sensors

MDPI

Article

Object Occlusion Detection Using Automatic Camera Calibration for a Wide-Area Video Surveillance System

Jaehoon Jung [1], Inhye Yoon [1,2] and Joonki Paik [1,*]

[1] Department of Image, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 06974, Korea; gjslkjs@gmail.com (J.J.); inhyey@gmail.com (I.Y.)

[2] ADAS Camera Team, LG Electronics, 322 Gyeongmyeong-daero, Seo-gu, Incheon 22744, Korea

* Correspondence: paikj@cau.ac.kr; Tel.: +82-10-7123-6846

Academic Editor: Gonzalo Pajares Martinsanz
Received: 3 May 2016; Accepted: 23 June 2016; Published: 25 June 2016

Abstract: This paper presents an object occlusion detection algorithm using object depth information that is estimated by automatic camera calibration. The object occlusion problem is a major factor to degrade the performance of object tracking and recognition. To detect an object occlusion, the proposed algorithm consists of three steps: (i) automatic camera calibration using both moving objects and a background structure; (ii) object depth estimation; and (iii) detection of occluded regions. The proposed algorithm estimates the depth of the object without extra sensors but with a generic red, green and blue (RGB) camera. As a result, the proposed algorithm can be applied to improve the performance of object tracking and object recognition algorithms for video surveillance systems.

Keywords: occlusion detection; automatic camera calibration; depth estimation; moving object detection; video surveillance system

1. Introduction

Recently, the demand for object tracking and recognition algorithms is increasing due to video surveillance. An object occlusion is a major factor for the performance degradation of a video surveillance system. For this reason, various object occlusion detection and handling methods were studied.

Mei et al. proposed an object tracking with consideration of occlusion that is detected using the occlusion map [1]. Since this method uses a target template to obtain the occlusion map, it is difficult to detect the object occlusion when the target template is unavailable. Zitnick et al. generated a depth map using a stereo camera and detected object occlusion regions [2]. However, this method needs two cameras for stereo matching to generate the depth map. Sun et al. proposed an optimization approach using the visibility constraint for the stereo matching and then generated the depth map by minimizing the energy function [3]. Since a stereo camera-based occlusion detection method needs an additional camera, it is not easy to implement in an already installed wide-area surveillance system.

To solve this problem, single camera based depth map estimation methods were proposed. Matyunin et al. estimated the depth using an infrared sensor [4]. However, this method cannot work in the outdoor scene since an infrared sensor is interrupted by sunlight. Im et al. proposed a single red, green, and blue (RGB) camera-based object depth estimation method using multiple color-filter apertures (MCA) [5]. However, this method needs a special aperture for the object depth estimation, and produces color distortion at boundary of the out-focused objects. Zonglei et al. used a patterned box for semi-automatic camera calibration [6]. Lin et al. estimated vanishing points using traffic lanes, and estimated the distance of a frontal vehicle using a single RGB camera for a collision warning system [7]. Since this method uses the traffic lane for vanishing point estimation, distance estimation is impossible when an input image does not contain a traffic lane. Song et al. detected features from a moving object, and automatically calibrated the camera [8]. However, Song's method cannot avoid the camera calibration error when feature points change while the object is moving.

To solve these problems, the proposed method first performs automatic camera calibration using both moving objects and background structures to estimate camera parameters. Given the camera parameters, the proposed algorithm estimates the object depth with regard to a reference plane, and then detects the object occlusion. To estimate vanishing points and lines, the proposed algorithm detects parallel lines in the input image. Bo et al. detected straight lines from background structures using the one-dimensional (1D) Hough transform for automatic camera calibration [9]. Since this method uses only a single image, it is impossible to automatically calibrate the camera when the background does not have line components. Moreover, accuracy of the camera calibration is degraded by with non-parallel lines. Lv et al. detected human foot and head points for automatic camera calibration [10]. However, if the estimated foot and head points are not sufficiently accurate or if the object motion is linear, the camera calibration is impossible. Moreover, accuracy of the camera calibration result depends on the object detection results. To solve these problems, the proposed algorithm combines the background structure lines with human foot and head information to estimate vanishing points and lines.

This paper is organized as follows: Section 2 describes background theory of the camera geometry, and Section 3 presents the proposed object occlusion detection algorithm. Experimental results of the proposed algorithm are shown in Section 4, and Section 5 concludes the paper.

