Vertical, the y-axis component v_y is the direction of linear velocity change of longitude B perpendicular to the meridian circle, and the z-axis component v_z is the direction of change of altitude H perpendicular to the ellipsoidal plane. To interpolate the longitude and latitude, it is also necessary to convert the linear velocities v_x and v_y into angular velocities w_B and w_L, and the solution formula is as shown in Equation (4), where M denotes the radius of the Meridian circle, N denotes the radius of the Prime Vertical, a denotes the long semi-axis of the WGS-84 ellipsoid, e denotes the Angular Eccentricity of the WGS-84 ellipsoid, and B denotes the longitude at that moment.

$$\begin{gathered} w_B = \frac{v_y}{N}, w_L = \frac{v_x}{M}, \\ M = a\left(1-e^2\right)\left(1-e^2\sin^2 B\right)^{-\frac{3}{2}}, \\ N = a\left(1-e^2\sin^2 B\right)^{-\frac{1}{2}}. \end{gathered} \tag{4}$$

2.2. Dynamic Projection of Smartphone Video

This section integrates the video data and the position and pose data corresponding to the video frames, as well as the model displayed on the 3DGIS. The dynamic projection is achieved through the construction and spatialization of depth camera view bodies, projection of single video images, and dynamic depth camera update techniques. The key of the technology is to dynamically construct the depth camera based on the position and pose data corresponding to the video frames and obtain the depth value of each image frame, as well as invert the depth values with the slice elements captured by the 3DGIS window camera to the depth camera space to perform depth judgments and project the video frame textures on the 3DGIS model based on the texture mapping method.

2.2.1. Construction and Spatialization of Depth Camera Viewing Frustum

The principle of video projection ground is to project a two-dimensional image focused on the camera into a three-dimensional scene [42,43]. The cell phone camera will only record texture information of the closest object to the camera, which means that only slice element textures with smaller depth values can be recorded on the video image [44]. However, the video image lacks depth information and cannot be accurately spatialized. Therefore, to simulate the shooting position and pose of the camera in the 3D scene, a depth camera view cone is constructed, as shown in Figure 3. A point light source is placed at the viewpoint position of the depth camera, and by drawing on the principle of shadow mapping, a shadow is cast inside the view cone and the depth image and transformation matrix of the view cone at the point light source are obtained.

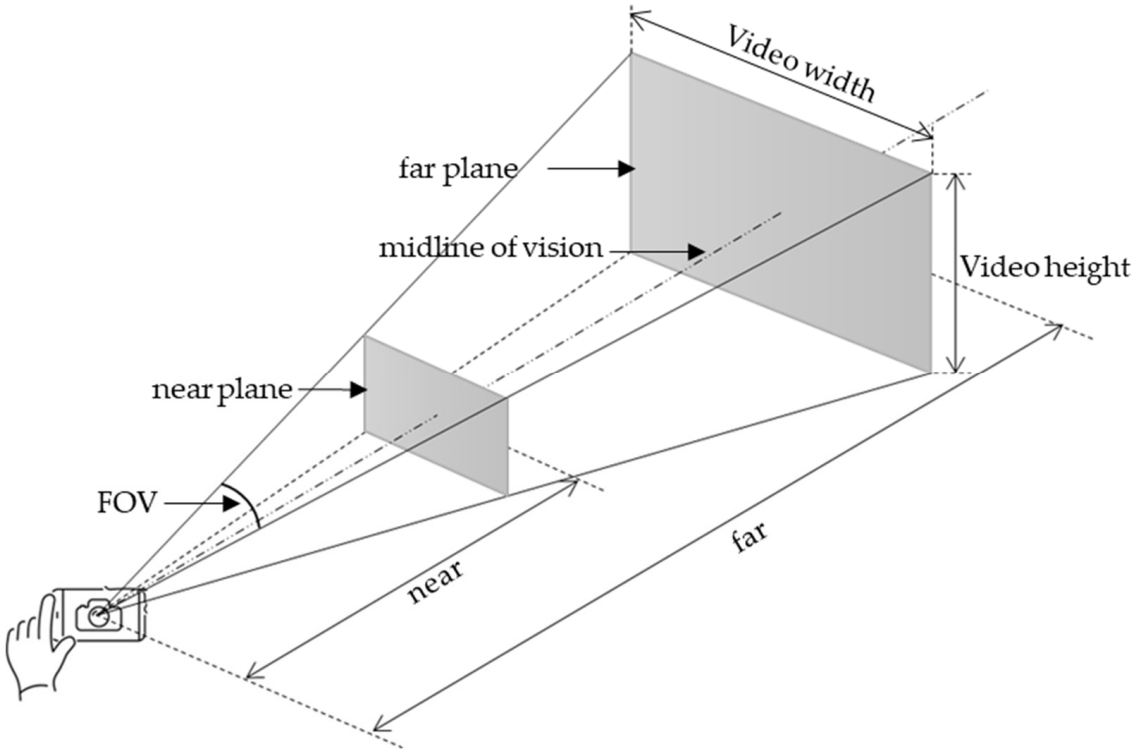

Figure 3. The chart of viewing frustum.

The camera model coordinate system is established with the viewpoint as the origin, the direction from the center of the far plane to the viewpoint, i.e., the opposite direction of the line of sight, as the positive z-axis, the orientation of the camera tip as the positive y-axis, the direction perpendicular to the y-o-z plane and located in the right-hand direction at the time of shooting as the positive x-axis. The parameters required to establish the viewing frustum model coordinate system are shown in Table 1. To realize the spatialization of

video images is to transform the camera model coordinate system of the camera viewing frustum into the world coordinate system, which requires the achievement of the model matrix ($R = R_{rotation}R_{translation}$).

Table 1. The parameters of establishing the viewing frustum model coordinate system.

Parameters	Meaning
FOV	The opening angle of the camera's field of view.
near	Vertical distance from origin to near plane.
far	Vertical distance from origin to far plane.
aspect	Aspect ratio of video.
BLH	Latitude, longitude, and elevation in geodetic coordinate system.
Eulerian angles	Yaw, pitch and roll angle during mobile phone shooting.

2.2.2. Projection of A Single Video Image

The 2D image seen on the browser window is captured by the window camera in 3DGIS according to its position, pose, and a series of operations such as view transformation, projection transformation and viewport transformation are performed on the window camera view cone. The transformation process is shown in Figure 4.

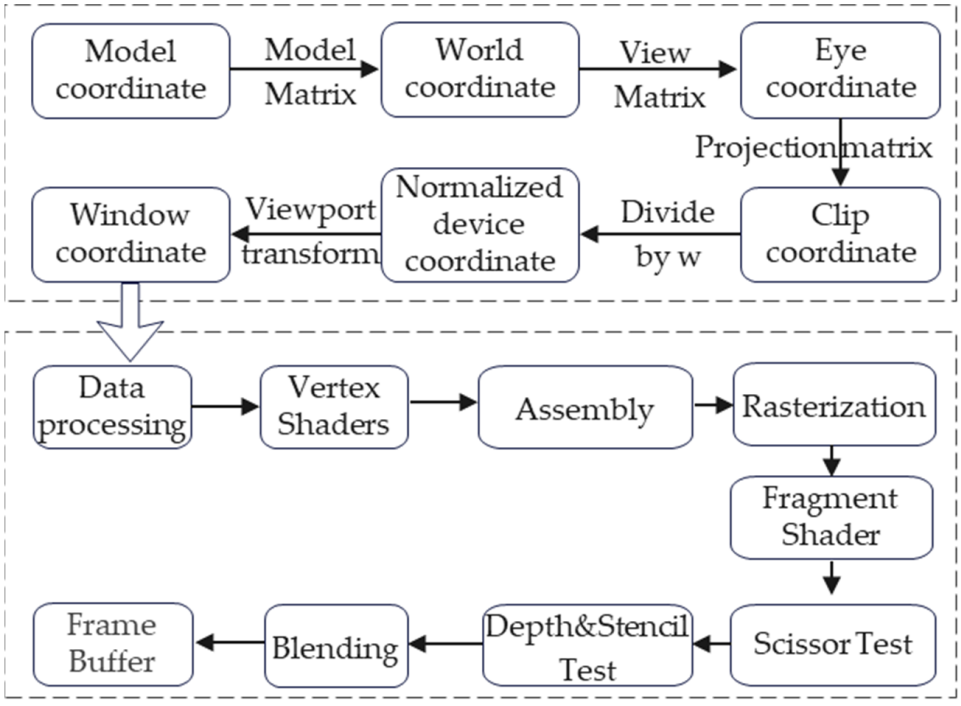

Figure 4. WebGL rendering pipeline.

