International Journal of Geo-Information

MDPI

Article

Orthographic Video Map Generation Considering 3D GIS View Matching

Xingguo Zhang [1,*], Xiangfei Meng [1], Li Zhang [2], Xianguo Ling [1] and Sen Yang [1]

[1] School of Geographic Sciences, Xinyang Normal University, Xinyang 464000, China; mengxaingfei@xynu.edu.cn (X.M.); lxg006@xynu.edu.cn (X.L.); yangsen@xynu.edu.cn (S.Y.)
[2] School of Physics and Electronic Engineering, Xinyang Normal University, Xinyang 464000, China; zhangli0923@xynu.edu.cn
* Correspondence: zhangxingguo@xynu.edu.cn

Abstract

Converting tower-mounted videos from perspective to orthographic view is beneficial for their integration with maps and remote sensing images and can provide a clearer and more real-time data source for earth observation. This paper addresses the issue of low geometric accuracy in orthographic video generation by proposing a method that incorporates 3D GIS view matching. Firstly, a geometric alignment model between video frames and 3D GIS views is established through camera parameter mapping. Then, feature point detection and matching algorithms are employed to associate image coordinates with corresponding 3D spatial coordinates. Finally, an orthographic video map is generated based on the color point cloud. The results show that (1) for tower-based video, a 3D GIS constructed from publicly available DEMs and high-resolution remote sensing imagery can meet the spatialization needs of large-scale tower-mounted video data. (2) The feature point matching algorithm based on deep learning effectively achieves accurate matching between video frames and 3D GIS views. (3) Compared with the traditional method, such as the camera parameters method, the orthographic video map generated by this method has advantages in terms of geometric mapping accuracy and visualization effect. In the mountainous area, the RMSE of the control points is reduced from 137.70 m to 7.72 m. In the flat area, it is reduced from 13.52 m to 8.10 m. The proposed method can provide a near-real-time orthographic video map for smart cities, natural resource monitoring, emergency rescue, and other fields.

Keywords: GIS; video; orthographic video; image matching; DEM

check for updates

Academic Editors: Wolfgang Kainz and Sisi Zlatanova

Received: 15 August 2025
Revised: 30 September 2025
Accepted: 9 October 2025
Published: 13 October 2025

Citation: Zhang, X.; Meng, X.; Zhang, L.; Ling, X.; Yang, S. Orthographic Video Map Generation Considering 3D GIS View Matching. *ISPRS Int. J. Geo-Inf.* **2025**, *14*, 398. https://doi.org/10.3390/ijgi14100398

Copyright: © 2025 by the authors. Published by MDPI on behalf of the International Society for Photogrammetry and Remote Sensing. 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 (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Rapid developments of new-generation information technology have demonstrated significant potential in smart city construction, emergency response, and dynamic natural resource monitoring [1]. Surveillance videos contain rich semantic content, from which multi-dimensional spatiotemporal features can be extracted using intelligent algorithms. These real-time, objective, and large-scale spatiotemporal data are the basis for generating an accurate understanding of geographic scenes. However, camera networks mostly use a nine-split screen layout, and each camera is independent of the others. This makes it difficult to implement unified management and collaborative processing, posing significant challenges for the analysis of complex events [2]. GIS provides a unified spatial framework that facilitates the management of spatiotemporal data on a global scale, incorporating videos into a unified geographic spatial reference and realizing the integration

of video surveillance and GIS, namely video GIS [3], which has become an important research hotspot.

Surveillance videos use a perspective view, while maps and remote sensing images in the GIS field often use an orthographic view, resulting in significant differences between the two views. The integration of video and geographic space is still an urgent problem that needs to be solved. If the video is converted into an orthographic perspective, it will be consistent with the remote sensing (RS) image and map in terms of geometric position, which will be conducive to real-time perception and analysis of geographic scenes. In recent years, in the field of natural resources monitoring, a large number of tower-based cameras have appeared to monitor cultivated land, forests, disasters, etc. Compared with traditional cameras, these cameras are mostly located in natural scenes, have a higher position (common tower height is 10 m to 40 m), are rotatable (horizontally 360 degrees, vertically more than 100 degrees), have better zoom (with dozens of times zoom), and have a wider observation field of view [4]. In the scenarios of tower-based camera monitoring, the most commonly used data for three-dimensional scenes are the DEM and high-resolution RS. Meanwhile, due to historical and installation reasons, for most cameras, only partial camera parameters are available, and the DEM has low horizontal and vertical resolution. Existing studies mostly use the homography method [5] and the camera parameter mapping method [6]. The homography method is suitable for flat areas, while the camera parameter method requires extremely high camera parameters and high-precision 3D models. Therefore, it is difficult to use the existing homography and camera modeling methods to meet the need of generating an orthographic video map.

This paper proposes a strategy from rough mapping to precise mapping; that is, based on the camera parameter method, the processing of 3D GIS view and video frame matching is added, which greatly improves the geometric accuracy of the orthographic video map. This method includes three major processes, namely geometric alignment between video and 3D GIS, view matching, and video orthorectification. The core of this method is to achieve accurate matching between video frames and 3D GIS views, which is accomplished using image matching methods on the basis of coarse matching between video frames and 3D GIS.

2. Related Work

There are two main methods for converting surveillance videos into an orthographic video map, namely homography and camera parameter methods. The former is applicable only to flat areas, while the latter requires detailed camera parameters and a high-precision 3D model. This paper used an image matching algorithm; therefore, we also conducted a literature review of the algorithm. A summary of related research in this field is presented below.

2.1. Homography Method

The homography method is one of the commonly used methods in the field of video GIS. This method is suitable for monitoring flat areas such as parking lots and playgrounds. The process is as follows: firstly, select four or more control points in the flat region of the video frame; then, find the corresponding points in the high-resolution RS and obtain their geographic coordinates; finally, calculate the homography matrix H, which transforms image coordinates into geographic coordinates. Image-to-geographic coordinate conversion is performed using H, while the reverse conversion is obtained through its inverse matrix. As demonstrated by Xie Y et al., targets detected in videos such as people, crowds, cars, etc., are projected onto corresponding locations in 2D maps or RS images using the homography method [7,8]. Zhang X. et al. employed auxiliary planes to solve the

homography matrix, enabling the projection of human head positions onto indoor maps [9]. Shao Z et al. tried to match the video image with the two-dimensional reference images using feature point matching and dynamically solved the homography matrix [10]. For complex monitoring scenes in cities, Zhang X et al. proposed a multi-plane constrained homography method, which essentially maps video frames into different regions based on elevations [11]. The homography method has a wide range of applications in urban scenes. However, it is difficult to apply in natural scenes, as the terrain is often highly irregular, making it challenging to compute a unified homography matrix.

2.2. Camera Parameter Method

Compared with the homography method, the camera parameter method has a wider range of adaptability scenarios and no planar area constraints, but it has stricter requirements on camera parameters. The process is as follows: firstly, calibrate the internal and external parameters of the camera; then, set the corresponding parameters of the virtual camera in 3D GIS; finally, video frames are mapped into the 3D virtual geographic scene.

Camera calibration involves calculating the camera's internal and external parameters through image coordinates and corresponding spatial rectangular coordinates. Traditional methods mainly include reference-based methods, active vision methods, and self-calibration methods. The target image corner points are extracted as control points based on the reference camera calibration method, such as Zhang Zhengyou's calibration method based on a checkerboard calibration plate [12] and the Tsai two-step method [13], which are classic methods representative of this type. The camera calibration method based on active vision determines the internal and external parameters of the camera by manually controlling the camera or target to make special movements, such as calibration based on pure rotational motion [14], calibration based on three-axis translational motion [15,16], etc. The camera self-calibration method utilizes geometric consistency constraints between corresponding points across multiple image frames [17,18] to compute the fundamental matrix of the camera [19], which is independent of scene structure and motion information [20]. They include direct solution of the Kruppa equation, absolute quadratic curve and absolute quadratic surface method [21,22], and hierarchical stepwise calibration method under variable internal parameters [23]. With the rapid development of computer technology, intelligent calibration has become the mainstream, mainly including camera calibration methods based on the error backpropagation neural network [24], camera calibration methods based on the multilayer perceptron neural network [25,26], and camera calibration methods based on the convolutional neural network [27]. Zhou F et al. projected video frames in real time onto a 3D scene through texture mapping technology [28]. Liu Z et al. mapped multiple videos into three-dimensional scenes to achieve the efficient retrieval of targets [29]. The core camera parameter method based on geographic mapping is the alignment of video frames with corresponding three-dimensional models. However, camera calibration is difficult, especially for large-scale camera networks, which are usually composed of tens of thousands or more cameras. It is time-consuming and labor-intensive in practical applications and difficult to apply. At the same time, it is required that the monitoring scene has a high-precision and realistic 3D model in order to be mapped with video frames. Therefore, the camera parameter method is suitable for small video monitoring networks. The internal and external parameters of the camera are obtained through calibration, measurement, and other methods so as to achieve a good mutual mapping effect. However, it is difficult to be applied in large-scale camera networks, especially in natural scenes.

2.3. Image Matching

Image matching is an important task in computer vision that aims to identify and align the content or structure with the same or similar attributes in two images at the pixel level. In traditional methods, feature extraction and matching algorithms such as SIFT [30], SURF [31], and ORB [32] have been widely used in the matching task between static images. However, in complex natural scenes, these methods are often difficult to use to maintain high robustness and accuracy due to the influence of illumination changes, scale differences, occlusion interference, and perspective changes. In recent years, the development of deep learning has provided a new technical path for feature point matching. Matching models based on convolutional neural networks [33] (CNNs) or Transformer architecture [34] have shown significant advantages in representation ability and adaptability. For example, the SuperPoint and SuperGlue algorithms have realized the application of complete deep neural networks in image feature matching [35]. The surveillance video is a real-time image, while the 3D GIS view is a side view of the scene, which consists of historical remote sensing images and DEM. Because the time span between the two is relatively long, even some geographical features and landforms have changed. Traditional algorithms, such as feature point detection and matching, may yield a low number of matching points or introduce numerous errors. This paper introduces a newer deep learning-based matching algorithm, GIM [36].

