

Research Article

An Indoor Location-Based Positioning System Using Stereo Vision with the Drone Camera

Young-Hoon Jin , Kwang-Woo Ko, and Won-Hyung Lee 

Culture Technology Research Institute, Chung-Ang University, 221 Heucsuk-dong, Dongjak-ku, Seoul, Republic of Korea

Correspondence should be addressed to Won-Hyung Lee; whlee@cau.ac.kr

Received 21 May 2018; Revised 7 August 2018; Accepted 29 August 2018; Published 17 October 2018

Academic Editor: Jaegaeol Yim

Copyright © 2018 Young-Hoon Jin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Stereo vision is used to reconstruct 3D information of the space by estimating the depth value from the simulation of human eyes. Spatial restoration can be used as a means of location estimation in an indoor area, which is impossible to accomplish using the relative location estimation technology, GPS. By mapping the real world in virtual space, it is feasible to clear the boundary between real space and virtual space. This paper presents a method to control the drone indoors through a positioning system using Structure from Motion algorithm (SfM). SfM calculates the relative relationship between cameras based on images to be acquired from various locations and obtains disparity to enable restoration of 3D space. First, the 3D virtual space is reconstructed using several photographs taken from an indoor environment. Second, the real-time drone position is determined by comparing the 3D virtual space camera with the image displayed on the drone camera. In this case, if the direction of the virtual camera used for 3D virtual space construction is the same as the amount of yaw rotation of the drone, it is possible to quickly find the same position as the image seen in the real drone camera in the virtual space. As a result, if the scale of the actual camera image and the virtual camera image is 1 : 1 matched, then it is possible to know that the drone is in the position of the virtual camera. The proposed indoor location-based drone controlling method can be applied to various drone applications such as group flight in an indoor environment because of its ability to fly the drone without the use of the traditional remote-control and flight trajectory programming.

1. Introduction

In modern society, drones are aviation system with a wide range of applications. The drone developed in the military is efficient equipment that can perform tasks at a low cost and without any risk in industry, agriculture, and disaster prevention. Drones can generally be controlled using equipment such as GPS, cameras, laser scanners, and ultrasonic sensors. For example, GPS can measure position and altitude, cameras can acquire images that are difficult to see with the naked eyes, and laser sensor can measure the distance between objects. The measured data can be used for autonomous flying or object recognition.

The drone can acquire various information while measuring the position as well as altitude, but it is relatively limited in the indoor environment. GPS works only outdoors and a good laser scanner is big and heavy for flying small indoor drones. On the contrary, big drones that could

carry laser scanners are dangerous and difficult to control in the indoor environment. Even though using a light-weight ultrasonic sensor can measure the distance to some extent, the measurement is inaccurate. In order to address this problem, research in the field of computer vision is incorporated in drone technology.

Computer vision can be used for position control of the drone. Distance measurement through a camera generally uses two cameras, which is similar to the principle of the human eyes. Since the two eyes of a human observe an object at different positions, there is a difference between the vantage point angles, which is called disparity, and this disparity can be used to estimate the distance of the observed object from the ‘eye’ or the camera. Stereo vision is based on this principle.

After measuring the distance, it is possible to determine the relative position of the drone through the camera without using GPS or laser sensors. In order to determine the relative position of the drone, it is necessary to determine the

relative distance from the origin in a 3D virtual environment. The 3D reconstruction of the indoor space can be constructed using Structure from Motion (SfM). SfM reconstructs 3D coordinates from 2D images with disparity, which is the characteristic of stereo vision using multiple images. Multiple images for SfM can be easily acquired with a drone.

As shown in Figure 1, the SfM pipeline finds the feature points in each image and finds the feature points corresponding to the points of interest in one image into the other image, and this process is called correspondence. Various studies have been carried out to find corresponding points; Lowe proposed an algorithm to extract features that are invariant to image size and rotation [1]. The algorithm constructs a scale space and finds the key points through the DoG (difference of Gaussian) operation, then removes any keys that do not meet the criteria, and assigns the directionality to the appropriate key. By configuring the descriptors by assigning a fingerprint to these keys, a unique matching point with scale and rotation invariance is constructed. These matching points have high accuracy but require a large computational complexity and long execution time.

Bay proposed a SURF (Speed-Up Robust Features) algorithm that resolves these drawbacks [2]. To use the integral image, the fast Hessian detector was used. If the determinant is a positive number and the sign of the given value is the same, it is assigned as a key point. Similar to SIFT (scale invariant feature transform), key points are extracted from various scales. Afterwards, the direction is given through the Haar wavelet response. SURF is relatively fast but uses only gray space information.

Once good feature points are found, the path through which the feature points move in each image can be analyzed. Carlo Tomasi proposed a method of tracking strong feature points in every frame through optical flow [3]. It is shown that the optical flow in the region N of the window centered on the pixel is the same and can be traced. Since the above method is a local algorithm, the size of the window is important. It also has a disadvantage because optical flow is affected by the instantaneous changing light.

On the contrary, there is a guided searching method approaching through epipolar geometry. The feature points of an image are mapped to other images by the matrix F and near the epipolar line. In this case, F is called the fundamental matrix, and the rotation and translation $[R|t]$ vectors of the camera can be constructed from F [4–11]. This allows us to identify camera movement.

Wang constructed an indoor space using VisualSfM by obtaining images of multiple angles and the camera trajectory of the acquired images and making “belief maps” through FCNN (fully convolution neural networks) to complete the space [12].

Snavely uses image-based rendering technology to complete 3D model correspondence from various images published on the web and to navigate the virtual space through a smooth transition between images [13].

Ryan et al. conducted an epidemiological investigation at low cost in the western part of the Greenland ice sheet and proved to be efficient in characterizing large jet glaciers [14].

In addition, indoor location-based studies using RSSI (received signal strength indication) or TOF (time of flight) are introduced in this study [15, 16].

Multiview reconstruction is possible, based on the above studies, and spatial reconstruction is utilized efficiently in various areas.

Dense reconstruction is required because the 3D coordinates formed along the camera motion are low in density. In general, dense reconstruction through multiview stereo (MVS) algorithm of Figure 2 has the type of depth map, point cloud, volume scalar field, and mesh [17].

The depth map can be used in various areas such as scene analysis and visualization, but there is a problem in merging the 3D model of the entire area. Likewise, the quality of the 3D model may deteriorate. The point cloud is easy to merge and split, and it can overcome the drawbacks of depth map because it creates a single point cloud from all input images.

The volume scalar field method can be restored from images, depth maps, and point clouds, but integrating into a single mesh is a difficult problem.