To make the video texture visible on the browser window, it is necessary to obtain the depth value of the video image. The depth value is used to perform a depth check on the fragment shader after transforming the window camera viewing frustum from the world coordinate to the window coordinate system. For each fragment of the window, it is determined whether it has a video texture or not. If there is a video texture, then the texture value is modified in the fragment shader to replace the corresponding fragment texture. The flowchart is shown in Figure 5 and the Steps are implemented as follows.

1. Obtain the transformation matrix of the depth camera's viewing frustum. Compute the transformation clip matrix based on the coordinates and orientation of the depth camera in the window camera coordinate system that transforms the fragment from the window camera clip coordinate system to the depth camera clip coordinate system.
2. Pass the depth map and related parameters into the fragment shader. The depth map is passed into the fragment shader as a consistent variable of type sampler2D for 2D textures, and the transform clip matrix, video image texture, and viewpoint window camera coordinates are passed into the shader for depth testing. As long as the phone's position and pose do not change, the texture coordinates of each fragment corresponding to the video image also remain relatively unchanged and the depth map does not need to be updated. In contrast, to achieve dynamic video projection of the phone, the position and pose of the depth camera have to be constantly refreshed.
3. Perform inter-camera spatial transformation. Since the depth detection of the fragments within the depth camera's viewing frustum is performed in the fragment shader, the fragments in the window coordinate system of the window camera need to be inverted and transformed to the camera coordinate system of the window camera. Then, multiply the transform clip matrix of the depth camera to transform the fragments to the clip coordinate system of the depth camera. After the perspective division and viewport transformation of the fragment coordinates, the fragment is transformed to the window coordinate system of the depth camera, and the z-value of the fragment coordinates is its depth value depth at this time.
4. Conduct the depth camera Scissor Test, which determines whether a fragment is within the depth camera's viewing frustum. In the depth camera's normalized device coordinate system, the various components of the fragment's coordinates located within the view cone are between -1 and 1. Coordinates beyond this interval are not within the depth camera's frustum. For the fragments that are not in the frustum, the original texture is retained and moves to the rendering pipeline of the next fragment. For the fragments that are in the frustum, proceed to Step 5 for depth testing.
5. Perform the depth test. In the actual window coordinate system of the depth camera, the UV coordinate of the fragment can be taken out of the depth value vdepth of the video texture of the depth map. The depth value of the depth map is the depth of the closest fragment to the depth camera, i.e., the smallest value. If the depth value of the fragment minus the depth deviation Δd is less than or equal to the depth value vdepth of the video texture, then the fragment is considered to be covered by the video texture and the texture coordinates correspond to the UV coordinates of the fragment.

To improve the accuracy of rendering, the video image depths of the eight surrounding fragments of the fragment in the depth camera window coordinate system are compared with the slice element depths one by one and recorded, and only the slice elements that meet the following nine cases are assigned texture values of the video. Figure 6a,c,g,i are the cases where the slice element is located at the corner of the video image; Figure 6b,d,f,h are the cases where the slice element is located at the edge of the video image; and e is the case where the slice element is in the middle of the video image.

2.2.3. Dynamic Video Image Fusion

When shooting video dynamically with a smartphone, the position and pose of the phone are changing all the time. Each frame has its own position and pose, and the timestamp of each video frame can be calculated by obtaining the frame rate of the cell phone video and the timestamp of the start of the video shooting, as shown in Equation (5), where Δt is the average time interval between two frames in milliseconds (MS) and $timeStamp_i$ is the timestamp of the ith frame of the video image. According to the timestamp of each

frame, the corresponding position and pose can be looked up in the table of position and pose after cubic polynomial interpolation.

$$timeStamp_i = startTime + i \times \Delta t,\ \Delta t = \frac{1000}{v_{FPS}}. \tag{5}$$

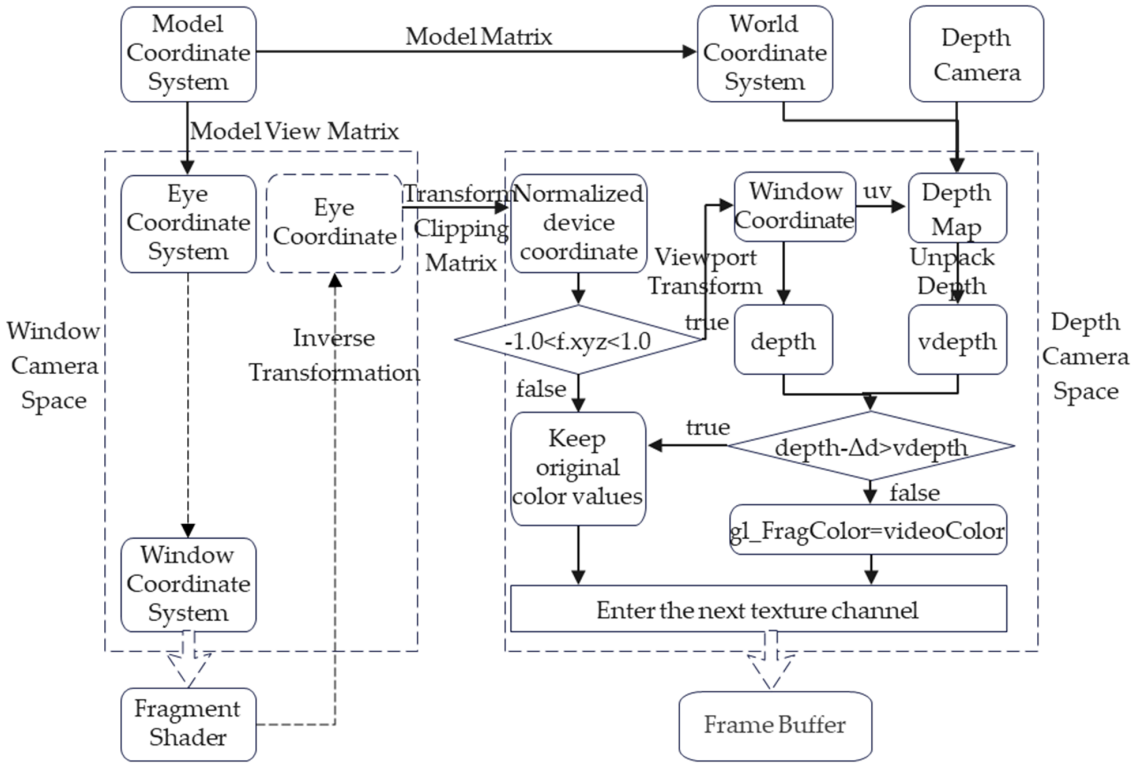

Figure 5. Flow chart of projection texture mapping for a single video image.