The above literature review indicates that the homography method requires the mapping area in the image to be a plane, while the camera modeling method requires precise camera parameters and a high-precision 3D model. In practical applications, especially for large-scale tower-based cameras, accurate internal and external parameters are often lacking. At the same time, the monitoring scenes are mostly natural scenes, resulting in the loss of high-precision 3D model data. In this context, this article introduces view matching technology based on the camera model method to improve the accuracy of video mapping.

3. Research Methodology

This section describes the method of orthographic video map generation in detail, and its technical route is shown in Figure 1. The method consists of three parts, namely video and 3D GIS synchronizing, view matching, and video orthographic generation. The data sources used in this method mainly include DEM, high-resolution RS, and surveillance videos. Firstly, the synchronization between video and 3D GIS view is achieved by calculating camera parameters and aligning the virtual camera in 3D GIS. Then, based on the idea of image matching, automatic view matching is realized between 3D GIS views and video frames, and the homography matrix is solved. Finally, by converting the video frame image coordinates into three-dimensional coordinates in geographic space and generating point clouds, an orthographic video map is finally generated.

3.1. Video and 3D GIS Synchronizing

Camera parameters differ from the geographical concepts of azimuth, inclination, and field of view in 3D GIS and cannot be directly converted. Based on the data of DEM and RS in 3D GIS, the conversion model of camera parameters and virtual camera parameters of 3D GIS is the key to synchronizing video and 3D GIS.

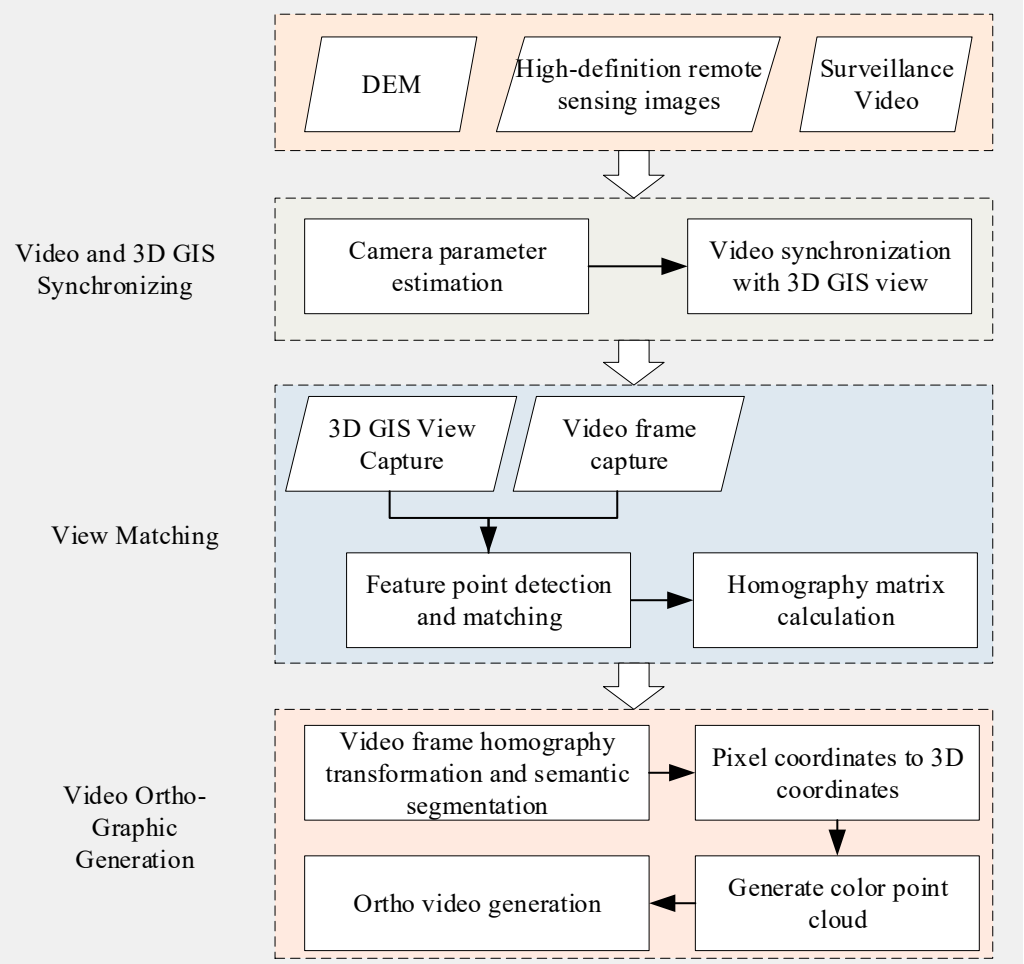


Figure 1. Orthographic video map generation.

3.1.1. Camera Parameter Estimation

Camera (C) parameters include two major types, internal and external parameters. The internal parameters mainly include the principal distance (f), the image principal point O (u_0, v_0), lens distortion, and so on. The external parameters mainly include the camera position coordinates S (Xs, Ys, Zs), azimuth (Ageo), tilt (Tgeo), rotation, and other related parameters. The coordinates S of the camera position are usually captured when the camera is installed, including the latitude and longitude of the camera (lon, lat), altitude (alt), tower height (h), etc. Other external parameters can be obtained by the spatial relationship between high-resolution RS, DEM, and video frames. On the basis of external parameter calculation, different internal parameter estimation methods are adopted according to the different types of monitoring scenarios, namely the vanishing point method is used for structured scenarios, and the 3D GIS view fitting method is used for unstructured scenarios.

For a certain viewpoint of the camera, assume that the C position coordinates are (Xs, Ys, Zs), where Xs and Ys are the projection coordinates of longitude and latitude, respectively, and Zs = alt + h, as shown in Figure 2. Based on the main point O of the video frame and its corresponding point O′ (Xo′, Yo′, alt) in the high-resolution RS, the distance between O′ and the camera's projection coordinates (Xs, Ys) is d, which is calculated as shown in Equation (1). According to d and h, the camera tilt angle Tgeo can be obtained, as shown in Equation (2). The azimuth angle of the camera is Ageo, which is calculated as Equation (3).

$$d = \sqrt{(X_s - X_o')^2 + (Y_s - Y_o')^2} \quad (1)$$

$$T_{geo} = \arctan\frac{d}{h} \tag{2}$$

$$A_{geo} = \arctan\frac{d_x}{d_y} \tag{3}$$

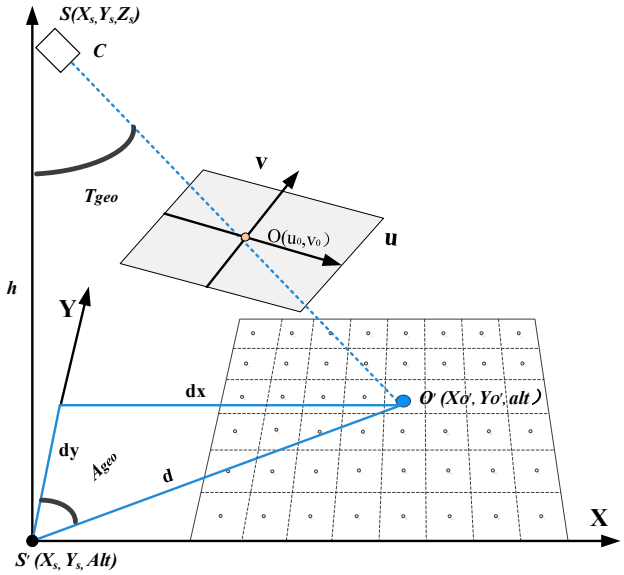


Figure 2. Schematic diagram of camera external parameter calculation.

For the internal parameters, the camera main distance f can be calculated according to the scene type. For structured scenes, the vanishing point method can be used [37]. The two vanishing points are calculated in the horizontal and vertical directions in the image, with coordinates of (u1, v1) and (u2, v2), respectively. The coordinates of the main point O are set as (u0, v0), Wvid is the width of the video frame, and Hvid is the height of the video frame. Then, u0 is calculated as in Equation (4), v0 is calculated as in Equation (5), and the focal length f is calculated as in Equation (6).

$$u_0 = W_{vid}/2 \tag{4}$$

$$v_0 = H_{vid}/2 \tag{5}$$

$$f = \sqrt{-(u_1 - u_0)(u_2 - u_0) - (v_1 - v_0)(v_2 - v_0)} \tag{6}$$

For the unstructured scene, according to the current camera S, Ageo, and Tgeo, set the position Svir, azimuth Avir, and tilt Tvir of the virtual camera in the 3D GIS and dynamically adjust the vertical field of view angle so that the video frames are consistent with the 3D GIS view and the current vertical field of view angle VFOV can be obtained. Then, according to the VFOV and the height of the view Hvir, the camera focal length f corresponding to the current video frame can be calculated, as shown in Formula (7).

$$f = \frac{H_{vir}/2}{\tan(VFOV/2)} \tag{7}$$

If the camera lens distortion is large, lens distortion correction is required.

3.1.2. Video Synchronization with 3D GIS View

In order to obtain the parameters of the 3D GIS virtual camera (C′) corresponding to camera C, it is necessary to analyze the mapping relationship between the parameters of the two cameras. The position, inclination, and azimuth of C′ are similar to that of C, and

the main difference lies in the difference between the camera's f and the field of view (FOV) of the virtual camera C′, as shown in Figure 3.