This paper aims at 3D reconstruction through an indoor location-based drone control method using a single-view camera mounted on a drone. Various researches in computer vision are examined in the development of this study. In the proposed method, the camera image obtained from the drone is transmitted to the ground station and the spatial coordinates reconstructed by SfM are visualized by the 3D modelling program. If the position of the drone is estimated by comparing the image projected on the virtual camera with the current image of the drone camera, it is possible to control the drone using the virtual spatial coordinates. If the yaw rotation of the drone and the rotation of the virtual camera are equal, the position of the drone can be estimated with high accuracy and it is proven with experimental result (Section 4) in the proposed method.

The rest of this paper describes the proposed method. First, the feature point estimation in each image is explained in Sections 2.1 and 2.2 discusses the epipolar geometry configuration using correspondence. Sections 2.3 and 2.4 describe the 3D reconstruction of correspondence. In Sections 3.1 and 3.2, an environment map is presented and the position estimation of the drone is discussed, respectively.

The experimental verification and discussion are presented in Section 4, and finally, Section 5 draws the conclusion on the proposed method.

2. 3D Reconstruction at Indoor Positioning System

The pipeline for SfM is shown in Figure 1. SIFT extracts the feature from the input image sequence and correspondence is calculated from each image based on the extracted feature. By calculating the fundamental matrix using features with high accuracy, the essential matrix through the camera matrix can be determined. By decomposing the essential matrix into singular value decomposition (SVD), the rotation/translation matrix is calculated. Once calculating $[R|t]$, the movement path of the camera can be known and 3D coordinates can be obtained through triangulation.

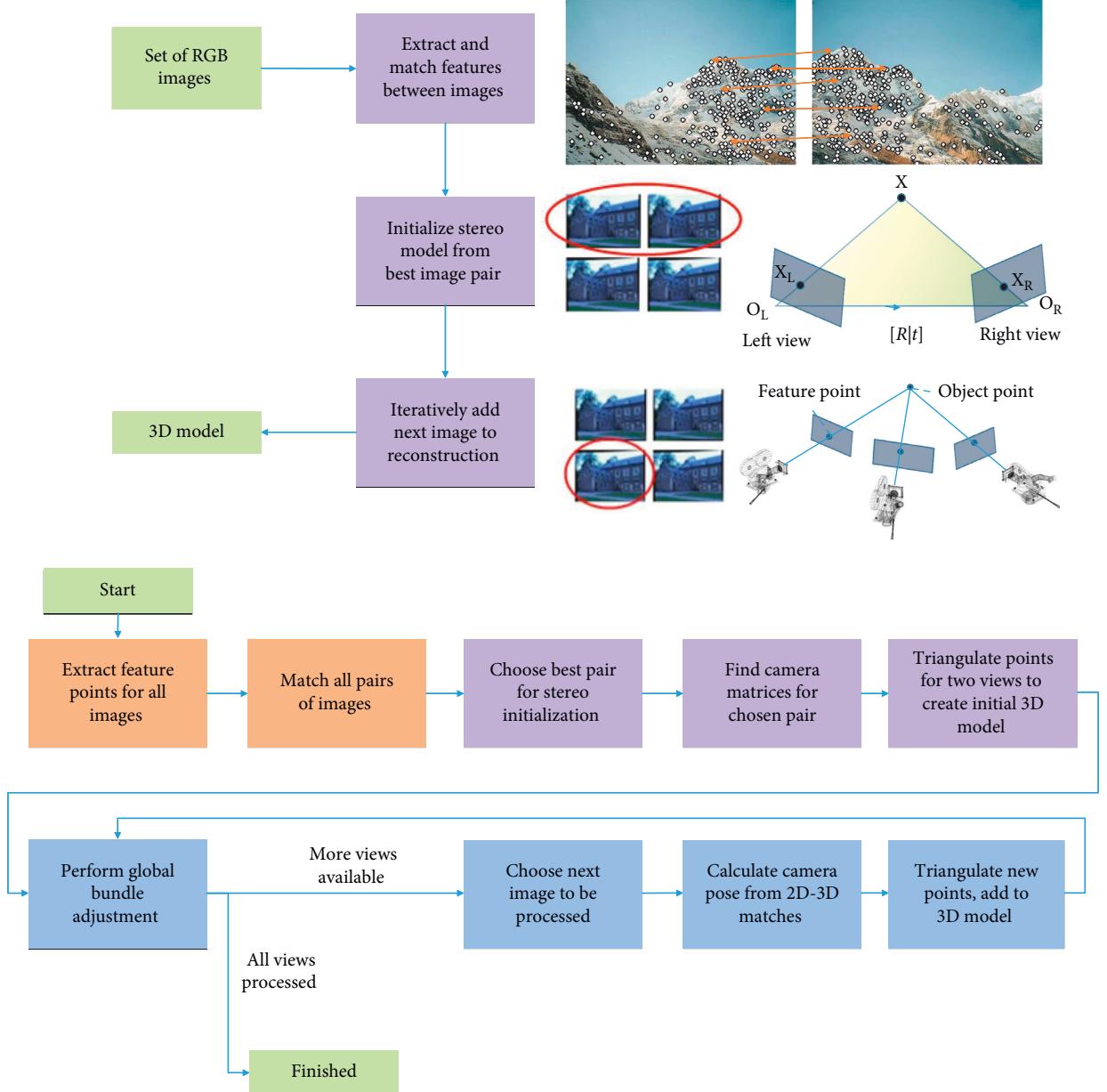


FIGURE 1: Incremental multiview reconstruction [9] and pipeline.

Repeatedly applying this to all image sequences will enable 3D reconstruction.

2.1. Feature Detection and Matching. SIFT is an algorithm that extracts feature points that are robust to scale and rotate. Figure 3 depicts SIFT algorithm procedure.

Scale-space extrema detection generates Gaussian pyramid and calculates DoG to extract strong feature point candidates for scale change. The key point localization step extracts the correct feature points from the candidate group through the Taylor series. Extracted feature points are assigned directionality in the orientation assignment step. The orientation histogram is formed by Gaussian blurring, and then the orientation is estimated.

Finally, the directionality is assigned to a certain area and the descriptor is completed as shown in Figure 4.

The matching method of feature points in the two images finds the same region through comparison of calculated descriptors. In this case, the easiest comparison method is to find the same pair by comparing the distances of two sets as shown in Equation (1) by pairwise matching. Distance matching has advantages of being relatively faster because of being simple to calculate and easy to implement.