2. Theoretical Background of Camera Geometry

Estimation of the depth needs a 3D space information. To obtain the projective relationship between the 2D image and 3D space information, a camera geometry is used with camera parameter that describe camera sensor, lens, optical axis, and a position of the camera in the world coordinate. In the pin-hole camera model [11], a point in the 3D space is projected onto a point in the 2D image as

$$s\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & skew & p_x \\ 0 & f_y & p_y \\ 0 & 0 & a \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathrm{A}[\mathrm{R}|t]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

where s represents the scale, $[\ x \quad y \quad 1\ ]^T$ a point in the 2D image, matrix A consists of intrinsic camera parameters, f_x and f_y focal length in the x- and y-axis focal length, respectively, $skew$ the skewness, a the aspect ratio, camera rotation matrix R consists of camera rotation parameters r_{ij}, $[\ t_1 \quad t_2 \quad t_3\ ]^T$ the camera translation vector, and $[\ X \quad Y \quad Z \quad 1\ ]^T$ a point in the 3D space.

To simplify the description without loss of generality, we assume that the focal lengths f_x and f_y are equivalent, the principal point is at the image center, the skewness is equal to zero, and the aspect ratio is equal to 1. In the same manner, we also assume that the camera rotation angle with regard to the Z-axis is equal to zero, and the camera translation with regard to both X- and Y-axis is equal to zero to calculate the extrinsic matrix $[\mathrm{R}|t]$ as

$$[\mathrm{R}|t] = [R_Z(\rho)R_X(\theta)T(0,0,h_c)] \quad (2)$$

where R_Z represents the rotation matrix with regard to the Z-axis, R_X the rotation matrix with regard to the X-axis, T the transformation for a translation, h_c the camera height.

The 2D image is generated by the light that is reflected by an object and than arrives at the camera sensor. In this process, a single object is projected onto the 2D image plane with different sizes according to the distance from the camera as shown in Figure 1. For this reason, parallel lines in the 3D space are projected on the 2D image plane as non-parallel lines depending on the depth. Using the apparent non-parallel lines in the image, the vanishing points can be estimated as a intersected points of those lines. Since the projective camera transformation model projects a point in the 3D space onto the camera sensor, the camera parameters can be estimated using vanishing points.

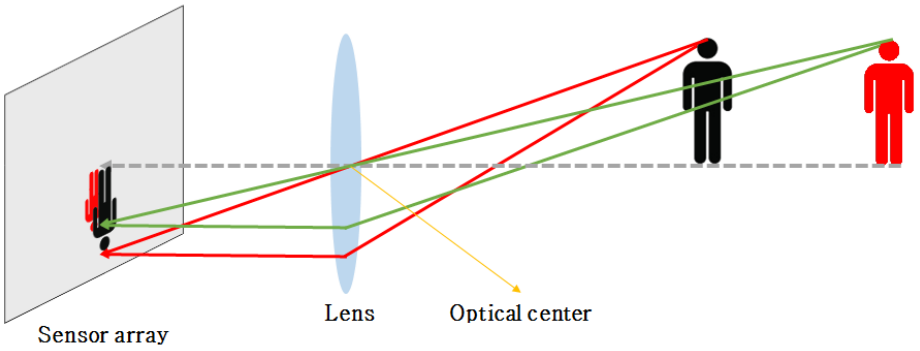


Figure 1. Projective model of the camera.

The point in the 3D space is projected onto a camera sensor corresponding to a point in the 2D image using a projective transform. However, a point in the 2D image cannot be inversely projected onto a unique point in the 3D space since the camera projection transform is not a one-to-one function. On the other hand, if there is a reference plane, a point in the 2D image can be inversely projected onto a point on the reference plane that is defined in the 3D space. As a result, the proposed algorithm estimates the object depth using the 2D image based on a pre-specified reference plane.

3. Automatic Calibration-Based Occlusion Detection

The proposed object occlusion detection algorithm consists of three steps: (i) automatic camera calibration; (ii) object depth estimation; and (iii) occlusion detection. Figure 2 shows the block diagram of the proposed algorithm, where I_k represents the k-th input frame, L represents the extracted lines, P represents the projective matrix, D represents the object depth information, and O represents the detected region of an occluded object.