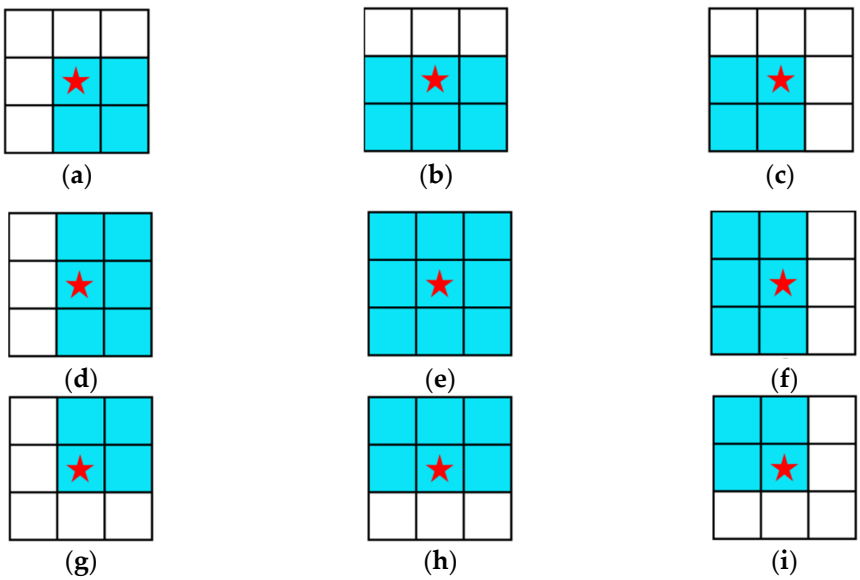

Figure 6. Position of fragment in video image. (**a**–**i**) show the nine position cases of slice elements in the video. Where the blue blocks indicate the slice elements located in the video image. The white blocks indicate slice elements that are outside the video image. The slice elements with red stars indicate the current slice element.

For dynamic projection of the video ground, the 3D model is first loaded on Cesium. Then, the video image is imported and parsed, and the image texture is stored frame by frame in the Texture class to pass into the shader and wait for texture mapping. Finally, the timestamp, solved pose data and sensor data such as GPS are read into the browser as a table for spatio-temporal interpolation. After the interpolation is completed, the viewing frustum drop shadow is constructed based on the position and pose and related internal and external parameters. The depth image and transform clipping matrix within the viewing frustum are obtained and passed to the shader for texture mapping of a single image. After one mapping is completed, the position and pose parameters of the next frame are passed in. The depth image and transform clipping matrix of the viewing frustum are updated and the texture mapping is performed for the next frame to achieve dynamic projection fusion of the moving video, as shown in Figure 7.

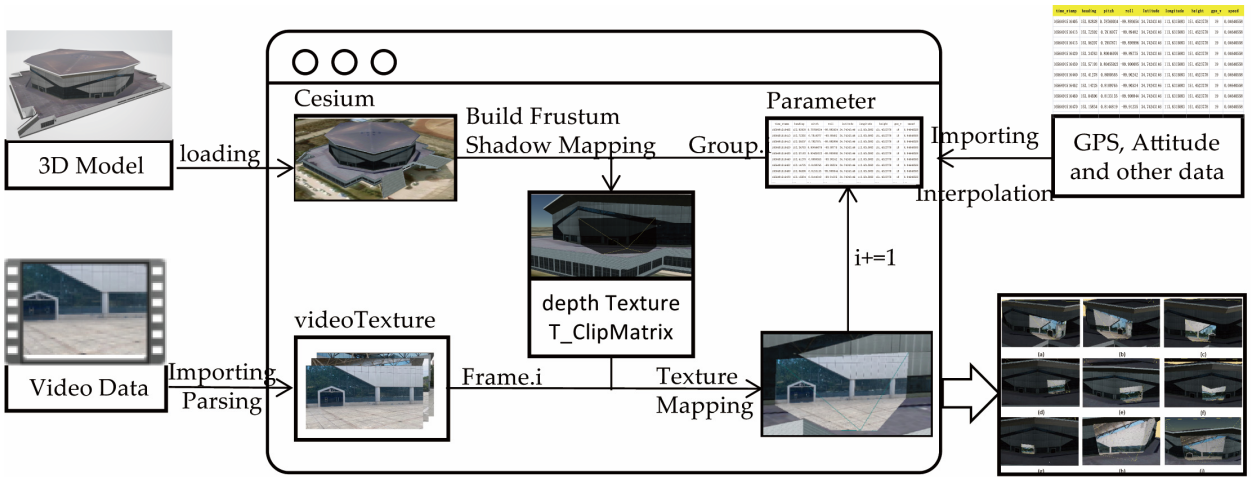

Figure 7. Dynamic fusion of mobile video.

In practical development, the timestamp of each image frame can be calculated and queried. This will increase the caching and rendering burden of the browser and cause the problem of lagging delay. To improve the rendering efficiency, instead of calculating the timestamp of each video image frame, we combine the timer function to perform cumulative sampling of the start timestamp. Every Δt time, the video is played from the previous frame to the next frame, at which time the accumulation of the start timestamp is executed and the position and pose of the phone at that time are extracted and updated into the depth camera until the video is played over. When the start timestamp is greater than the end timestamp, the timer is turned off and the variable is destroyed.

3. Results

In order to verify the effect of the video dynamic projection algorithm in the front-end display, a live 3D model of the stadium of the South Campus of Zhengzhou University is used as a test site in this paper. The external hexahedron of the stadium is 90 m long, 81 m wide and 20 m high. Smartphones can be used with any phone that supports accelerometer, magnetometer, gyroscope, and GPS positioning. In this article, we used HUAWEIY92019 with a focal length of 3.6 mm, video format MP4 and frame rate of 27.9 fps. The computer system used for the experiment is Windows 11, the CPU is Intel (R) Core (TM) i7-10700 at 2.90 GHz, the GPU is AMD Radeon (TM) 625, and the RAM is 8.00 GB. The browser selected for the front-end fusion display is chrome. The experiment starts with the acquisition of video, GPS and sensor data by a cell phone, which is then transferred offline to a computer for subsequent fusion operations.

3.1. Analysis of Experimental Results on the Influence of Magnetometer on Attitude Solution Accuracy

While using the cell phone to shoot video, the GPS information of the cell phone and the data of gyroscope, accelerometer and magnetometer are collected, the collection frequency is 20–25 times/second, and the collection method is recording each time any sensor callback for a reference. The acquisition scene is unobstructed over the gymnasium of Zhengzhou University South Campus, with normal GPS signal and a small high-rise building (9-story high) in the surrounding 50 m. An initial orientation is determined and recorded before data acquisition to provide initial values for the subsequent update of the quadrature to solve the Euler angle.

3.1.1. Experiment Results of Attitude Solution

The AHRS system has more magnetometer data compared to the IMU system, and due to the gyroscope time drift, the error in integrating the rotation angle during the motion becomes larger and larger over time [45]. Therefore, the magnetometer is needed to improve the accuracy by calibrating the geographic azimuth angle of the moving target motion in a timely manner. However, the presence of factors such as the magnetometer's own errors and complex magnetic fields will not only not improve the accuracy of the solution, but also lead to a decrease in the accuracy of the measurement. Therefore, we use two ways of two algorithms, gradient descent method and complementary filter method, to perform attitude solution on the collected 955 sets of data, and compare and analyze the effect of using different system magnetometers on the attitude solution accuracy in outdoor sites. The experimental results are shown in Figure 8.