Euclidean distance matching:

$$D = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \quad A = (a_1, a_2, a_3, \dots, a_n), \quad B = (b_1, b_2, b_3, \dots, b_n). \quad (1)$$

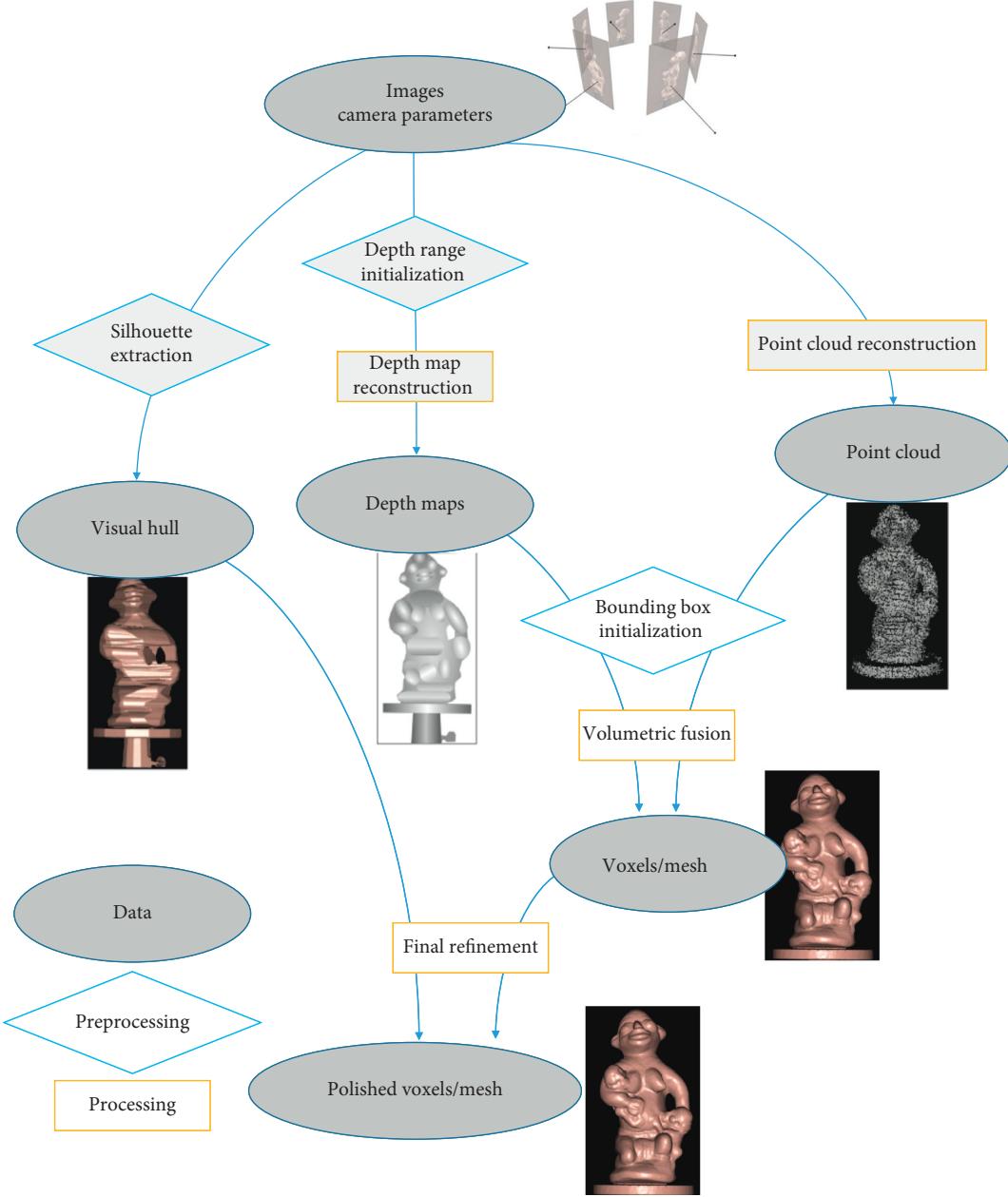


FIGURE 2: MVS processing [17].

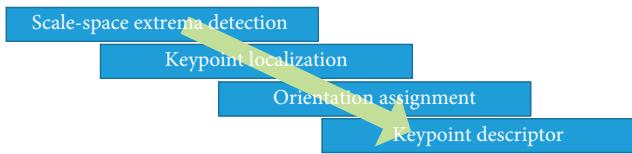


FIGURE 3: SIFT algorithm procedure [1].

After establishing the correspondence of each image, the path of camera movement can be obtained as described in Section 2.2.

2.2. Fundamental and Essential Matrixes. The fundamental matrix is a matrix containing the properties of the camera,

while the essential matrix contains the geometric relationships of the pixel coordinates on the two images. The matrix F in Equation (2) can be calculated when a correspondence is constructed, if there are the eight corresponding points.

Fundamental matrix [4]:

$$p_2^T F p_1 = 0, \quad (2)$$

where F : fundamental matrix. p_1 and p_2 : corresponding points.

The essential matrix implies a geometric relationship in the normalized image plane. This means that the geometric relationship of both cameras is toward a point in the space. In other words, matrix E shows the rotation and movement relationship between the two cameras.

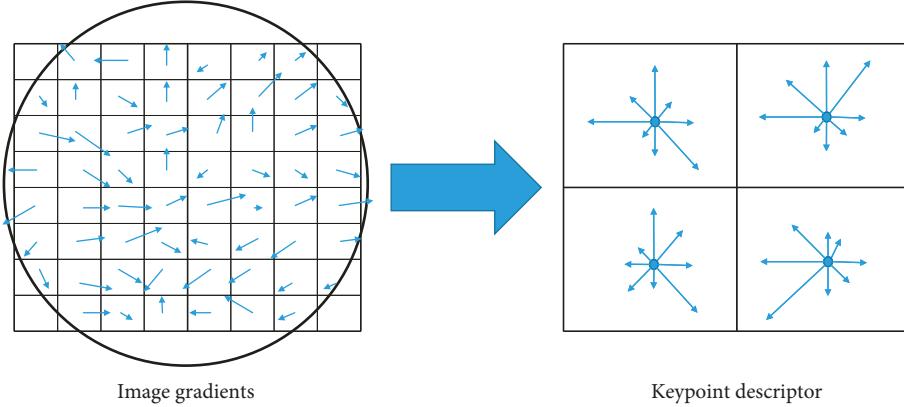


FIGURE 4: Key point descriptor [1].