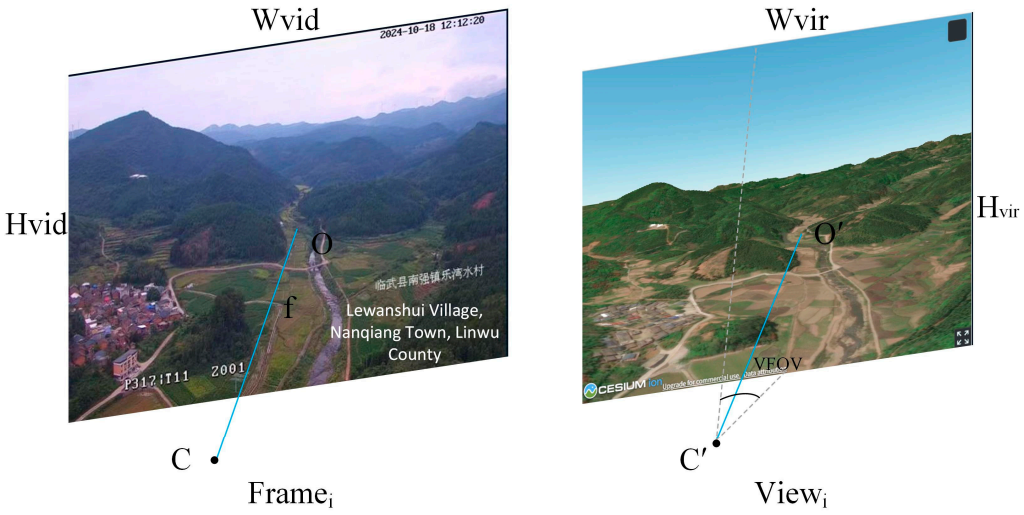


Figure 3. Schematic diagram of synchronization between video frames and 3D GIS views.

A mapping relationship exists between f and FOV; based on f of C, video frame width Wvid, and height Hvid, the vertical field of view angle VFOV of C′ can be calculated, which is shown in Equation (8).

$$\mathrm{VFOV} = 2 \times \mathrm{Arctan}(\mathrm{H_{vir}}/2/\mathrm{f}) \tag{8}$$

Based on the mapping relationship between C and C′ parameters, the synchronization of video frames and 3D GIS view in a certain viewpoint can be realized. For PTZ camera, its attitude changes dynamically, and each attitude has corresponding parameters of Pi, Ti, and Zi values. These correspond to the geospatial parameters Ageoi, Tgeoi, fi, Pi, and Ageoi usually satisfy a linear or piecewise linear relationship, while Ti and Tgeoi, Zi, and fi also satisfy similar relationships. Cameras Ageoi, Tgeoi, and fi correspond to Aviri, Tviri, and VFOVi in 3D GIS. Therefore, for any attitude of PTZ camera, Pi, Ti, and Zi can be used to obtain its virtual camera parameters in 3D GIS so as to realize the synchronization of video and 3D GIS. Conversely, the camera parameters of PTZ camera can be dynamically solved according to the parameters Aviri, Tviri, and VFOVi under 3D GIS, thus enabling dynamic video viewing within 3D GIS.

3.2. View Matching

The video frame is a perspective image of the real world, while the 3D GIS view is a side view of the DEM, RS, and 3D model. Large-scale tower-based cameras are mostly distributed in natural scenes, and the spatial data in 3D GIS mainly consist of publicly available DEM and high-definition remote sensing images. There is a lack of 3D models of land features such as trees and buildings, which poses significant challenges in the accurate mapping of videos. Even when all internal and external parameters of the PTZ camera have been precisely calculated, it is difficult to achieve accurate mapping between the two due to DEM data accuracy issues. The view matching method proposed in this article can effectively improve the accuracy of video mapping, especially in the region where the matching points are located, achieving pixel-level mapping accuracy. On the basis of the coarse mapping in 3.1, this paper proposes an accurate geo-localization method for video considering frame and view matching based on image matching.

3.2.1. Feature Point Detection and Matching

Feature point detection and matching aims to achieve the matching of frame and view. The main process is as follows: (1) According to the method of video and 3D GIS synchronization in 3.1, the video frame $Frame_i$ is used to obtain its corresponding 3D GIS view $View_i$. (2) Feature points are detected in $Frame_i$ and $View_i$, yielding the sets $Fpts_i$ and $Vpts_i$, respectively. (3) Feature point matching is performed between the two point sets, $Fpts_i$ and $Vpts_i$, to obtain the matched point-pair set $FVpts_i$. As shown in Figure 4, the left side is $Frame_i$, the right side is $View_i$, $Fpts_i$ contains seven feature points (Fpt_1 to Fpt_7), and $Vpts_i$ contains seven feature points (Vpt_1 to Vpt_7). Using the feature point matching algorithm, the seven feature points can be matched between the two views.

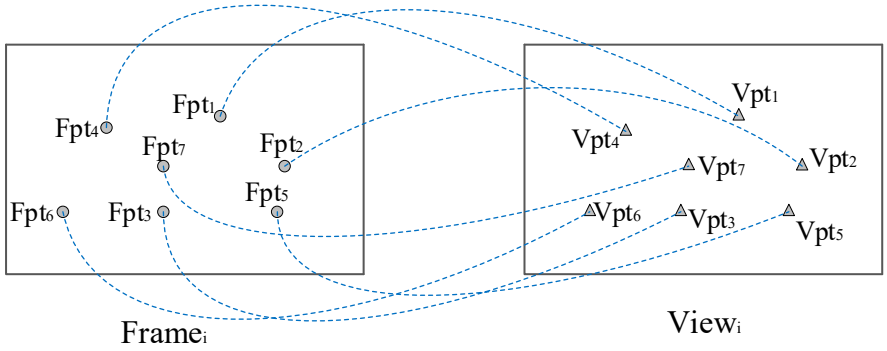


Figure 4. The feature point matching diagram between frame and view. The circles in $Frame_i$ represent the seven control points in the video frame, and the triangles in $View_i$ represent the points with the same name in the corresponding 3D GIS view.

Currently, the commonly used algorithms for feature point detection and matching mainly include SIFT, ORB, and deep learning-based methods. $Frame_i$ and $View_i$ come from different data sources. The former is the video frame captured by the camera, and the latter is the corresponding view rendered by 3D GIS. They are similar in viewpoint but differ in content. It is difficult to realize the matching by using the conventional algorithms for feature point detection and matching. This paper adopts the algorithms based on deep learning, such as GIM, which achieve better matching performance.

3.2.2. Homography Matrix Calculation

In order to realize the mutual conversion between $Frame_i$ pixel coordinates and $View_i$ screen coordinates, it is necessary to calculate the homography matrix Hi. The specific process is as follows: (1) The matched point-pair set $FVpts_i$ is obtained between $Frame_i$ and $View_i$. (2) The optimization algorithms, such as the least squares method and the RANSAC algorithm, are used to solve Hi. (3) Based on Hi, the matrix can realize the mutual conversion between $Frame_i$ and $View_i$ pixel coordinates.

Assume that the coordinates of the pixel point p in $Frame_i$ are (u, v), and the coordinates of the corresponding point P in $View_i$ are (u′, v′); then, the calculation of p to P coordinates is as shown in Equation (9).

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \times \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (9)$$

Based on the H^{-1} matrix, the opposite coordinate transformation can be realized, as in Equation (10).

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix}^{-1} \times \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} \quad (10)$$

Suppose that the homography matrix of Frame1 and View1 is H1, and the pixel coordinates of a point in Frame1 are known to be p1 (u1, v1), then the screen coordinates of the point in View1 can be calculated through the homography matrix H1. Conversely, the corresponding p1 (u1, v1) in Frame1 can be calculated through P1 (u′, v′).

3.3. Video Orthographic Generation

3.3.1. Video Frame Homography Transformation and Semantic Segmentation

Based on the homography matrix H, the mutual conversion of each pixel point pi (u, v) in $Frame_i$ and Pi (u′, v′) in $View_i$ can be performed. Based on the H matrix, $Frame_i$ can be converted to $Frame_i'$. That is, pi (u, v) is converted to the corresponding point pi′ (u, v) in $Frame_i'$ through the H matrix, and $Frame_i'$ is consistent with $View_i$. As shown in Figure 5, the three points (p1, p2, p3) in $Frame_i$ are converted to the corresponding points p1′, p2′, p3′ in $Frame_i'$ through the H matrix. The coordinates of p1′, p2′, p3′ are equal to those of P1, P2, P3 in $View_i$, so as to realize the mutual conversion of the coordinates from $Frame_i$ to $View_i$.

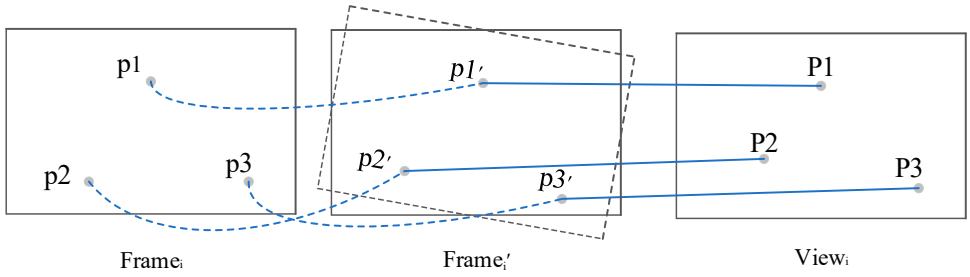


Figure 5. Schematic diagram of video frame homography transformation. The circles in the figure represent the control points in the three views, such as p1, p2, and p3. After homography transformation, their positions become p1′, p2′, and p3′. The dashed lines represent the same-name points in the $Frame_i$ and $Frame_i'$, and the solid lines represent the same-name points in the $Frame_i'$ and $View_i$.

Since there are regions with no data (black areas) in $Frame_i'$, these regions are non-orthorectified regions, which can be automatically removed by semantic segmentation. In this paper, we adopt the DeepLabV3+ semantic segmentation model [38] to classify $Frame_i'$ into three categories, namely sky, regions with no data (black areas), and orthorectified area, and realize the automatic extraction of orthorectified area in $Frame_i'$ through sample collection and model training.

3.3.2. Pixel Coordinates to 3D Coordinates

For each pixel pi′ (u′, v′) in $Frame_i'$, how to obtain its corresponding geospatial 3D coordinates is the key to video orthorectification. The specific process is as follows: (1) pi′ (u′, v′) is consistent with Pi (u′, v′) in $View_i$, and under 3D GIS, each Pi point has corresponding geospatial 3D coordinates Gi (lon, lat, alt). (2) In order to facilitate the subsequent processing, this paper adopts Web Mercator map projection to convert Gi into the corresponding three-dimensional rectangular coordinates Wi (X, Y, Z).