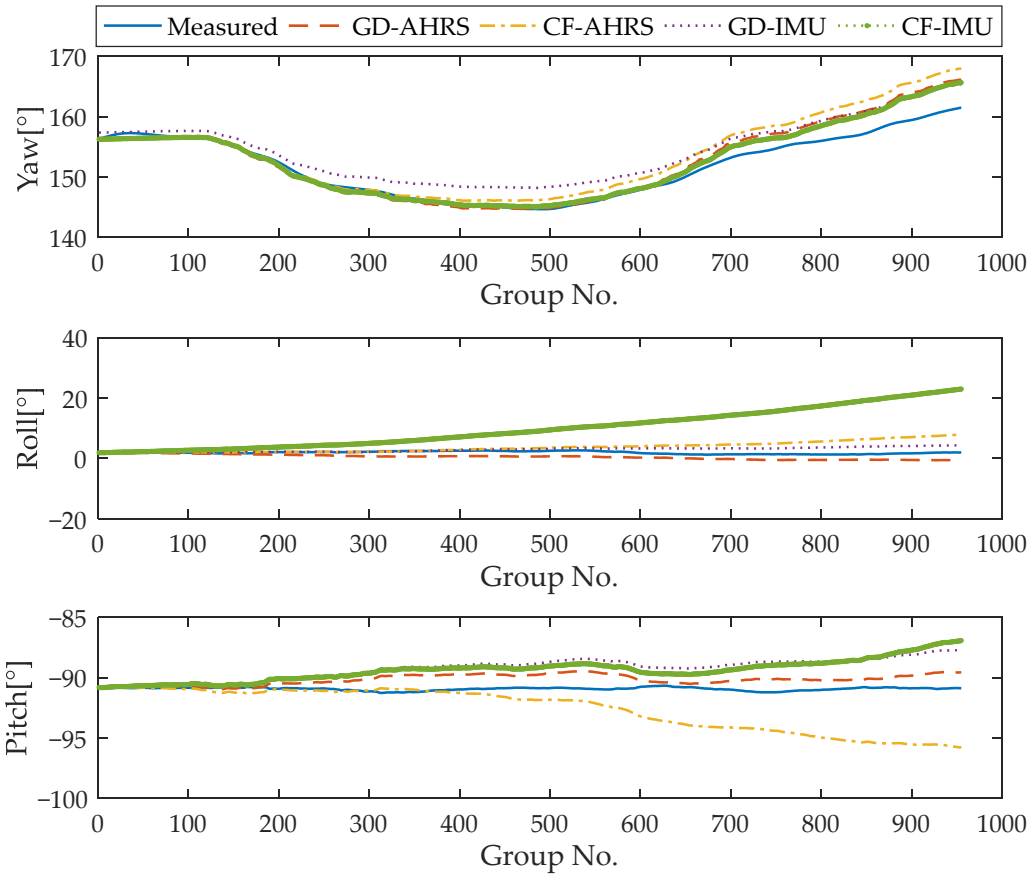

Figure 8. Comparison of AHRS and IMU Attitude Solution Results of Gradient Descent (GD) and Complementary Filtering (CF).

Figure 8 represents the actual measurements with the predicted values of the gradient descent method and the complementary filtering method with and without magnetometer data. As can be seen from the figure, the pitch and roll angles remain essentially unchanged when the video is recorded, and the floating yaw angle varies considerably. It is relatively a little easier to obtain the accurate roll and pitch angles than to obtain the accurate yaw angle. Meanwhile, the drift error of the gyroscope becomes larger and larger as time goes by, and the four predicted values are more and more offset from the actual measured values. Table 2 presents the deviation of the four predicted values from the actual measured values and the root mean square (RMS) error.

Table 2. RMS of different attitude solution methods.

Euler Angle	Gradient Descent		Complementary Filtering	
	AHRS	IMU	AHRS	IMU
RMS (yaw)	1.9433°	2.7463°	2.8762°	1.5940°
RMS (pitch)	1.5649°	1.3788°	2.6331°	10.2496°
RMS (roll)	0.8943°	1.8922°	2.3976°	1.7918°

3.1.2. Experimental Analysis of Attitude Solution

From the different perspectives of the algorithms, the complementary filtering method is less stable for the gradient descent method in the pitch and roll angles relative to the gradient descent method, which causes a larger offset. As for the solution of yaw angle, the complementary filtering method has one best and one worst prediction result in both systems, which indicates that the correction of the magnetometer plays the opposite effect, and confirms that the complementary filtering method has a high sensitivity to the measurement error of the magnetometer, and if the error of the magnetometer is larger it will lead to worse accuracy of the solution. The gradient descent method combines the attitude quaternion and gyroscope quaternion linearly, which reduces the influence of the magnetometer error on the accuracy, so the two systems of the gradient descent method have similar prediction values in the solution of yaw angle.

From the point of view of the influence of the magnetometer, the magnetometer correction of the bias also brings errors. In the solution of the yaw angle, the magnetometer correction improves the accuracy of the gradient descent method, which has a counter effect on the complementary filtering method. In the solution of the roll angle, the magnetometer error causes the gradient descent method to improve the RMS by 0.2, which has the opposite effect, but the accuracy of the complementary filter method has a significant improvement. In the solution of the pitch angle, it is consistent with the yaw angle, which improves the accuracy of the gradient descent method and reduces the accuracy of the complementary filter method.

From the overall accuracy, the AHRS gradient descent method under magnetometer correction is optimal by counting the root mean square total error of Euler angles. While the IMU system lacks the processing operation process of magnetometer data, it is also a question whether the time benefit relative to AHRS will make up for the lack of accuracy. Three groups of different data volumes were solved and the solving time was recorded, as shown in Table 3. The first group has a small amount of data totaling 6624 sets of data, and the complementary filtering method can clearly show the time advantage of IMU, but only about the 200 ms difference; for the gradient descent method, with only a 5 ms difference, it can be said that there is no gap The time difference between the two solutions did not increase exponentially with the data, and the maximum time difference did not exceed 0.7 s. Therefore, the time cost advantage of IMU without the introduction of magnetometer was not significant, and the data solved by the gradient descent method with magnetometer correction was finally selected for motion interpolation and dynamic projection fusion.

Table 3. Time consumption of processing data with different filters.

Number of Data Groups	Gradient Descent		Complementary Filtering	
	AHRS	IMU	AHRS	IMU
6624	1095.45 ms	1091.95 ms	1376.60 ms	1123.53 ms
66,240	8885.37 ms	8766.04 ms	8946.30 ms	8924.70 ms
662,400	87,297.87 ms	86,925.46 ms	87,856.31 ms	87,235.14 ms

3.2. Experimental Results and Analysis of Projection Fusion

The 3D model of the stadium in the south campus of Zhengzhou University is loaded on CesiumJS Earth, and the location and parameters of the view body are passed into the cell phone to shoot the video, construct the view cone and build the depth light source camera according to the corresponding parameters at the view point location, obtain the depth image within the viewing frustum, and pass it into the shader for post-processing of the projection texture mapping. The single image projection texture mapping effect is shown in Figure 9.