The matrix E can be computed by $E = K' FK$ or the corresponding five points and the camera matrix K , where K is obtained from the EXIF metadata.

Essential matrix [4]:

$$p_2^T K^T E K p_1 = 0, K = \begin{bmatrix} f & 0 & x_{pp} \\ 0 & f & y_{pp} \\ 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

f : focal length and x_{pp} , y_{pp} : the center of the image.

Matrix E can be decomposed into a rotation matrix R and a movement matrix t by SVD. If the camera matrix of the first image is P , then the camera matrix P' of the next image can be obtained as Equation (4).

SVD decomposition of the essential matrix [4]:

$$\begin{aligned} E &= U_{\text{diag}}(1, 1, 0)v^T, P = [I|0], \\ P' &= [UWV^T|+u_3], \\ &\quad [UWV^T|-u_3], \\ &\quad [UW^TV^T|+u_3], \\ &\quad [UW^TV^T|-u_3]. \end{aligned} \quad (4)$$

There are four cases of P' , and the solution in which the feature points exist in front of the two cameras at a single point can be selected.

2.3. Sparse Reconstruction. The motion of the camera in the 3D space can be estimated by obtaining the rotation and movement matrix $[R|t]$. Figure 5 depicts the configuration of the 3D coordinates through camera movement in spatial coordinates. If the epipolar geometry [4, 17] is constructed from the position of the camera estimated from the initial image in the sequence image, a 3D coordinate can be obtained through triangulation [11]. The calculated 3D coordinates are projected on the added camera, and the error is calculated with the feature points formed in the image. This is called the reprojection error [4, 17, 18], and the reconstruction work is performed to correct the error to the minimum. Expression of the reprojection error Equation (5) is as follows. P_i is the projection matrix of the i -th image, X_j

is the j -th 3D point, and x_{ji} is the observation of the 3D point.

Reprojection error:

$$\text{RE}(P, X) = \sum_{j=1}^m \sum_{i=1}^n D(x_{ji}, P_i X_j)^2. \quad (5)$$

2.4. Dense Reconstruction. The sparse reconstruction method reconstructs the camera position and direction at the moment of acquiring the image through the corresponding point. The dense reconstruction method, on the contrary, performs dense reconstruction based on known camera motion [17, 19, 20]. Figure 6 depicts the method of dense reconstruction, which constructs the epipolar geometry through the camera's direction and movement path in the 3D space and finds the corresponding points in the epipolar line. It can be quickly found as an epipolar constraint. When the corresponding points are found, triangulation is performed centering on the corresponding points to construct dense 3D space coordinates.

The triangulation can be confirmed by the optical ray intersection of the direction of the feature point mapped to the image from the origin of the camera. However, since the exact intersection cannot be confirmed due to the measurement noise, a pixel with a small error is selected. When a set of selected pixels is referred to as a patch, as shown in Figure 6, the patch that occludes other patches or is hidden from multiple patches is judged to be outliers, thereby increasing the accuracy.

3. Indoor Location Estimation of the Drone Camera

The drone flight usually involves acceleration, gyroscope, and geomagnetic sensors. Acceleration/gyro sensors are involved in the horizontal plane of the drone, and geomagnetic sensors are involved in yaw rotation of the drone. In addition, various sensors can be mounted in the drone. The laser tracking or GPS is often used to locate the drone. But laser tracking is not traceable when the drone is obstructed by obstacles, and the GPS does not work indoors

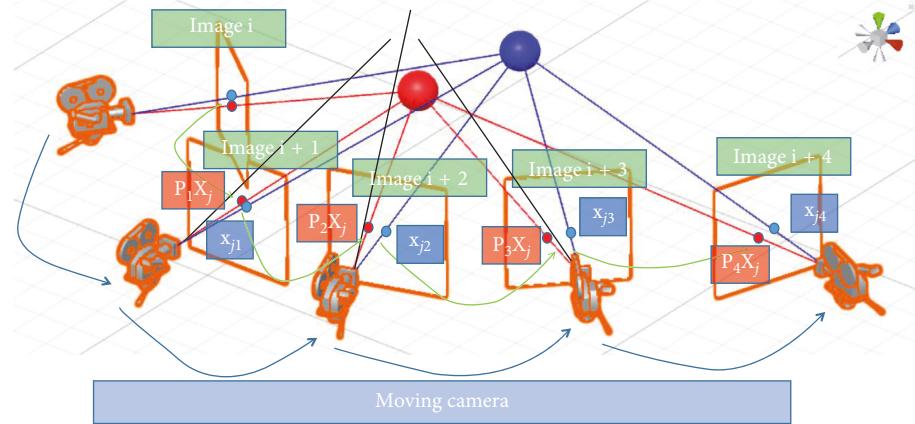


FIGURE 5: Sparse 3D reconstruction with the reprojection error.