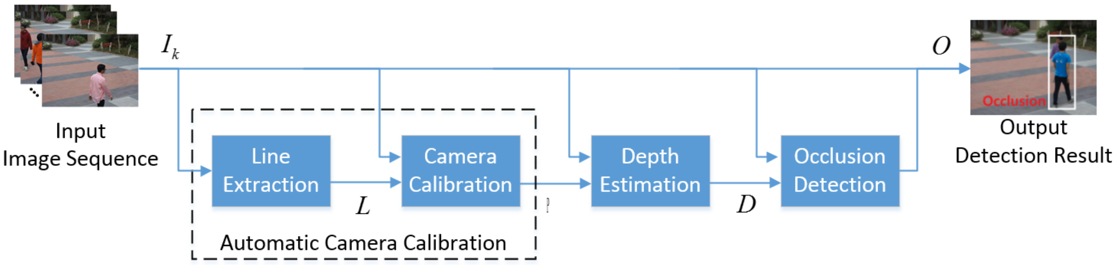


Figure 2. Block diagram of the proposed occlusion detection algorithm.

3.1. Automatic Camera Calibration

The proposed algorithm estimates the camera parameters for the object depth estimation followed by object occlusion detection. Since semi-automatic camera calibration is the simplest way to estimate parameters using a synthetic calibration pattern [6], its performance is highly dependent on the experience of a user. To solve this problem, the proposed algorithm uses an automatic camera calibration method that extracts lines from the image, and then estimates vanishing points and lines [12].

To detect human foot and head points, the proposed algorithm detects the foreground by modeling the background using the Gaussian mixture model (GMM) [13]. The object region is then detected by labeling a sufficiently large object. Given an object region, the vertically highest point is determined as the head point. On the other hand, the average of the bottom 20 percent points is determined as the foot point.

A pair of parallel lines that connect head points and foot points are used to detect vanishing points and lines. Since the lines connecting head points and foot points are non-parallel when the height of an object changes while walking, the proposed algorithm detects the uniform height of the object only when pedestrian's legs are crossing as

$$\frac{1}{n}\sum_{i=1}^{n}(p_i - p_f)^2 < T_C \tag{3}$$

where p_i represents the candidate foot points, n the number of candidate foot points, p_f the detected foot point, and T_C the threshold value.

To combine object foot and head information with the background structure lines, the proposed algorithm detects edges that are used to detect vanishing points and lines [14]. The detected foot and head points and background structure lines are shown in Figure 3.

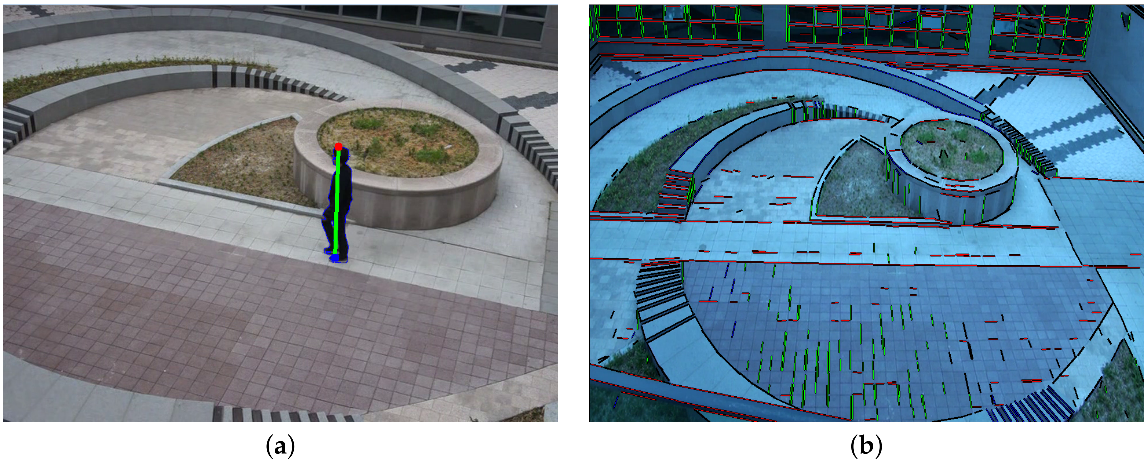

(**a**) (**b**)

Figure 3. Results of the line detection: (**a**) foot-to-head line and (**b**) background structure lines.

The vanishing points and lines are estimated using the detected foot-to-head line and background structure lines. For the robust vanishing points and lines estimation, a sufficient number of foot and head points are required. For that reason, the proposed algorithm estimates the vanishing points and lines depending on the number of the human foot and head points according to the following three cases:

Theoretically, homography estimation for camera calibration requires four 2D coordinates that can solve eight linear equations. However, a practical random sample consensus (RANSAC) based robust camera calibration needs at least eight points such that the calibration error is minimized as shown in Figure 4.