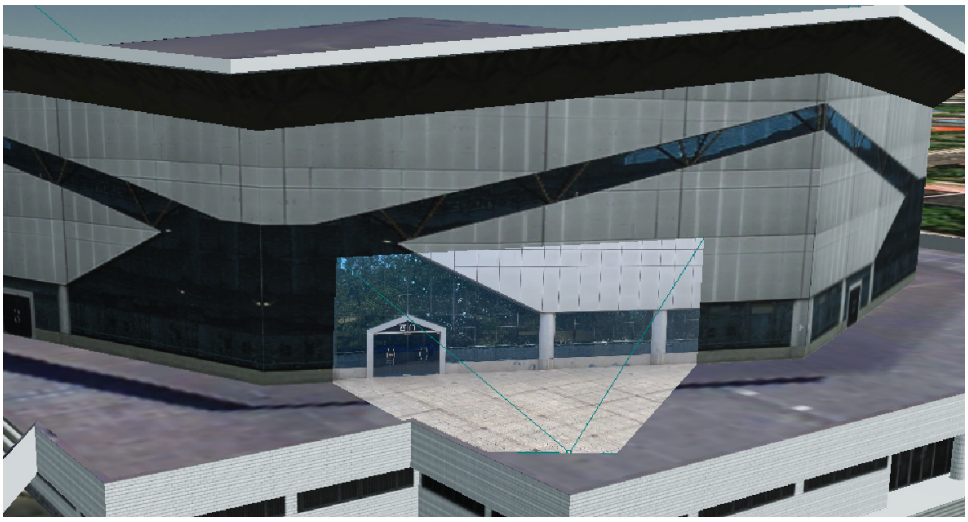

Figure 9. Experimental Results of Image Projection Texture Mapping.

The Euler angles and GPS coordinates obtained from the gradient descent method attitude solution are imported into the browser, and Figure 10 shows the change in the attitude and position of the phone in the first 14 s.

The dynamic projection algorithm is applied for the projection. The experimental results are shown in Figure 11, which shows the dynamic projection fusion effect for different viewpoints and at different moments. It can be clearly seen that the position of the viewpoint and the pose of the viewing frustum are changing with time. It can also be seen that there are some deviations in the accuracy of the projection affected by various errors.

In order to visually detect the homonymous feature points, Figure 12 turns down the transparency of the projected video so that the texture of the video and the texture of the model can be seen at the same time. By measuring the texture deviation of the homonymous points, the error of the dynamic projection is larger compared to the static projection. The error of the view cone midline is only affected by the error of the pitch and yaw angles and is centered on the midline; the further away from the midline, the more the fragment is affected by the roll angle error (see the red box on the left side of the projection in Figure 12b,c). At the same time, the deviation of image fusion is also affected by the projection depth, i.e., the farther the fragment is from the viewpoint and the deeper it is projected, the relatively larger the Euler angular error it suffers (see the blue boxes on the right side of plots a, b, and c and on the left side of the projection of plot d). By measuring

the deviation of the same name point, the projection error of the apparent center line is around ±1 m. In the video edge slice element, where the relative error is large, the error is between 2 and 3 m. The error contains not only the Euler angle solution error, but also GPS positioning error, timing correspondence error, and related model error.

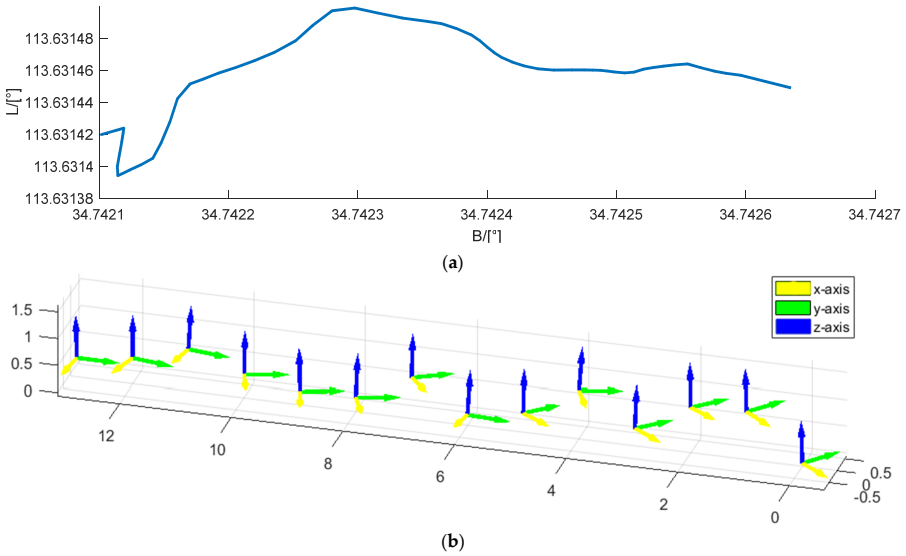

Figure 10. The first 14 s of the phone's position and posture change. (**a**) The two-dimensional moving trajectory of the phone in the first 14 s with latitude as the horizontal coordinate and longitude as the vertical coordinate; (**b**) the change in the phone's posture and relative height.

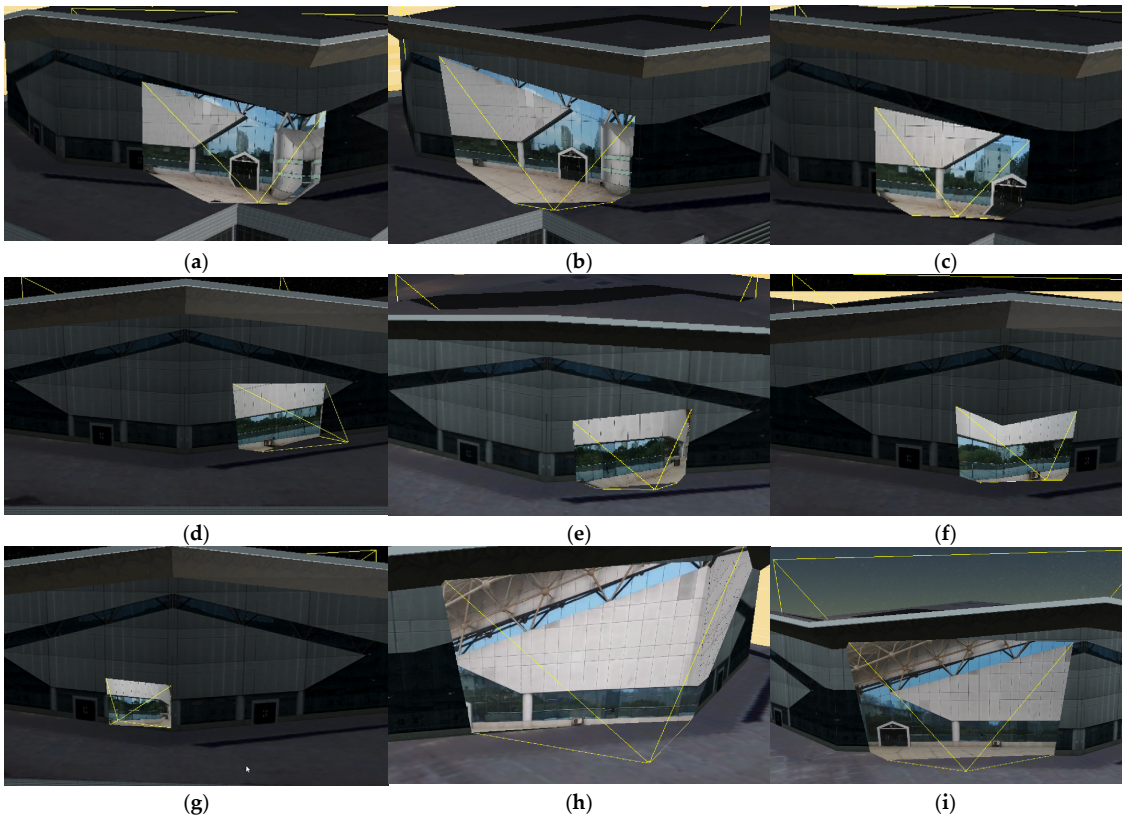

Figure 11. Experimental results of dynamic projection texture mapping at different moments. (**a**–**i**) show the dynamic fusion results in chronological order for the different views.