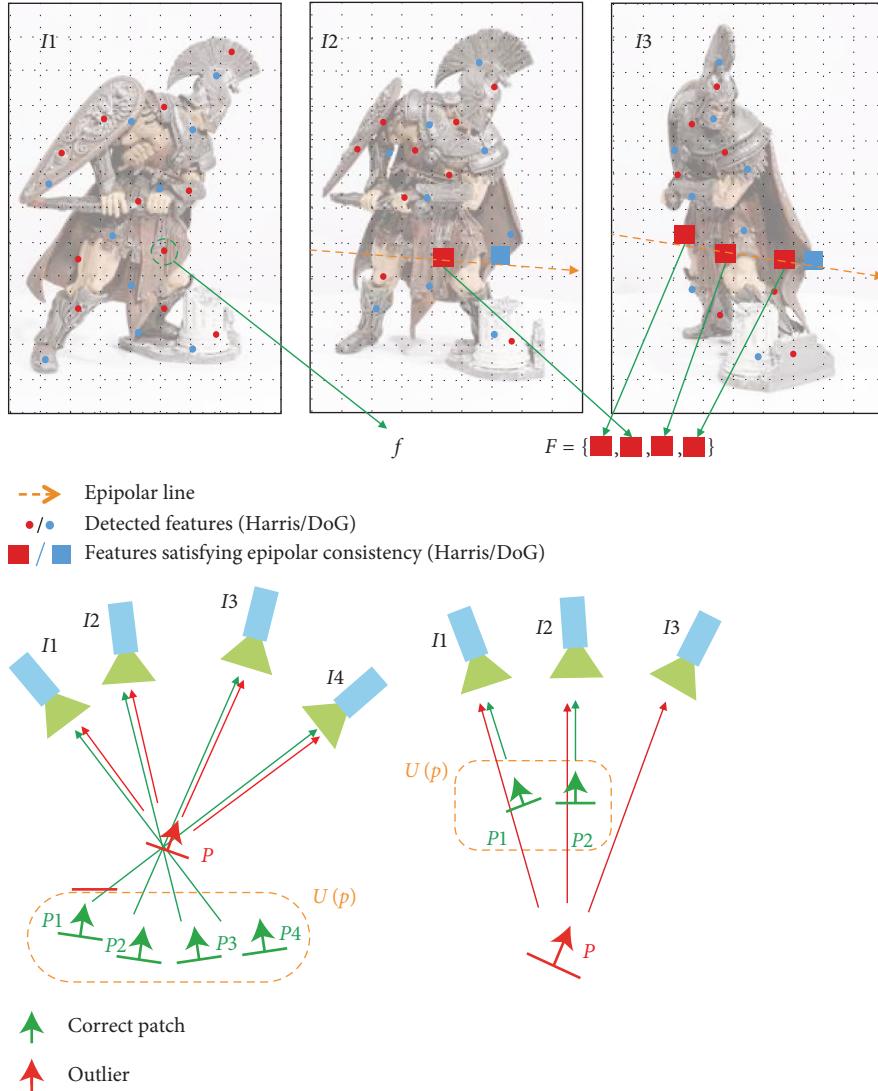


FIGURE 6: Dense reconstruction [19].

because the signal cannot reach. Indoor location estimation can be performed by installing a beacon or an infrared sensor, but accuracy and installation limitations exist. In this paper, the authors propose a method for estimating the indoor position through the computer vision technique.

3.1. Configure Environment Map. In a three-dimensional space, as the position is a relative concept, the distance and direction away from the origin in the space where the origin exists is the position. In other words, in order to locate the drone, an environment map (virtual space) must be constructed first. The origin of the environment map is set as the last position or the average value of the camera positions when acquiring the images for the spatial configuration. The 3D space can be constructed from the sequence images formed by moving along the wall as discussed in Section 2.3. As shown in Figure 7, the spatial configuration is reconstructed through the SfM procedure by obtaining 360° space images through yaw rotation and circular trajectory for a certain time after the takeoff of the drone. The yaw rotation amount of the drone at the time of the last image acquisition is the forward vector.

The partially constructed environment maps are integrated into one coordinate system by registration. Figure 8 illustrates how to make registration with the 12-zone point cloud. The registration can be done through an iterative closest point (ICP) process by constructing key points and looking for correspondence. When two sets of coordinate systems $X = x_1, \dots, x_{N_x}$ and $P = p_1, \dots, p_{N_p}$ exist, the registration as shown in Figure 8 is completed by repeating the calculation of $[R|t]$ where the error of the transformation relation $E(R, t) = (1/N_p) \sum_{i=1}^{N_p} x_i - Rp_i - t^2$ is minimized [22].

Since the configured environment map is in the form of a point cloud, restoration of the mesh shape is necessary. The mesh is generally constructed by removing the noise properly, smoothly treating the rough surface, and then repeatedly bundling the mesh into the appropriate triangular unit.

Mesh reconstruction uses marching cube, Delaunay triangulation, and Poisson surface reconstruction. The Poisson surface reconstruction, which is strong against matching error and noise, is widely used.

Figure 9 depicts the Poisson surface reconstruction. Since the interior of the model has a negative number and the outer one has a positive number, the boundaries have zero, so a directed sample can be interpreted as a gradient discretization. Thus, we can solve this problem by calculating the divergence of successive vector fields in a point cloud and solving Poisson's equation to find the scalar field with the most appropriate gradient and discretizing the octree. In this case, the screening algorithm is used to reduce the time complexity and improve the accuracy of the solver by interpolating and processing the points in the partial domain rather than the whole domain.

3.2. Location Estimation of the Drone. There are two cases of a location estimation of the drone. First, the position of the

camera can be determined by analyzing the sequence image added to the existing sequence, as shown in Section 2.3. As a result, the position of the camera becomes the position of the drone. In this case, the added sequence should be a position where the corresponding point can be found in the existing sequence. The other is a method of comparing an image captured by a virtual camera in a 3D space with an image captured by the current drone camera as depicted in Figure 10. As shown in the figure, when the drone moves toward the target point C, it approaches as the horizontal-vertical flight. This is to increase the accuracy of the image comparison by reducing the change of the image rather than the straight flight toward C.

Let D be the position of the current drone, C be the position of the virtual camera, and P be the plane parallel to the drone. Then the vector M is an orthogonal projection vector of the vector DC to the plane P. At this time, the angle between the forward vector F of the drone and the vector M can be obtained through $\theta = \cos^{-1}(\vec{F} \cdot \vec{M} / \| \vec{F} \| \| \vec{M} \|)$. If the virtual camera also looks toward the M vector, the image similarity can be estimated.

Image similarity can be achieved by the template matching, which is a method to search and check whether a given small template image exists in a large image. Template matching has the disadvantage of only showing good results with the same scale and direction, but it can produce good results because the direction of the drone is the same as the direction of the virtual camera. The similarity of the image can be calculated by Equation (6).

Template matching similarity:

$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2. \quad (6)$$

Figure 11 depicts the template matching. Since the drone camera contains more area than the virtual camera image of the target point, it searches the image of the virtual camera in the image of the drone. At this time, when the image of the virtual camera and the image of the drone become the same size, it can be known that the drone has arrived at the target point.

The location estimation through the template matching can reduce the error if the field-of-view of the virtual camera is the same as the drone camera. Also to overcome the error due to the image scale, we construct an image pyramid for the template image. When the drone reaches the target point, a new image sequence is formed and the procedure of Section 2 is repeated to finalize the position. In this case, if the correspondence of the current sequence cannot be found in the existing sequence, it can be estimated as shown in Figure 12. Then, the image of the drone camera is searched from the images projected on the virtual camera in the same direction as the yaw rotation amount of the current drone. At this time, the image of the drone camera becomes a template, the virtual camera moves in the X-Y plane so that the template area is located at the center of the virtual camera image, and the point at which the image scale becomes 1:1 by the z-axis is determined as the position of the drone.