3.3.3. Generate Color Point Cloud

For each pixel point pi in $Frame_i$, its three-dimensional spatial rectangular coordinates Wi and color values Ci (R, G, B) are calculated. The process is as follows: (1) converting

pixel coordinates to 3D coordinates, i.e., pi-pi′-Pi-Gi-Wi; (2) obtaining the RGB value Ci of pixel pi; (3) the corresponding color point cloud of $Frame_i$ is assembled as Ptsci, and each color point is composed of Wi and Ci, i.e., Ptsci = {Wi, Ci}. As shown in Figure 6, suppose that there are two points (p1, p2) in $Frame_i$, whose color values are C1, C2, respectively, which are transformed into corresponding points p1′, p2′ in $Frame_i'$ after the H matrix, and the coordinate values of p1′, p2′ are equal to the screen coordinates P1, P2 in $View_i$. In $View_i$, the geographic three-dimensional coordinates G1 and G2 of P1 and P2 are latitude and longitude. Through Web Mercator map projection, G1, G2 can be converted to the corresponding three-dimensional space right-angle coordinates W1, W2, and finally $Frame_i$ the corresponding three-dimensional geospatial collection of colored point clouds Ptsci = {(W1, C1), (W2, C2). . .(Wi, Ci)}.

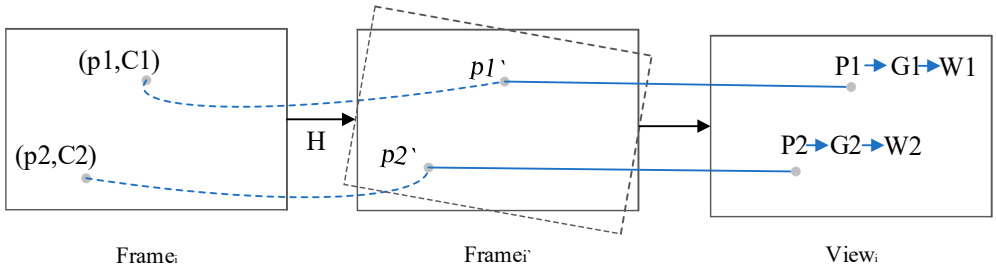


Figure 6. Schematic diagram of color point cloud generation. The circles in the figure represent control points in the three views. The dotted lines represent the same-name points in the $Frame_i$ and $Frame_i'$ views, and the solid lines represent the same-name points in the $Frame_i'$ and $View_i$ views.

3.3.4. Orthorectified Video Generation

Based on the color point cloud collection Ptsci generated above, it is converted into orthophoto Oimgi, and the time-sequenced orthophoto Oimgi is orthophoto video. The conversion process from Ptsci to Oimgi consists of (1) generating a rectangular range region (Range region) based on the coordinate information of the point cloud in Ptsci (the maximum and minimum values of X; the maximum and minimum values of Y); (2) setting the size of the grid cell as D and dividing the Range region into a regular grid (Grid) in accordance with the rows and columns; (3) according to the coordinate value of the point cloud in Ptsci, its row and column number in the Grid can be calculated, and a set of row and column numbers and corresponding colors GC (i, j, R, G, B) can be formed; (4) generate an image based on the row and column numbers of the Grid, and the corresponding color of its pixels is the color in GC, and the image is Oimgi. At the same time, coordinate files corresponding to Oimg_i are generated based on the range region, such as JGW. JGW is a text format file, and there are six components, i.e., the x-direction size of the pixel under the map unit, the rotation item (generally 0.00), the rotation item (generally 0.00), the negative value of the pixel size in the y-direction, representing the map resolution, the translation item (the x-map coordinate of the pixel at the upper-left corner), and the translation item (the y-map coordinate of the pixel at the upper-left corner).

As shown in Figure 7, Ptsc1 on the left contains 15 points. Based on their coordinate values, a rectangular region R1 is generated. R1 is divided into 7 rows and 10 columns with D as the interval. The RGB values of the points in Ptsc1 are assigned to the corresponding grid cells. The same row and column numbers and corresponding RGB images are generated according to this grid. In order to make the Oimg1 image have a spatial reference, a coordinate file with the same name in JGW format is generated, and then Oimg1 can be overlaid on RS or maps.

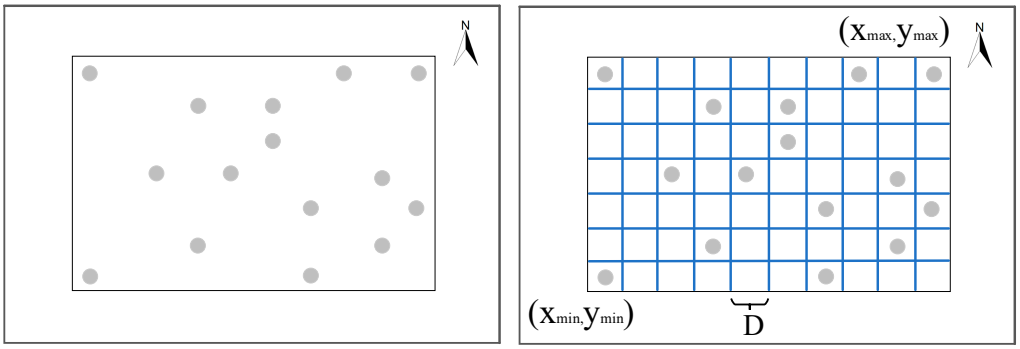


Figure 7. Schematic diagram of orthorectified video generation. The circles in the figure represent the point cloud, and the blue gridlines on the right represents the detailed division of the rectangular area according to the interval D.

4. Experiments and Analysis

4.1. Experimental Environment and Data

The software environment and programming language used in this study were VS2012 C#, ArcGIS10.2, and python3.9. The hardware included GPU RTX 2080Ti, CPU i9-12500H, and 32 GB memory. The data sources mainly include tower-based camera Camera1 in a certain area of Hunan Province, China; Camera2 in a certain area of Henan Province, China, public dataset ASTER GDEM V3, and ESRI online high-definition RS (0.5 m). The geospatial data in this experiment used WGS84 coordinates, and the projection used the Web Mercator projection. The coordinates of the control points in the experiment were all obtained from 3D GIS.

In order to verify the effectiveness of our method under different terrain conditions, this study selected two cameras for experiments. Among them, the monitoring area of Camera1 was relatively undulating, and the monitoring area of Camera2 was relatively flat, as shown in Figure 8.

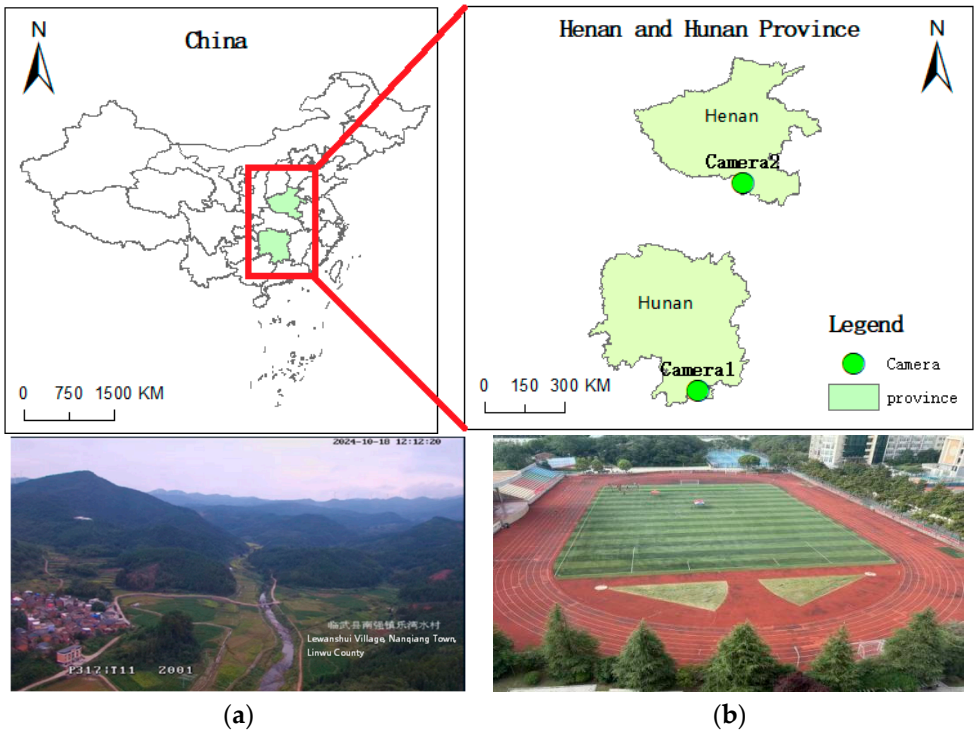


Figure 8. Overview of the study area: (**a**) Camera1 video frame; (**b**) Camera2 video frame.