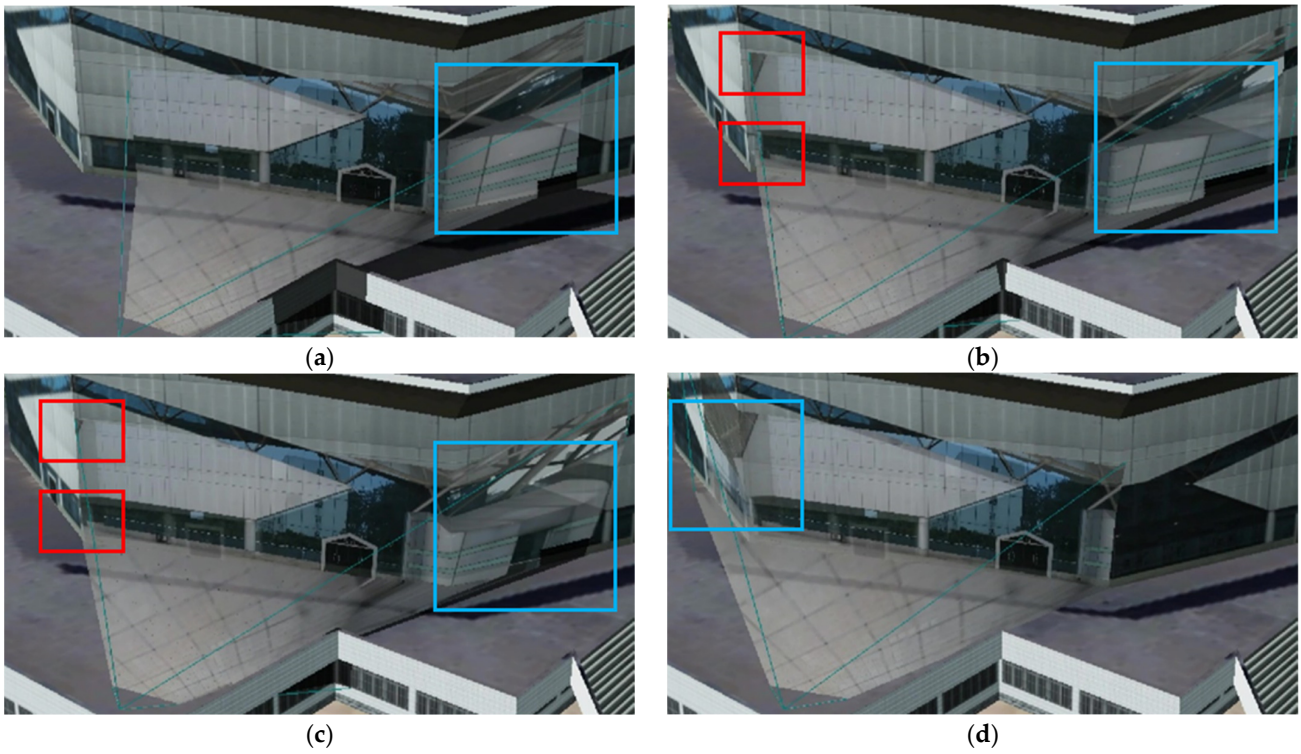

Figure 12. Experimental results of dynamic projection texture mapping at different moments under low transparency. (**a**–**d**) show the results of the dynamic fusion at different moments of low transparency. The red boxes show the error characteristics with respect to the distance from the line of sight. The blue boxes indicate the error characteristics related to the depth of projection.

The fusion efficiency is indexed by the frame rate and playback latency displayed on the 3DGIS. In this paper, the fusion latency and loading frame rate of dynamic fusion for 60 s are counted, as shown in Figure 13, the average latency of one-minute fusion is 28.21 ms and the average rendering frame rate is 34.9 fps. When the frame rate is less than 15 fps, the subjective perception of the human eye drops sharply and the incoherence of the picture is immediately apparent; when it is greater than 15 fps, the subjective perception of the human eye does not differ much and is basically at a higher high level [46]. The fusion method in this paper enables a frame rate of 30 fps or more, which is greater than the video shooting frame rate of 27.9 fps, allowing the human eye to watch the video coherently and comfortably.

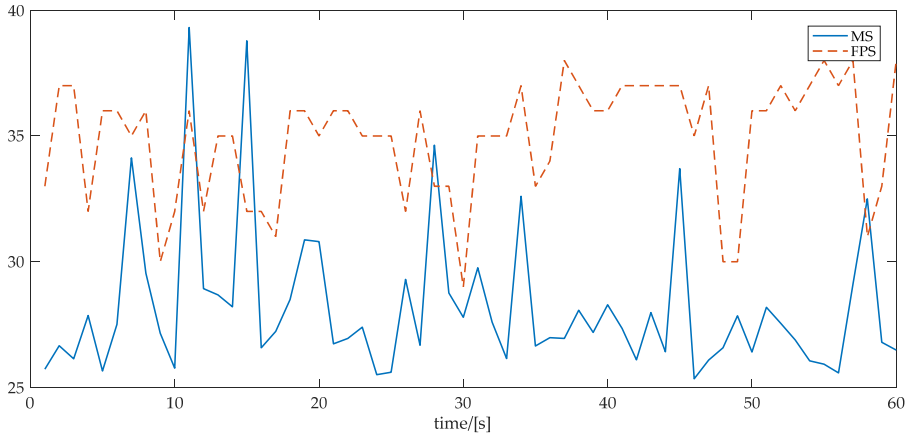

Figure 13. Dynamic fusion of 60 s delay (MS) and frame rate (fps).

4. Discussion

The fusion experiments are mainly performed on the computer. The cell phone is used to collect video data, attitude data and position data. These data are transferred offline to the computer, and the attitude data are solved and input to Cesium along with the position parameters and the collected video data for dynamic fusion of 3DGIS and video textures. The experiment is mainly divided into two parts, one of them the influence of magnetometer on attitude solution accuracy. The experiment uses quantitative analysis to perform a horizontal comparison for four methods with different systems. The experiment uses quantitative analysis to compare the four methods with different systems in terms of the accuracy and efficiency of the solutions. It is determined that the solution efficiency of different methods is almost the same. Thus, by using accuracy as the selection criterion, the gradient descent method of AHRS is more suitable for the attitude solution of cell phones. The experimental procedure is relatively scientific and reasonable. Another one is a dynamic projection fusion experiment. This experiment is conducted on the browser, loading the 3D model, passing in and recording the parameters in the browser variables, parsing the video frames, and projecting the video images frame by frame onto the model according to the corresponding positions and poses, providing the video texture for the model. The experiment verifies the feasibility of the method by taking the frame rate and playback delay of the video displayed on the 3DGIS as indicators to meet the requirements for comfortable viewing of the video by human eyes.

This paper proposed a dynamic fusion algorithm for mobile video with multi-system integration. Compared with previous studies, the following are the contributions of the present work:

(1) A complete set of theoretical methods for dynamic fusion of mobile video and 3D GIS is proposed.
(2) Comparison of the cell phone to an aircraft, introducing the Air Attitude Resolution System (AHRS) to solve the cell phone attitude and comparing the accuracy and efficiency of the solution is performed.
(3) Proposal of texture mapping method for dynamically building depth cameras which meets the requirements of comfortable viewing by human eyes while dynamically fusing, and expands the video sources fused with 3DGIS.