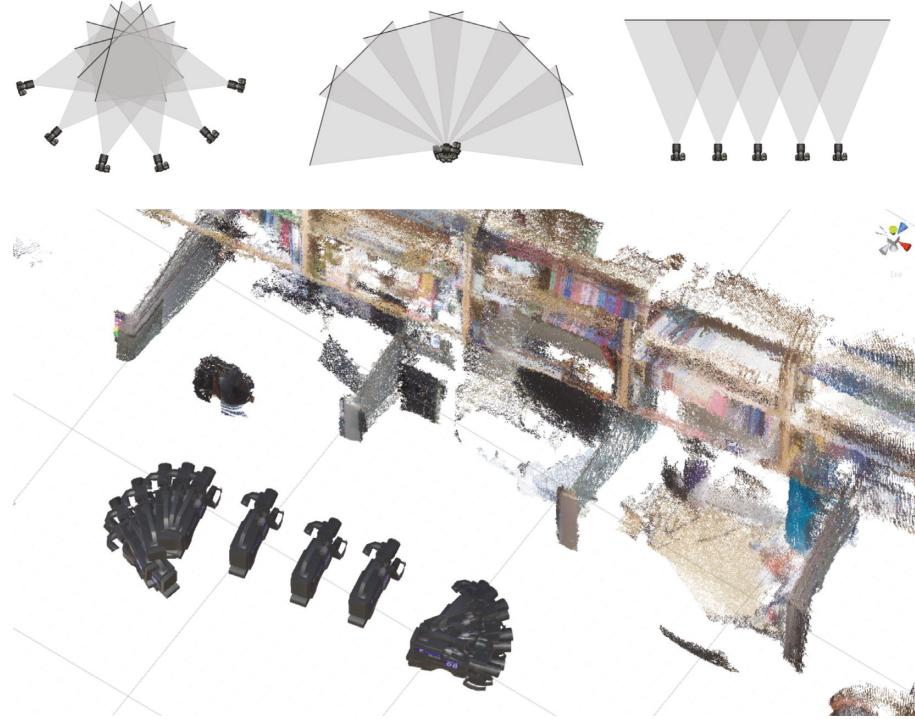


FIGURE 7: Image acquisition method for spatial configuration [21] and result.



FIGURE 8: Registration.

4. Experimental Results and Discussions

Experiments utilize a programmable Arduino drone. Since an Arduino drone is equipped with a low-performance micro-processor, image processing is not possible, so the station is constructed on the ground. The system of the station role consists of Intel 4405U 2.1 GHz CPU, 8 GB ram, uses Unity3D, VisualSFM [25], CloudCompare, and MeshLab for 3D visualization. Figure 13 depicts the entire system. As shown in Figure 13, the drone and the station on the ground exist on the same network and communicate data with each other.

After the drone takes off, it keeps a constant height and scans the space using the image acquisition method of

Figure 7. Precise spatial restoration requires large amounts of photographs with good disparity. The acquired images are sequentially transmitted to the station as shown in Figure 13.

As shown in Figure 14, VisualSFM performs sparse reconstruction from the transmitted image and completes dense reconstruction through clustering views for multiview stereo (CMVS). The point cloud data are meshed by MeshLab, and then the environment map is finally constructed.

Once the environment map is configured, Unity3D visualizes the environment map and takes care of all the control of the drone. The current drone is maintained in the position and orientation in which the last image of the

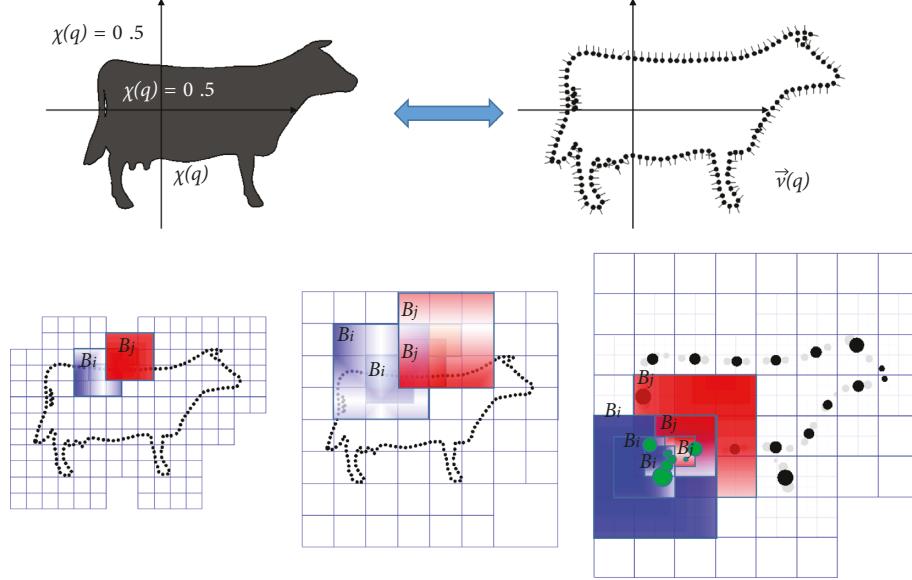


FIGURE 9: Screened Poisson surface reconstruction [23, 24].

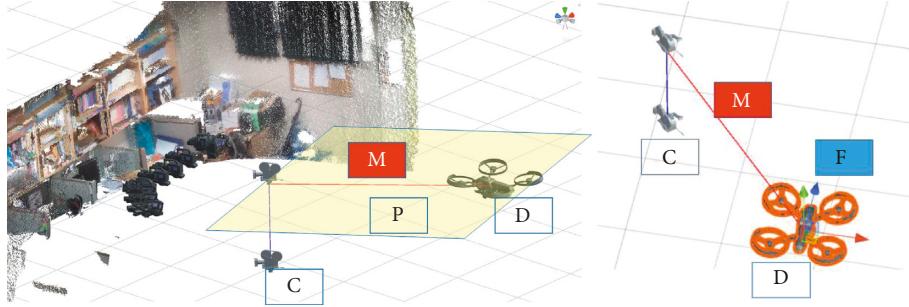


FIGURE 10: Location estimation through virtual camera analysis.

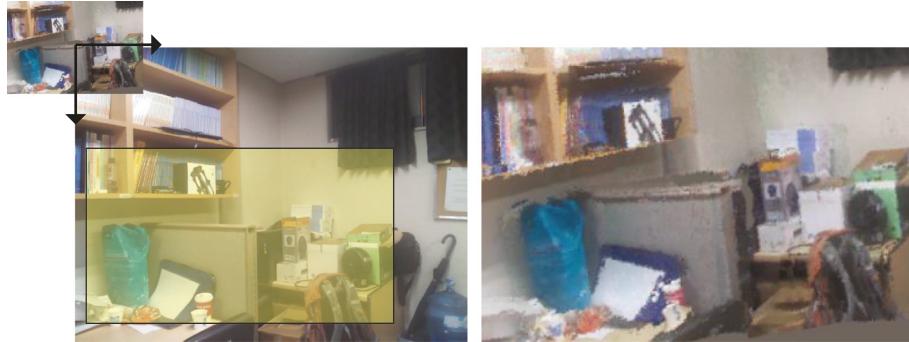


FIGURE 11: Template matching.

environment map configuration was obtained. Based on this, the origin and direction (forward vector) are matched in the world coordinate system of Unity3D. Once the basic setup is complete, the location can be confirmed via the virtual camera.