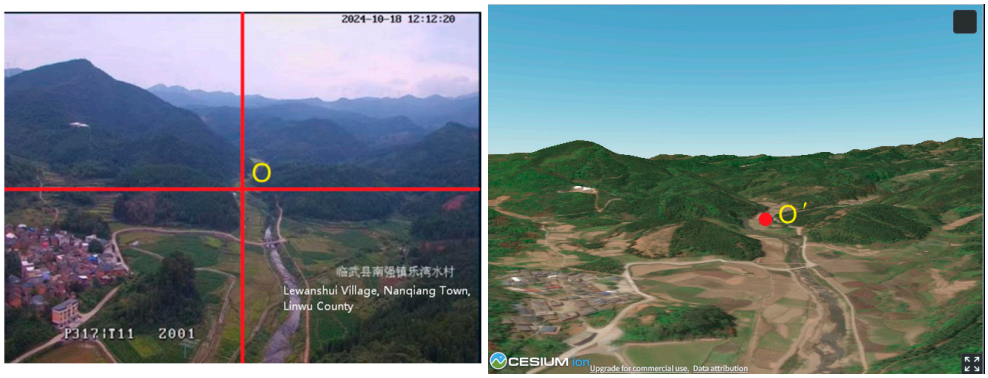
4.2. Experimental Analysis

4.2.1. Synchronization of Video and 3D GIS View

For Camera1 and Camera2, the external and internal parameters of the cameras were calculated according to the method described in 3.1. Given the longitude and latitude (lon, lat), altitude (alt), tower height (h), and video frame size (Wvid, Hvid) of the two cameras, the position S1 of Camera1 was (112.55, 25.18, 420.90), and the position S2 of Camera2 was (114.03, 32.14, 107.35), as shown in Table 1. According to the main point O, find its corresponding point O′ in 3D GIS, with coordinates of (Xo′, Yo′, Ho′), Camera1 is shown in Figure 9, and Camera2 is shown in Figure 10.. Due to the fact that the camera coordinates S and the O′ coordinates in 3D GIS are both geographical coordinates, in order to accurately calculate the azimuth and inclination, they need to be projected onto a map and converted into spatial Cartesian coordinates. Because both monitoring scenes are unstructured scenes, the focal length f cannot be solved by the vanishing point method. This paper achieved synchronization between video frames and 3D GIS views by dynamically adjusting the scaling ratio of the 3D GIS view. The principal distance f was calculated based on the vertical field of view (VFOV) of the current 3D GIS virtual camera. The calculated parameters for the two cameras are presented in Table 2.

Table 1. Camera positions and video frame sizes.

ID	Lon	Lat	Alt	h	Zs	Wvid	Hvid
Camera1	112.55	25.18	408.96	11.94	420.90	752	566
Camera2	114.03	32.14	77.35	30	107.35	1000	750



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

Figure 9. Mapping relationship between image principal point in Camera1 and corresponding point in RS: (**a**) video frame; (**b**) 3D GIS view. The solid red line on the left represents the midline dividing the video frame width and height, and their intersection is the principal point O. The red circle on the right represents the point O′ with the same name as the principal point in the 3D GIS view.

Table 2. Calculation results of camera focal length, tilt angle, and azimuth angle.

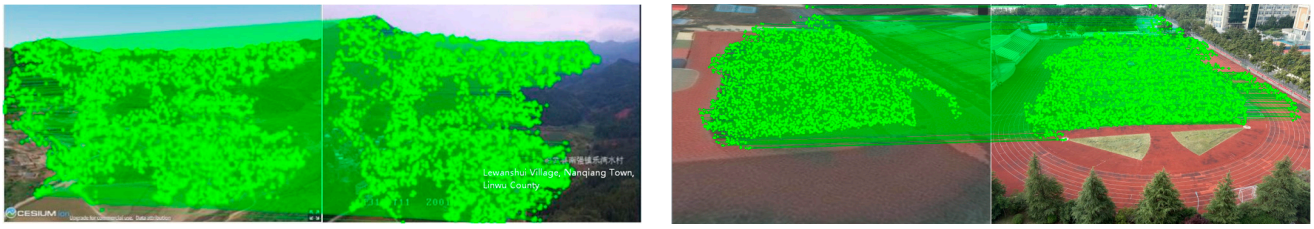
ID	Xo′	Yo′	Ho′	Ageo	Tgeo	f	VFOV
Camera1	112.54	25.17	4292	−142	−7	101.74	60
Camera2	114.03	32.14	76.94	13	−30	136.62	60

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

Figure 10. Mapping relationship between image principal point in Camera2 and corresponding point in RS: (**a**) video frame; (**b**) 3D GIS view. The solid red line on the left represents the midline dividing the video frame width and height, and their intersection is the principal point O. The red circle on the right represents the point O′ with the same name as the principal point in the 3D GIS view.

4.2.2. View Matching Analysis

The deep learning-based GIM algorithm was used to conduct feature point detection and matching between the video frame $Frame_i$ and its corresponding 3D GIS view $View_i$. As shown in Figure 11a, regarding the matching effect of Camera1 view, View1 on the left, and Frame1 on the right, it can be seen that the two views have a large number of feature points and are correctly matched, namely Vpts1 and Fpts1. Based on Fpts1 and Vpts1, the homography matrix H1 was calculated using RANSAC optimization. Figure 11b shows the matching effect of Camera2 view. The homography matrix H2 was calculated using a method similar to Camera1.



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

Figure 11. View matching effect: (**a**) Camera1 feature point detection and matching effect; (**b**) Camera2 feature point detection and matching effect.

4.2.3. Video Orthography

(1) Video frame homography transformation and semantic segmentation

In order to achieve accurate mapping between video frames and 3D GIS views, it is necessary to perform homography transformation on video frames through the homography matrix, as shown in Figure 12a,b. The DeepLabV3+ semantic segmentation model was used to classify the content in the video frame into sky, black no data area, and orthophoto area. Because the sky and no data area belong to the non-orthophoto area, after the homography transformation of the video frame, the non-orthophoto area in video frame was removed, and the effect is shown in Figure 12c,d, Frame1′, and Frame2′, respectively.

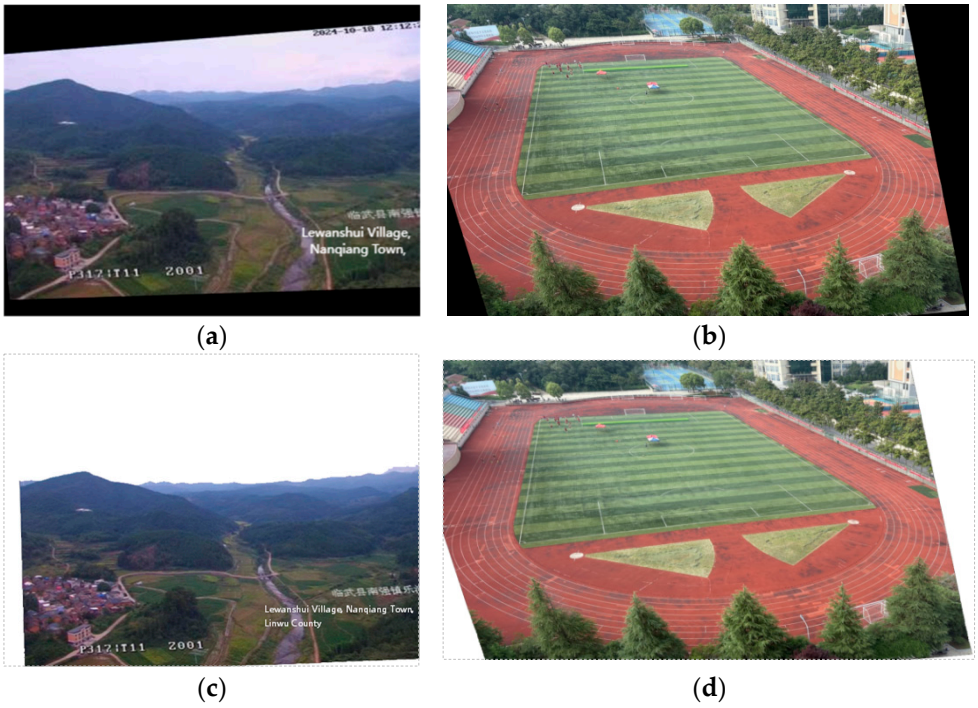


Figure 12. Video frame homography transformation: (**a**) the homography transformation effect of Camera1 video frame; (**b**) the homography transformation effect of Camera2 video frame; (**c**) removal effect of non-orthorectified areas in Camera1 video frames; (**d**) removal effect of non-orthorectified areas in Camera2 video frames.

(2) Orthophoto generation

After the video frames undergo homography transformation and semantic segmentation, the next step is to perform video orthorectification processing. First, because each pixel pi′ (u′, v′) of $Frame_i'$ was consistent with Pi (u′, v′) in $View_i$, a color point cloud Ptsci could be obtained, in which the coordinate information came from $View_i$ and the color information came from $Frame_i$. Second, according to the coordinate range of the point cloud, rectangular areas were drawn, and the grids were divided according to the interval D. Third, the color point cloud Ptsci was mapped to the corresponding grid, and finally an orthophoto image was generated. The orthophoto images Oimg1 and Oimg2 generated by Camera1 and Camera2 are shown in Figure 13a,b. In order to overlay the image with the RS or map, a text file in JGW format was added to achieve the overlay effect, as shown in Figure 13c,d.

4.2.4. Comparative Analysis

Traditional orthographic video generation methods mainly include homography and camera parameter methods. Since homography is mainly used for flat areas and is mostly used for bullet cameras, it cannot meet the dynamic geographic mapping requirements of PTZ cameras. This article focuses on two aspects, visualization effect and control point deviation, and compares and analyzes the method proposed in this article with the traditional camera parameter method.

(1) Comparative analysis of visualization effects

For Camera1 and Camera2, the camera parameter method (CPM) and the method in this paper (GVM) were used to generate orthophoto videos, as shown in Figure 14. The red lines in the orthophoto video image represent real line features, such as rivers and roads, which were drawn from high-definition RS and have accurate positions. As can be seen

from Figure 14a,b, there is a large deviation between the red lines and the orthographic video generated by the CPM method. However, the red lines are completely consistent with the rivers and roads in the GVM method and have a good visualization effect, as shown in Figure 14c,d. Therefore, compared with the CPM method, the GVM method is more consistent with the characteristic lines such as roads and rivers in high-definition RS and has a better fusion effect with base maps such as RS and maps.