Notably, the lens distortion is not discussed in this paper. It is because the sensor error and attitude resolution error have a greater impact on the accuracy of the fusion effect, and the timing correspondence error and lens aberration have a smaller negative impact. The impact of timing correspondence error and lens distortion on the fusion effect is not considered until a reasonable optimization of the larger errors is performed. Therefore, the follow-up work of this study in terms of accuracy should focus on the optimization of cell phone sensor error and attitude resolution error. In terms of application, the focus should be on accessing the real-time video captured by cell phones and 3DGIS for fusion.

Although the video lacks real-time and the experiment does not achieve high-precision dynamic fusion, the research still has a broad application prospect. In the law enforcement supervision of land resources, videos can be recorded with mobile devices and batch transferred into 3DGIS for land identification, management of illegal houses, monitoring of green area, etc. In the law enforcement of urban security and police, the dynamic fusion management of PTZ cameras for security management can be performed to recognition of dynamic targets in law enforcement recorders, capture objects in video frames, locate them in 3DGIS and perform retrospective restoration of target trajectories. In the application to ubiquitous data, the use of short video software such as TikTok can be used for tracking and evaluation of captured emergencies, vehicle camera video for monitoring the health of roads and city parts, etc.

5. Conclusions

In order to realize the dynamic fusion of mobile video and 3DGIS, this paper integrates various algorithms and proposes a projection texture mapping method to build a dynamic depth camera to dynamically fuse video data, GPS position data and pose data with 3DGIS. The feasibility of the method is experimentally verified, and the accuracy and efficiency of dynamic fusion are quantitatively analyzed. The method solves the problem of dynamic projection of mobile captured video and improves the wide range of video sources fused with 3DGIS. The method is implemented based on front-end browser and has cross-platform and portability. It can be widely used in the direction of government land titling and real-time supervision of mobile video sources, 3D scene restoration of public security law enforcement record videos, tracking and restoration of emergencies captured by short video software such as TikTok, real-time mapping inspection and search and rescue of UAV flight videos, and AR navigation. However, the method does not optimize the sensor error, attitude resolution error, lens distortion and timing correspondence error, and there is still some room for improvement in terms of accuracy.

Author Contributions: Conceptualization, Ge Zhu; methodology, Ge Zhu and Hongwei Li; software, Huili Zhang; validation, Ge Zhu, Juan Lei and Linqing He; formal analysis, Yirui Jiang; resources, Hongwei Li; writing—original draft preparation, Ge Zhu; writing—review and editing, Ge Zhu and Hongwei Li; visualization, Linqing He; supervision, Yirui Jiang and Hongwei Li. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by "Research on Smart City Management Department/Event Intelligent Identification and Extraction Method Based on Multi-modal Data Fusion" of Henan Province Science and Technology Tackling Program Project. (2022 Henan Province, Project No. 222102320220); "Research on Machine Map Theory and Modeling Methods" of Key Program of National Natural Science Foundation of China. (National Natural Science Foundation of China, grant number: 42130112).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The first author may provide all data supporting the findings of this study upon reasonable request.

Acknowledgments: We sincerely thank each anonymous reviewer who provided comments and suggestions.

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

References

1. Debevec, P.E.; Taylor, C.J.; Malik, J. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *SIGGRAPH '96, Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, New Orleans, LA, USA, 4–9 August 1996*; University of California: Berkeley, CA, USA, 1996; pp. 11–20. [CrossRef]
2. Sawhney, H.S.; Arpa, A.; Kumar, R.; Samarasekera, S.; Aggarwal, M.; Hsu, S.; Nister, D.; Hanna, K. Video Flashlightsreal time rendering of multiple videos for immersive model visualization. In Proceedings of the Thirteenth Eurographics Workshop on Rendering (2002), Pisa, Italy, 26–28 June 2002; pp. 157–168. [CrossRef]
3. Chen, Y.Y.; Huang, Y.H.; Cheng, Y.C.; Chen, Y.S. Integration of Multiple Views for a 3-D Indoor Surveillance System. *Information* **2010**, *13*, 2039–2057.
4. Xie, Y.J.; Wang, M.Z.; Liu, X.J.; Wu, Y.G. Surveillance Video Synopsis in GIS. *ISPRS Int. Geo-Inf.* **2017**, *6*, 19. [CrossRef]
5. Xie, Y.J.; Wang, M.Z.; Liu, X.J.; Wang, X.; Wu, Y.G.; Wang, F.Y.; Wang, X.Z. Multi-camera video synopsis of a geographic scene based on optimal virtual viewpoint. *Trans. GIS* **2022**, *26*, 1221–1239. [CrossRef]
6. Cui, X.; Khan, D.; He, Z.; Cheng, Z. Fusing surveillance videos and three-dimensional scene: A mixed reality system. *Comput. Animat. Virtual Worlds* **2022**, *34*, e2129. [CrossRef]
7. Abrams, A.D.; Pless, R.B. Webcams in context: Web interfaces to create live 3D environments. In Proceedings of the 18th ACM International Conference on Multimedia, Firenze, Italy, 25 October 2010; pp. 331–340.
8. Chao, H. Design and Implementation of Three-Dimensional PoliGeographic Information System. Master's Thesis, China University of Geosciences, Beijing, China, 2019.
9. Shihai, Z. Reflections on "Fragmentation" and "Pictorialization" of Reading in the Digital Age. *Publ. Res.* **2016**, 62–65. [CrossRef]