Figure 12 depicts the current location estimate. The yaw rotation of the drone can be thought of as a camera that reflects a certain range in the space coordinates. As shown in the figure, since the current position is estimated except for

the back surface of the plane P which the current drone cannot be seen, the calculation amount can be reduced to 1/2. At this time, the normal vector of the plane P is in the same direction as the forward direction of the drone. It also searches at the last position of the drone, allowing faster calculation.

Once the environment map is configured, the current position of the drone is confirmed, and then a controller for controlling the drone is not required. It is possible to set the



FIGURE 12: Search area and projection viewable area by yaw rotation amount.



FIGURE 13: System configuration.

target coordinates of the drone in Unity3D. When the target coordinate is set to P as shown in Figure 15, the horizontal-vertical movement path is calculated as shown in Figure 10.

As shown in Figure 15, if a certain frame image is transferred to the station while moving horizontally, the similarity is calculated as shown in Figure 15 using template matching. In the horizontal movement, the virtual image

(T1) of the horizontal target point is located in the middle region of the actual image, and each image becomes 1:1 scale when the horizontal target point is reached. The initial coordinate of the drone is $x: -0.66, y: 1.149, z: 6.425$ and azimuth is $x: 0, y: -90.8, z: 0$. The target coordinate of the horizontal movement is $x: -0.405, y: 1.026, z: 2.7$, and the azimuth angle is $x: 0, y: -187, z: 0$.

When the horizontal movement is completed, the current image is transmitted to the station and the vertical movement is performed. The vertical movement is sometimes not possible with template matching because it is close to the wall or when there is a large change in the image between the instantaneous arrival point and the final arrival point in a narrow and high space. Therefore, template matching is performed by placing virtual camera images at regular intervals. Since the experimental environment of this paper is not so large, there is no big problem.

As shown in Figure 16, the vertical movement coordinate is $x: -0.405, y: -0.124, z: 2.7$, and the azimuth is equal to the azimuth of horizontal movement.

When the final target is reached, the current image is transmitted to the station and the position of the drone is updated as described in Section 2.3.

As in the experiments, it is possible to obtain the relative position of the drone and control the drone in the virtual space without the controller of the actual drone. But there is some problem, the camera is very sensitive to environmental illumination condition and it is a time-consuming process to construct a detailed virtual space using a sequence of images. Also an error exists. In addition, the small drone is vulnerable to hovering, so the drone camera images may not be clear. In Figure 15, the target point (T2) of the horizontal movement was not suitable for template matching because it was too far from the starting point of the drone. In this case, a good result can be obtained by placing a virtual camera at a coordinate closer (T1) to the starting point of the drone than the coordinate T2 and performing a matching procedure. This is considered to be a phenomenon caused by the qualitative effect of the virtual space or the low resolution of the camera.

An alternative to the proposed method is the RSSI triangulation method. The method of determining the overlapping point based on the three devices that emit the signal is shown in Figure 17(a). It is relatively easy to install, low cost, and easy to implement. However, it is

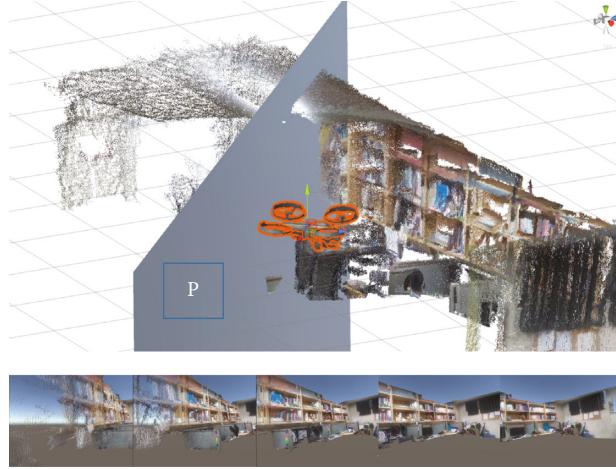


FIGURE 14: Sparse/dense/Poisson reconstruction.



FIGURE 15: After Poisson surface reconstruction, horizontal movement template matching process (virtual/drone camera, drone position). T1, T2: templates.

a two-dimensional estimation and the error rate is high because it generated inadequate noise, and latency is high in real-time tracking. To address this problem, another group estimates the height using ToF camera as shown in Figure 17(b). The ULPS measures the two-dimensional coordinates and the ToF measures the height of the drone

to complete the three-dimensional coordinates. However, there is a limitation in the location estimation of the 2D space and height estimation can be disturbed by the indoor structure. On the contrary, the proposed method takes a lot of time to construct the virtual space, but it has the advantage of being relatively accurate and posing fewer



FIGURE 16: Vertical movement of a virtual camera (drone position, drone/virtual camera) template matching.