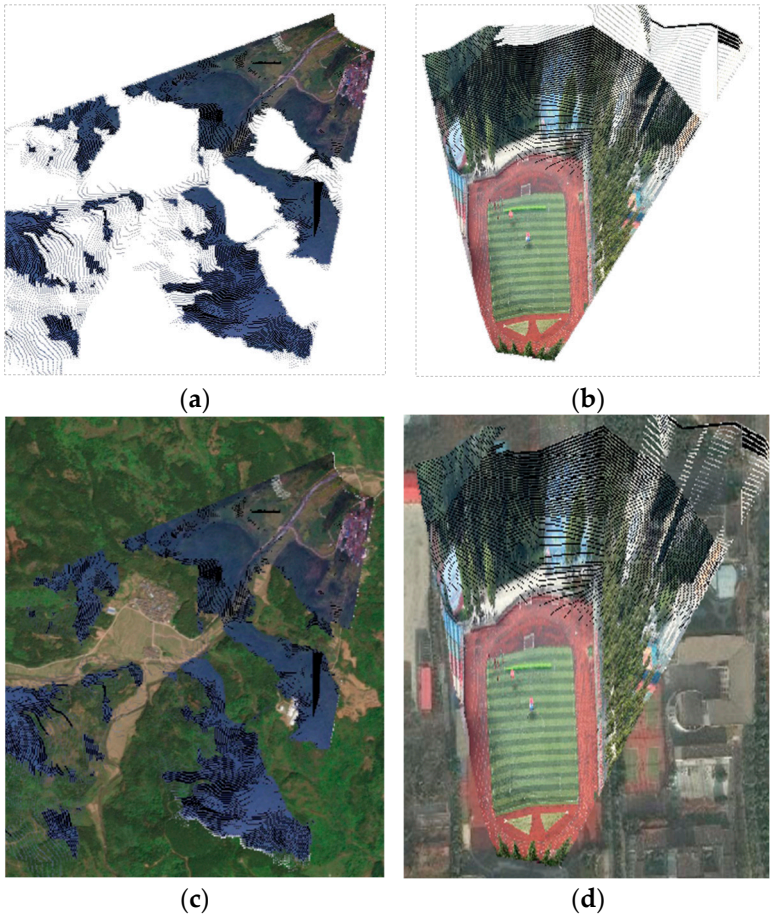


Figure 13. Orthophoto generation and its overlay effect with RS: (**a**) Camera1 orthophoto; (**b**) Camera2 orthophoto; (**c**) overlaying effect of Oimg1 and RS; (**d**) overlaying effect of Oimg2 and RS.

(2) Comparative analysis of control point deviations

In order to quantitatively analyze the mapping accuracy of the two methods, 12 control points with the same name were selected from the high-definition RS. Orthophotos were generated by the CPM and GVM methods, and their errors of geographic mapping were analyzed, as shown in Figure 15. The first column in Figure 15a represents the RS images of the two experimental areas, and 12 control points were selected, numbered from 1 to 12. These control points are feature points distributed on roads, rivers, etc. The second column in Figure 15b is the orthophoto generated by the CPM method, showing the positions of the 12 control points. The third column in Figure 15c is the orthophoto generated by the GVM method, showing the positions of the 12 control points.

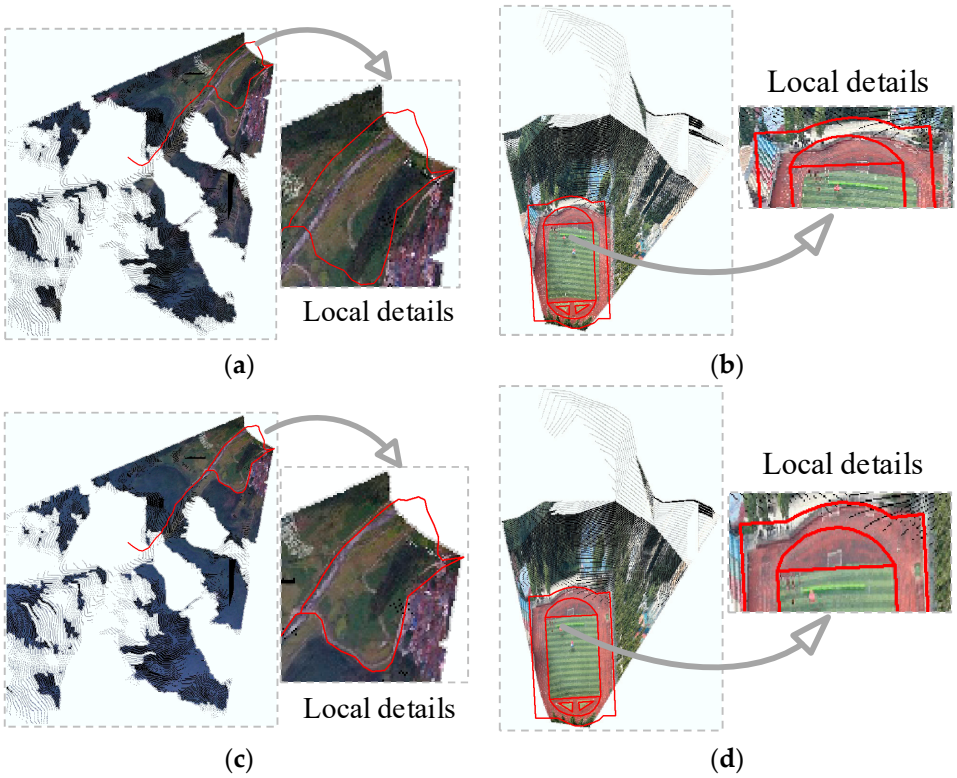


Figure 14. Comparative analysis of visualization effects in Figure 14: (**a**) experimental effect of CPM method in Camera1; (**b**) experimental effect of CPM method in Camera2; (**c**) experimental effect of GVM method in Camera1; (**d**) experimental effect of GVM method in Camera1.

In order to analyze the accuracy of orthophotos generated by CPM and GVM methods, this paper analyzes the offset of 12 control points. The offset of each point was calculated as the distance between the coordinates in the high-definition RS and the corresponding coordinates in the orthophoto video. Assuming that the coordinates of control point 1 in the high-definition RS are (X1, Y1) and the corresponding coordinates in the CPM image are (Xc1, Yc1), the offset calculation is as shown in Formula (11). The calculation of the control point offsets for the GVM method is performed in the same way.

$$\mathrm{dis} = \sqrt{((X_1 - Xc_1)^2 + (Y_1 - Yc_1)^2)} \tag{11}$$

Figure 15 shows the offsets of the 12 control points in Camera1 and Camera2. Overall, the offsets of the control points in Camera1 and Camera2 are smaller for the GVM method and larger for the CPM method. In Camera1, the maximum offset of the 12 control points of the CPM method is 321.78 m, and the minimum offset is 16.15 m, while the maximum offset of the GVM method is 23.11 m, and the minimum offset is 1.78 m. In Camera2, the maximum offset of the 12 control points of the CPM method is 28.07 m, and the minimum offset is 5.38 m, while the maximum offset of the GVM method is 20.32 m, and the minimum offset is 0 m. At the same time, RMSE shows that in the Camera1 area, the CPM method is 137.70 m, and the GVM method is 7.72 m. In the Camera2 area, the CPM method is 13.52 m, and the GVM method is 8.10 m. It can be concluded that the accuracy of the GVM method is higher than that of the traditional CPM method.

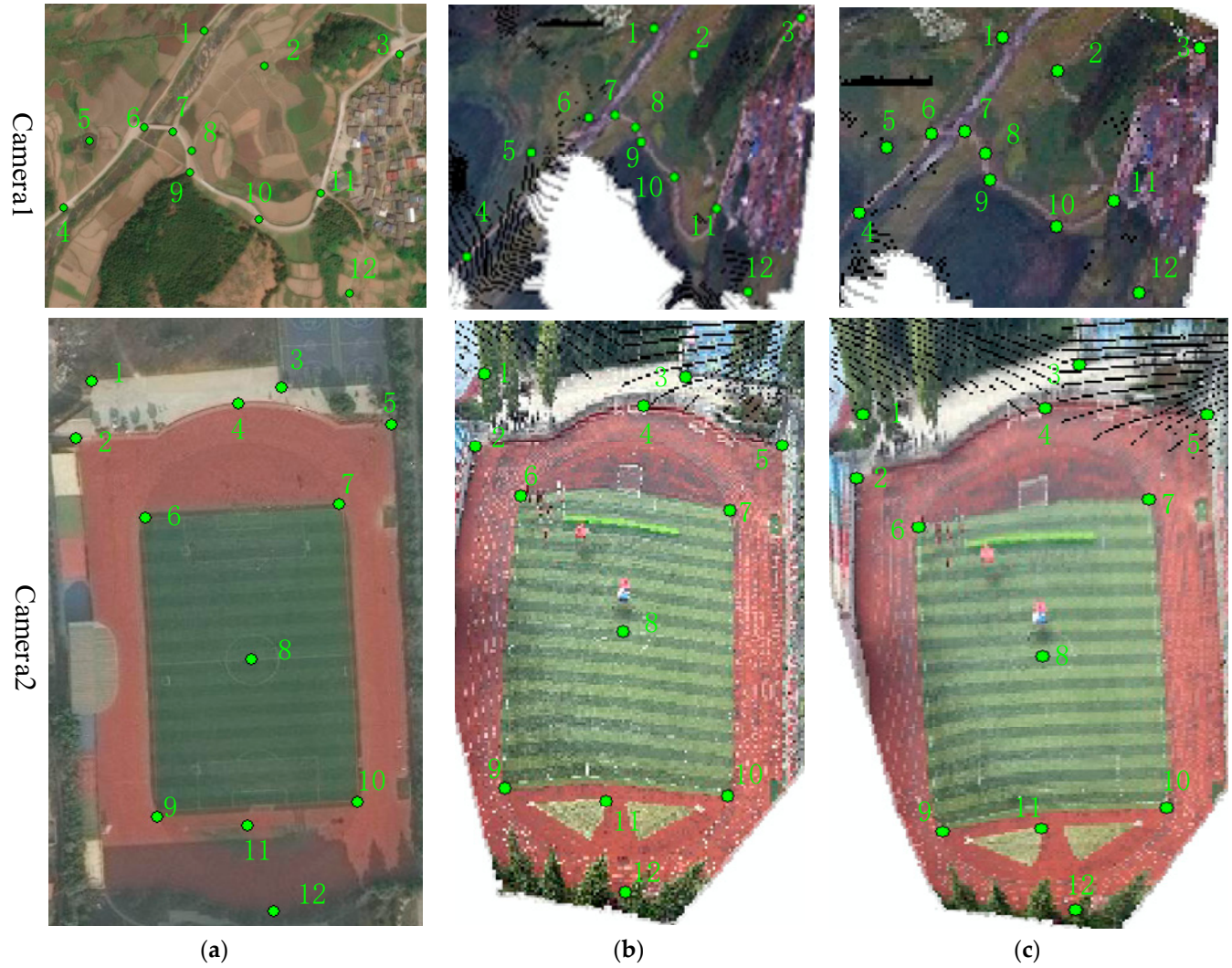


Figure 15. Control point distribution in Camera1 and Camera2: (**a**) high-resolution RS images; (**b**) CPM; (**c**) GVM. The green dots in the figure represent the 12 control points selected in the corresponding view, and the numbers next to them are the numbers of the control points.