10. Katkere, A.; Moezzi, S.; Kuramura, D.Y.; Kelly, P.; Jain, R. Towards video-based immersive environments. *Multimed. Syst.* **1997**, *5*, 69–85. [CrossRef]
11. Wang, Y.; Krum, D.M.; Coelho, E.M.; Bowman, D.A. Contextualized videos: Combining videos with environment models to support situational understanding. *IEEE Trans. Vis. Comput. Graph.* **2007**, *13*, 1568–1575. [CrossRef]
12. Milosavljevic, A.; Dimitrijevic, A.; Rancic, D. GIS-augmented video surveillance. *Int. J. Geogr. Inf. Sci.* **2010**, *24*, 1415–1433. [CrossRef]
13. Sebe, I.O.; Hu, J.; You, S.; Neumann, U. 3D video surveillance with Augmented Virtual Environments. In Proceedings of the First ACM SIGMM International Workshop on Video Surveillance, Berkeley, CA, USA, 2 November 2003; pp. 107–112.
14. Kim, K.; Oh, S.; Lee, J.; Essa, I. Augmenting aerial earth maps with dynamic information from videos. *Virtual Real.* **2011**, *15*, 185–200. [CrossRef]
15. Milosavljevic, A.; Rancic, D.; Dimitrijevic, A.; Predic, B.; Mihajlovic, V. Integration of GIS and video surveillance. *Int. J. Geogr. Inf. Sci.* **2016**, *30*, 2089–2107. [CrossRef]
16. Jian, H.D.; Liao, J.J.; Fan, X.T.; Xue, Z.X. Augmented virtual environment: Fusion of real-time video and 3D models in the digital earth system. *Int. J. Digit. Earth* **2017**, *10*, 1177–1196. [CrossRef]
17. Gay-Bellile, V.; Lothe, P.; Bourgeois, S.; Royer, E.; Naudet Collette, S. Augmented reality in large environments: Application to aided navigation in urban context. In Proceedings of the 9th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Seoul, Republic of Korea, 13–16 October 2010; Science & Technology Papers. pp. 225–226. [CrossRef]
18. Wenjie, Z.; Qingsong, Y.; Min, Z.; Jun, H. Research on GIS-based video surveillance system. *Comput. Eng. Des.* **2011**, *32*, 745–748. [CrossRef]
19. Han, L.T.; Huang, B.H.; Chen, L. Integration and Application of Video Surveillance System and 3DGIS. In Proceedings of the 18th International Conference on Geoinformatics, Beijing, China, 18–20 June 2010.
20. Binghu, H.; Litao, H.A.N.; Long, C. Intergration and application of video surveillance and 3DGIS. *Comput. Eng. Des.* **2011**, *32*, 728–731. [CrossRef]
21. Wu, Z.; Chang, Y.; Li, Q.; Cai, R. Innovative Application of Tunnel Operation Management Based on Three-Dimensional Video Fusion. *Tunn. Constr.* **2022**, *42*, 154–161.
22. Zhong, Z.; Ming, M.; Yi, Z. massive video integrated mixed reality technology. *ZTE Technol. J.* **2017**, *23*, 6–9. [CrossRef]
23. Zhang, X.; Zhou, Y.; Shi, X.; Luo, X.; Gu, Y. A method of panorama stitching and spatialization for speed dome camera video. *Sci. Surv. Mapp.* **2022**, *47*, 203–211. [CrossRef]
24. Zhou, Y.; Cao, M.J.; You, J.D.; Meng, M.; Wang, Y.H.; Zhou, Z. MR Video Fusion: Interactive 3D Modeling and Stitching on Wide-baseline Videos. In Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology (ACM VRST), Tokyo, Japan, 28 November–1 December 2018.
25. Abrams, A.; Fridrich, N.; Jacobs, N.; Pless, R. Participatory integration of live webcams into GIS. In Proceedings of the 1st International Conference and Exhibition on Computing for Geospatial Research & Application, Washington, DC, USA, 1 January 2010; p. 9.
26. Zhou, Y.; Meng, M.; Wu, W.; Zhou, Z. Virtual-reality video fusion system based on video model. *J. Syst. Simul.* **2018**, *30*, 2550–2557. [CrossRef]
27. Fan, Z. Research on Registration and Rendering Methodof Video to Enhance 3D Scene. Ph.D. Thesis, Wuhan University, Wuhan, China, 2014.
28. Zexi, N.; Xujia, Q.; Jiazhou, C. video fusion method based on 3D scene. *Comput. Sci.* **2020**, *47*, 281–285. [CrossRef]
29. Liu, Z.; Dai, Z.; Li, C.; Liu, X. A fast fusion object determination method for multi-path video and three-dimensional GIS scene. *Acta Geod. Cartogr. Sin.* **2020**, *49*, 632–643. [CrossRef]
30. Liu, X.; Wang, L.; Wu, C.; Huang, J.; Wei, M. Design and implementation of real scene fusion system based on 3D GIS. *Bull. Surv. Mapp.* **2021**, *4*, 141–145. [CrossRef]
31. Neumann, U.; Suya, Y.; Jinhui, H.; Bolan, J.; JongWeon, L. Augmented virtual environments (AVE): Dynamic fusion of imagery and 3D models. In Proceedings of the IEEE Virtual Reality 2003, Los Angeles, CA, USA, 22–26 March 2003; pp. 61–67. [CrossRef]
32. Xiaolong, W. a fast method to build 3D GIS platform Based on Webgl. *Henan Sci. Technol.* **2022**, *41*, 20–23. [CrossRef]
33. Fang, M.; Luo, N.; Xu, Y.; Qi, P. Research on Integrated Visualization of BIM and Real-Scene 3D Model Based on Cesium. *J. Geomat.* **2022**, *47*, 111–114. [CrossRef]
34. Madgwick, S.O.H.; Harrison, A.J.L.; Vaidyanathan, R. IEEE Estimation of IMU and MARG orientation using a gradient descent algorithm. In Proceedings of the IEEE International Conference on Rehabilitation Robotics (ICORR)/International Neurorehabilitation Symposium (INRS)/International Conference on Virtual Rehabilitation (ICVR), Zurich, Switzerland, 27 June–1 July 2011.
35. Liu, X.; Zhang, S.; Li, L.; Lin, X. Quaternion-based algorithm for orientation estimation from MARG sensors. *J. Tsinghua Univ. Sci. Technol.* **2012**, *52*, 627–631. [CrossRef]
36. Ren, H.L.; Kazanzides, P. Investigation of Attitude Tracking Using an Integrated Inertial and Magnetic Navigation System for Hand-Held Surgical Instruments. *IEEE-ASME Trans. Mechatron.* **2012**, *17*, 210–217. [CrossRef]
37. Kok, M.; Hol, J.D.; Schon, T.B. Using Inertial Sensors for Position and Orientation Estimation. *Found Trends®Signal Process.* **2017**, *11*, 92. [CrossRef]
38. Qi, W.; Liu, N.; Su, Z.; Qiao, L.; Wang, J. Conjugate Gradient Based Method for Attitude Calculation of MARG Sensor. *Electron. Opt. Control* **2022**, *29*, 13. [CrossRef]

39. De Haan, G.; Scheuer, J.; de Vries, R.; Post, F.H. Egocentric Navigation for Video Surveillance in 3D Virtual Environments. In Proceedings of the IEEE Symposium on 3U User Interfaces, Lafayette, LA, USA, 14–15 March 2009; p. 103.
40. Okaniwa, S.; Nasri, A.; Lin, H.W.; Abbas, A.; Kineri, Y.; Maekawa, T. Uniform B-Spline Curve Interpolation with Prescribed Tangent and Curvature Vectors. *IEEE Trans. Vis. Comput. Graph.* **2012**, *18*, 1474–1487. [CrossRef] [PubMed]
41. Pu, Y.; Shi, Y.; Lin, X.; Zhang, W.; Chen, Z. Joint motion planning of industrial robot based on hybrid polynomial interpolation. *J. Northwest. Polytech. Univ.* **2022**, *40*, 84–94. [CrossRef]
42. Debevec, P.; Yizhou, Y.; Borshukov, G. Efficient view-dependent image-based rendering with projective texture-mapping. In *Rendering Techniques '98, Proceedings of the Eurographics Workshop in Vienna, Austria, 29 June—1 July 1998*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 105–116. [CrossRef]
43. Hsu, S.; Samarasekera, S.; Kumar, R.; Sawhney, H.S. Pose estimation, model refinement, and enhanced visualization using video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662), Hilton Head, SC, USA, 15 June 2000; Volume 481, pp. 488–495. [CrossRef]
44. Pan, C.W.; Chen, Y.S.; Wang, G.P. Virtual-Real Fusion with Dynamic Scene from Videos. In Proceedings of the International Conference on Cyberworlds (CW), Chongqing, China, 28–30 September 2016; pp. 65–72.
45. Chao, L. Research on Attitude Measurement Algorithm of a Moving Vehicle Based on Marg Sensor. Master's Thesis, Shandong University, Jinan, China, 2017.
46. Ou, Y.F.; Liu, T.; Zhao, Z.; Ma, Z.; Wang, Y. IEEE Modeling the Impact of Frame Rate on Perceptual Quality of Video. In Proceedings of the 15th IEEE International Conference on Image Processing (ICIP 2008), San Diego, CA, USA, 12–15 October 2008; pp. 689–692.