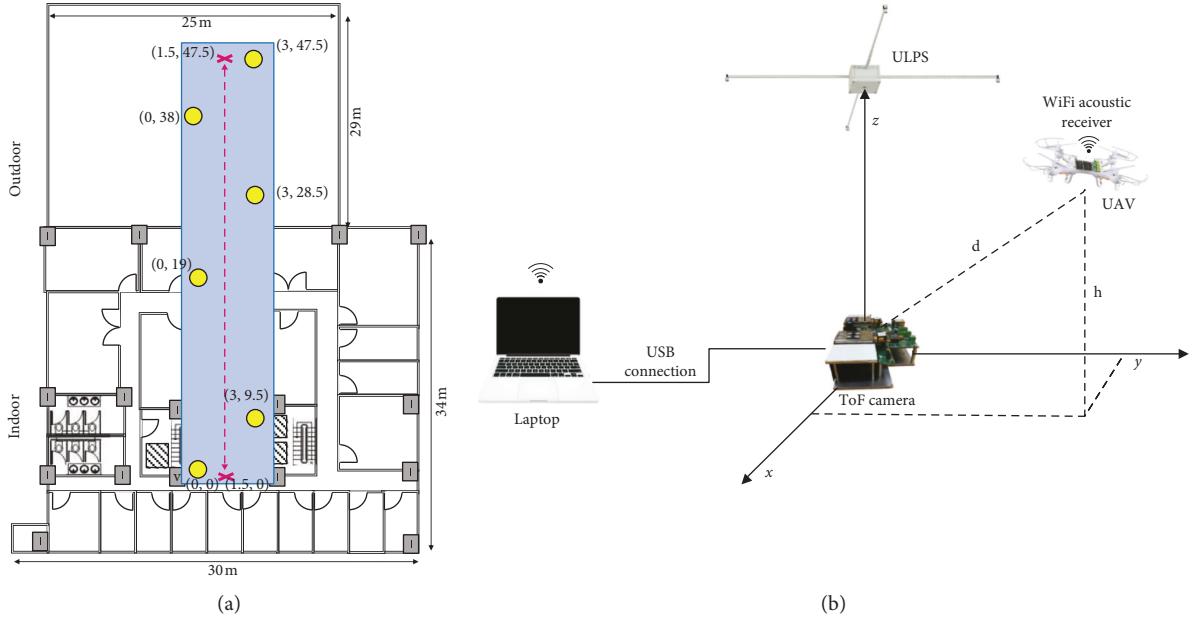


FIGURE 17: (a) Installation for RSSI triangulation [15]. (b) Position estimation with ToF [16].

constraints because the position is obtained based on the similarity between virtual and real space. Of course, direct performance comparison is difficult.

5. Conclusion

The method proposed in this paper shows that it is possible to track the location of the drone using only a single-view camera in the indoor environment. Compared with the position tracking through various sensors method, even

though the 3D restoration process takes a relatively long computational time and cannot be projected in real time, the experimental results guarantee that the accuracy is improved by position correction and image processing. Furthermore, in the proposed method, it is possible to estimate the position of the drone without installing sensors and make 3D reconstruction through additional calculations for shaded areas. Also, by controlling the target point without going through the existing manual controller, the present study may implicate further

research and development on group and/or autonomous flight.

By increasing the power of the station system and using the GPU-based parallel processing, it can be expected to complement the current visual shortcomings of the proposed method and it is expected that better results will be obtained by applying multiview camera or RGB-D sensor in the future.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [2] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: speeded up robust features," in *Proceedings of European Conference on Computer Vision*, pp. 404–417, Springer, Graz, Austria, May 2006.
- [3] C. Tomasi and T. Kanade, "Detection and tracking of point features," Tech. Rep. CMU-CS-91-132, Carnegie Mellon University, Pittsburgh, PA, USA, 1991.
- [4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2nd edition, 2000.
- [5] R. Shah, A. Deshpande, and P. J. Narayanan, "Multistage SFM: revisiting incremental structure from motion," in *Proceedings of 2nd International Conference on 3D Vision (3DV)*, pp. 417–424, IEEE, Qingdao, China, December 2014.
- [6] A. W. Fitzgibbon and A. Zisserman, *Automatic Camera Tracking*, Springer, Boston, MA, USA, 2003.
- [7] M. Han and T. Kanade, "Creating 3D models with uncalibrated cameras," in *Proceedings of Applications of Computer Vision*, pp. 178–185, IEEE, Palm Springs, CA, USA, September 2000.
- [8] M. Pollefeys, L. V. Gool, M. Vergauwen et al., "Visual modeling with a hand-held camera," *International Journal of Computer Vision*, vol. 59, no. 3, pp. 207–232, 2004.
- [9] A. Strupczewski and B. Czupryński, "3D reconstruction software comparison for short sequences," in *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2014*, International Society for Optics and Photonics, Bellingham, WA, USA, 2014.
- [10] C. Wu, "Towards linear-time incremental structure from motion," in *Proceedings of 2013 International Conference on 3D Vision*, pp. 127–134, IEEE, Seattle, WA, USA, June 2013.
- [11] R. I. Hartley and P. Sturm, "Triangulation," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146–157, 1997.
- [12] Y. Y. Wang, *Multi-View Indoor Spatial Layout Estimation*, Stanford University, Stanford, CA, USA, 2016.
- [13] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3D," in *Proceedings of ACM Transactions on Graphics (TOG)*, pp. 835–846, ACM, New York City, NY, USA, April 2006.
- [14] J. C. Ryan, A. L. Hubbard, J. E. Box et al., "UAV photogrammetry and structure from motion to assess calving dynamics at store glacier, a large outlet draining the Greenland ice sheet," *Cryosphere*, vol. 9, no. 1, pp. 1–10, 2015.
- [15] E. E. L. Lau, B. G. Lee, S. C. Lee, and W. Y. Chung, "Enhanced RSSI-based high accuracy real-time user location tracking system for indoor and outdoor environments," *International Journal on Smart Sensing and Intelligent Systems*, vol. 1, no. 2, pp. 534–548, 2008.
- [16] J. Paredes, F. Álvarez, T. Aguilera, and J. Villadangos, "3D indoor positioning of UAVs with spread spectrum ultrasound and time-of-flight cameras," *Sensors*, vol. 18, no. 2, p. 89, 2017.
- [17] Y. Furukawa and C. Hernández, "Multi-view stereo: a tutorial," *Foundations and Trends® in Computer Graphics and Vision*, vol. 9, no. 1-2, pp. 1–148, 2015.
- [18] C. Engels, H. Stewénius, and D. Nistér, "Bundle adjustment rules," in *Proceedings of Symposium on ISPRS Commission III Photogrammetric Computer Vision PCV'06*, pp. 266–271, Photogrammetric Computer Vision, Bonn, Germany, September 2006.
- [19] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.
- [20] M. Bujnák, *Dense Reconstruction from Uncalibrated Video*, Comenius University, Bratislava, Europe, 2005.
- [21] J. Dietrich's, "Advanced geographic research blog," 2014, <http://adv-geo-research.blogspot.com/2014/02/camera-geometries-for-structure-from.html>.
- [22] Z. Zhang, *Iterative point matching for registration of free-form curves*, Ph.D. Thesis, Inria, Le Chesnay, France, 1992.
- [23] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proceedings of Fourth Eurographics Symposium on Geometry Processing*, pp. 61–70, Eurographics Association, Cagliari, Sardinia, Italy, June 2006.
- [24] M. Kazhdan and H. Hoppe, "Screened poisson surface reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 3, p. 29, 2013.
- [25] C. Wu, *VisualSfM: A Visual Structure from Motion System*, 2011, <http://ccwu.me/vsfm/doc.html>.