In the GVM method, referring to Figure 11, the control points within the feature point coverage area have a small offset, while the control points at or outside the edge of the coverage area exhibit larger offsets. For example, in Camera1, all 12 control points are within the feature point coverage area, and the overall error is small. Control point 3 is located at the edge of the feature point coverage area, and the offset is large. In Camera2, the deviations of control points 1 and 2 are large relative to other points, as Figure 16.

4.2.5. Generation of Orthographic Video Map Based on Real-Scene 3D Model

To verify the correctness and accuracy of this method, we designed a new experimental scenario. Using a drone and a five-lens camera, we captured numerous photos of the scene and constructed a real-scene 3D model of the area. Figure 17a shows a specific viewing angle. Simultaneously, we captured a surveillance video (Camera3) at a specific location, as shown in Figure 17b.

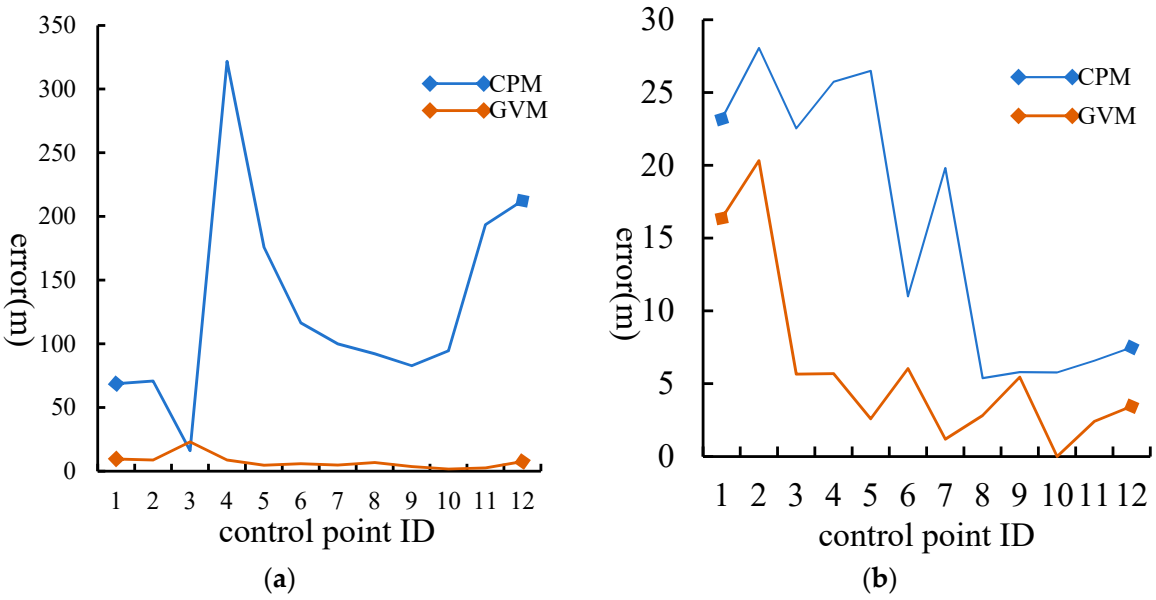
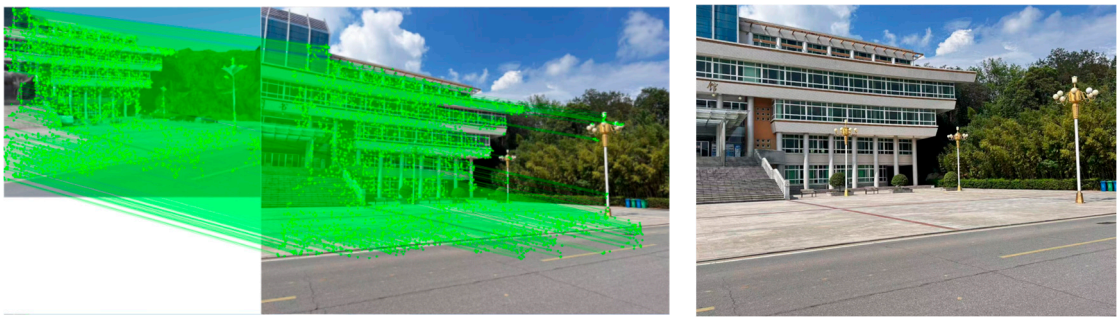


Figure 16. Error analysis of control points in Camera1 and Camera2: (**a**) Camera1 control points error; (**b**) Camera2 control points error.

Figure 17. Realistic 3D model view and video frame of the location where Camera3 is located: (**a**) realistic 3D model view; (**b**) video frame.

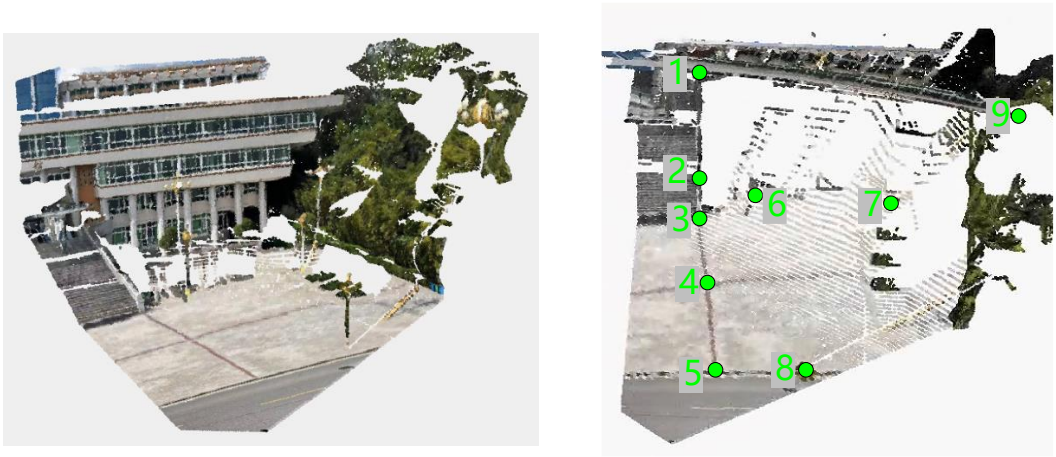
Experiments were conducted on the collected real-scene 3D model and surveillance video frame using the method described in Section 3. Among them, the results of the feature point detection and matching show that for buildings, most of the ground, and street lights, there are more feature points, and there are fewer feature points near the camera, as shown in Figure 18a. The surveillance video frame was then transformed using homography, as shown in Figure 18b.

Based on the homography transformed video frame and the corresponding perspective of the real-scene 3D model, the generated color point cloud is shown in Figure 19a, and the final orthorectified image is shown in Figure 19b. Among them, nine points were set in Figure 19b for deviation analysis.



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

Figure 18. Schematic diagram of feature point detection and matching results for Camera3: (**a**) schematic diagram of feature point detection and matching for real-scene 3D model and surveillance video frame; (**b**) the result of homography transformation after video frame.



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

Figure 19. Schematic diagram of Camera3 orthophoto: (**a**) 3D model of point cloud; (**b**) 9 control points selected.

Through control point deviation analysis, as shown in Table 3, the maximum offset is 4.79 m, and the minimum offset is 0.17 m. The control points with larger offsets are point 4 and point 5, which are 3.46 m and 4.79 m, respectively. The RMSE reached 1.61 m. Compared with the online remote sensing map of Camera1 and Camera2, using drone images will greatly improve the accuracy of the orthorectified video map.

Table 3. The control points deviation of Camera3.

ID	1	2	3	4	5	6	7	8	9
dis (m)	0.30	0.17	0.18	3.46	4.79	0.23	0.64	0.28	0.49

5. Conclusions and Discussion

To solve the problem of generating orthographic video maps from surveillance videos, traditional methods mostly use homography and camera model methods. The former is suitable for fixed cameras, while the latter has high requirements in terms of the accuracy of camera parameters. Both methods have difficulty meeting the requirements of large-scale tower-based video orthophotoization. Taking 3D GIS as the starting point, this paper proposes an orthophoto video map generation method that takes into account the matching of video and 3D GIS views. Based on two tower-based cameras in mountainous

areas and flat areas, this paper combines 3D GIS to carry out method design, algorithm implementation, experiments, and analysis. The contributions of this study are as follows: (1) Based on 3D GIS, a method for synchronizing video and 3D GIS views is proposed. Based on the 3D coordinates of the camera and the image principal point, the azimuth and inclination of the camera are solved. If it is a structured scene, the focal length of the camera can be obtained by the vanishing point method. Otherwise, for unstructured scenes, the focal length of the current camera can be calculated according to its field of view by dynamically adjusting the 3D GIS view. (2) Based on view synchronization, a view matching method based on feature point matching is proposed. The two views are the current video frame and the synchronized 3D GIS view. The two views have similar viewing angles, but the difference in the imaging time is large, which brings great difficulties to view matching. This paper introduces the feature point matching algorithm GIM based on deep learning, which has good results. (3) Taking into account the semantic information and the homography matrix of the two views, a video orthophoto method is proposed. Based on the homography matrix, the video image coordinates are converted into 3D coordinates in geographic space, and a color point cloud is generated. The grid is divided in the map, and the point cloud set is placed in the corresponding grid unit, and finally an orthophoto video map is generated.

This method can provide near-real-time orthorectified video maps, which can be overlaid and displayed with existing remote sensing images and maps, expanding the existing forms of geographic information and improving the timeliness of spatiotemporal information. However, the existing methods still have some shortcomings: (1) This method relies on three-dimensional data, mainly including DEM, three-dimensional models, three-dimensional real-scene models, etc. The higher the model accuracy, the better the orthophoto video map effect will be. For example, in the experiment of this paper, only DEM data were used, lacking three-dimensional models of trees, buildings, and other objects, resulting in a poor orthophoto effect and serious deformation. (2) Because the video is a perspective view, there is an occlusion problem in areas with undulating terrain, resulting in holes in orthophoto. Future research can reduce blind spots and improve the effect of orthophoto by collaborative monitoring with multiple cameras.

With the growing amount of surveillance videos and 3D geographic spatial data, relevant research will be carried out on the integration of multiple cameras and 3D models, real-time 3D maps, and the simulation and prediction of geographic scenes to serve the fields of smart cities, natural resources monitoring, emergency rescue, etc.

Author Contributions: Xingguo Zhang conceived and designed the method and coordinated the implementation; Xiangfei Meng and Li Zhang supervised and coordinated the research activities; Xianguo Ling and Sen Yang helped to support the experimental datasets and analyzed the data. All authors participated in the editing of the paper. All authors have read and agreed to the published version of the manuscript.

Funding: The work described in this paper was supported by the National Natural Science Foundation of China (NSFC) (NO. 41401436), the Innovation and Development Special Project of China Meteorological Administration (CXFZ2025Q009), and the Nanhu Scholars Program for Young Scholars of XYNU, the Key Scientific and Technological Research Project of Henan Province (NO. 222102210320).

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Acknowledgments: The author appreciates the editors and reviewers for their comments, suggestions, and valuable time and efforts in reviewing this manuscript.

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

References

1. Li, D.; Xu, X.; Shao, Z. On geospatial information science in the era of loE. *Acta Geod. Cartogr. Sin.* **2022**, *51*, 1–8. [CrossRef]
2. Milosavljević, A.; Dimitrijević, A.; Rančić, D. GIS-augmented video surveillance. *Int. J. Geogr. Inf. Sci.* **2010**, *24*, 1415–1433. [CrossRef]
3. Sourimant, G.; Morin, L.; Bouatouch, K. GPS, GIS and video registration for building reconstruction. In Proceedings of the 2007 IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007; IEEE: Piscataway, NJ, USA, 2007; Volume 6, pp. VI-401–VI-404. [CrossRef]
4. Chen, H.; Qi, Z.; Han, X.; Feng, N.; Li, C. Research and application on key technologies of natural resources intelligent monitoring with tower-based video. *Nat. Resour. Informatiz.* **2023**, *2023*, 1–6. [CrossRef]
5. Sankaranarayanan, K.; Davis, J.W. A fast linear registration framework for multi-camera GIS coordination. In Proceedings of the 2008 IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, Santa Fe, NM, USA, 1–3 September 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 245–251. [CrossRef]
6. Milosavljević, A.; Rančić, D.; Dimitrijević, A.; Predić, B.; Mihajlović, V. Integration of GIS and video surveillance. *Int. J. Geogr. Inf. Sci.* **2016**, *30*, 2089–2107. [CrossRef]
7. Xie, Y.; Wang, M.; Liu, X.; Wang, Z.; Mao, B.; Wang, F.; Wang, X. Spatiotemporal retrieval of dynamic video object trajectories in geographical scenes. *Trans. GIS* **2021**, *25*, 450–467. [CrossRef]
8. Luo, X.; Wang, Y.; Dong, J.; Li, Z.; Yang, Y.; Tang, K.; Huang, T. Complete trajectory extraction for moving targets in traffic scenes that considers multi-level semantic features. *Int. J. Geogr. Inf. Sci.* **2023**, *37*, 913–937. [CrossRef]
9. Zhang, X.; Shi, X.; Luo, X.; Sun, Y.; Zhou, Y. Real-time web map construction based on multiple cameras and GIS. *ISPRS Int. J. Geo Inf.* **2021**, *10*, 803. [CrossRef]
10. Shao, Z.; Li, C.; Li, D.; Altan, O.; Zhang, L.; Ding, L. An accurate matching method for projecting vector data into surveillance video to monitor and protect cultivated land. *ISPRS Int. J. Geo Inf.* **2020**, *9*, 448. [CrossRef]
11. Zhang, X.; Liu, X.; Wang, S.; Liu, Y. Mutual Mapping Between Surveillance Video and 2D Geospatial Data. *Geomat. Inf. Sci. Wuhan Univ.* **2015**, *40*, 1130–1136. [CrossRef]
12. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *22*, 1330–1334. [CrossRef]
13. Tsai, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robot. Autom.* **2003**, *3*, 323–344. [CrossRef]
14. Hartley, R.I. Self-calibration of stationary cameras. *Int. J. Comput. Vis.* **1997**, *22*, 5–23. [CrossRef]
15. Yang, C.; Wang, W.; Hu, Z. An active vision based camera intrinsic parameters self-calibration technique. *Chin. J. Comput.* **1998**, *21*, 428–435. [CrossRef]
16. Hou, W.; Shang, T.; Ding, M. Self-Calibration of a Camera with a Non-Linear Model. *Chin. J. Comput.* **2002**, *25*, 276–283. [CrossRef]
17. Hartley, R.I. Projective reconstruction and invariants from multiple images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1994**, *16*, 1036–1041. [CrossRef]
18. Pollefeys, M.; Van Gool, L.; Oosterlinck, A. The modulus constraint: A new constraint self-calibration. In Proceedings of the 13th International Conference on Pattern Recognition, Vienna, Austria, 25–29 August 1996; IEEE: Piscataway, NJ, USA, 1996; Volume 1, pp. 349–353. [CrossRef]
19. Agapito, L.; Hayman, E.; Reid, I. Self-calibration of rotating and zooming cameras. *Int. J. Comput. Vis.* **2001**, *45*, 107–127. [CrossRef]
20. Hemayed, E.E. A survey of camera self-calibration. In Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, Miami, FL, USA, 22 July 2003; IEEE: Piscataway, NJ, USA, 2003; pp. 351–357. [CrossRef]
21. Triggs, B. Autocalibration and the absolute quadric. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; IEEE: Piscataway, NJ, USA, 1997; pp. 609–614. [CrossRef]
22. Heyden, A.; Astrom, K. Flexible calibration: Minimal cases for auto-calibration. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; IEEE: Piscataway, NJ, USA, 1999; Volume 1, pp. 350–355. [CrossRef]
23. Hartley, R.I.; Hayman, E.; de Agapito, L.; Reid, I. Camera calibration and the search for infinity. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; IEEE: Piscataway, NJ, USA, 1999; Volume 1, pp. 510–517. [CrossRef]
24. Ge, D.; Yao, X.; Hu, C.; Lian, Z. Nonlinear camera model calibrated by neural network and adaptive genetic-annealing algorithm. *J. Intell. Fuzzy Syst.* **2014**, *27*, 2243–2255. [CrossRef]
25. Woo, D.M.; Park, D.C. An efficient method for camera calibration using multilayer perceptron type neural network. In Proceedings of the 2009 International Conference on Future Computer and Communication, Kuala Lumpar, Malaysia, 3–5 April 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 358–362. [CrossRef]
26. Woo, D.M.; Park, D.C. Implicit camera calibration using MultiLayer perceptron type neural network. In Proceedings of the 2009 First Asian Conference on Intelligent Information and Database Systems, Dong hoi, Vietnam, 1–3 April 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 313–317. [CrossRef]

27. Duan, Q.; Wang, Z.; Huang, J.; Xing, C.; Li, Z.; Qi, M.; Gao, J.; Ai, S. A deep-learning based high-accuracy camera calibration method for large-scale scene. *Precis. Eng.* **2024**, *88*, 464–474. [CrossRef]
28. Li, B.; Wang, X.; Gao, Q.; Song, Z.; Zou, C.; Liu, S. A 3D scene information enhancement method applied in augmented reality. *Electronics* **2022**, *11*, 4123. [CrossRef]
29. Li, C.; Liu, Z.; Zhao, Z.; Dai, Z. A fast fusion object determination method for multi-path video and three-dimensional GIS scene. *Acta Geod. Cartogr. Sin.* **2020**, *49*, 632. [CrossRef]
30. Wang, Y.; Yuan, Y.; Lei, Z.; Wang, Y.; Yuan, Y.; Lei, Z. Fast SIFT feature matching algorithm based on geometric transformation. *IEEE Access* **2020**, *8*, 88133–88140. [CrossRef]
31. Qi, F.; Weihong, X.; Qiang, L. Research of image matching based on improved SURF algorithm. *TELKOMNIKA Indones. J. Electr. Eng.* **2014**, *12*, 1395–1402. [CrossRef]
32. Li, S.; Wang, Q.; Li, J. Improved ORB matching algorithm based on adaptive threshold. *J. Phys. Conf. Ser.* **2021**, *1871*, 012151. [CrossRef]
33. Zhang, J.; Zhou, H.; Niu, Y.; Lv, J.; Chen, J.; Cheng, Y. CNN and multi-feature extraction based denoising of CT images. *Biomed. Signal Process. Control* **2021**, *67*, 102545. [CrossRef]
34. Hong, Y.; Li, D.; Luo, S.; Chen, X.; Yang, Y.; Wang, M. An improved end-to-end multi-target tracking method based on transformer self-attention. *Remote Sens.* **2022**, *14*, 6354. [CrossRef]
35. Fujimoto, S.; Matsunaga, N. Deep feature-based RGB-D odometry using SuperPoint and SuperGlue. *Procedia Comput. Sci.* **2023**, *227*, 1127–1134. [CrossRef]
36. Shen, X.; Cai, Z.; Yin, W.; Müller, M.; Li, Z.; Wang, K.; Chen, X.Z.; Wang, C. GIM: Learning generalizable image matcher from internet videos. *arXiv* **2024**, arXiv:2402.11095. [CrossRef]
37. Vishnu, C.; Khandelwal, J.; Mohan, C.K.; Reddy, C.L. EVAA—Exchange vanishing adversarial attack on LiDAR point clouds in autonomous vehicles. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5703410. [CrossRef]
38. Chen, L.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.