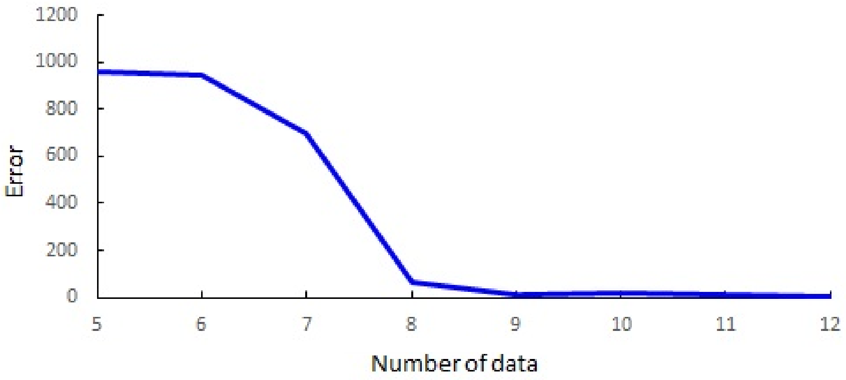


Figure 4. Focal length estimation error depending on the number of foot-head data sets with 30% outliers.

Case 1. If the number of detected foot and head point sets is less than N, the vanishing points and lines are estimated using background structure lines. More specifically, three vanishing points are selected from background lines intersecting points using the RANSAC algorithm. Among three vanishing points, the lowest one is determined as the vertical vanishing point. The line connecting the remaining two vanishing point is determined as the horizontal vanishing line.

Case 2. If the number of detected foot and head sets is more than N but the object motion is linear, the vertical vanishing point can be estimated only using the object foot and head points. The vertical vanishing point is determined at the intersected point of foot-to-head lines as shown in Figure 5. However, if the object moves linearly, estimation of a horizontal vanishing line is impossible since only one horizontal vanishing point is estimated. In this case, the vanishing line is estimated using background structure lines.

Case 3. If the number of detected foot and head sets is more than N and object motion is not linear, vanishing points and lines can be estimated using foot and head points. A foot-to-foot line that connects two foot points and the corresponding head-to-head line that connects two head points are used to estimate the horizontal vanishing points. As a result, the horizontal vanishing line can be estimated using the two horizontal vanishing points as shown in Figure 5.

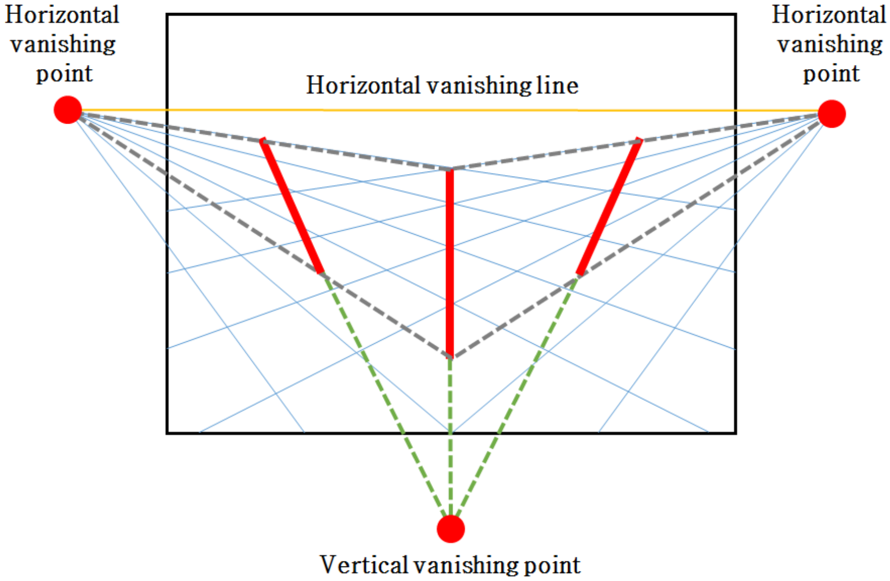

Horizontal vanishing point
Horizontal vanishing line
Horizontal vanishing point
Vertical vanishing point

Figure 5. Definition of vanishing points and the horizontal vanishing line.

Camera parameters are calculated using the estimated vertical vanishing point and the horizontal vanishing line as [15]

$$\begin{aligned} f &= \sqrt{(a_3/a_2 - p_y)(v_y - p_y)} \\ \rho &= \text{atan}(-v_x/v_y) \\ \theta &= \text{atan}(-\sqrt{v_x^2 + v_y^2}/f) \\ h_c &= h_o/(1 - \tfrac{d(o_h,v_l)\|o_f - v\|}{d(o_f,v_l)\|o_h - v\|}) \end{aligned} \tag{4}$$

where f represents the focal length, ρ the roll angle, θ the tilt angle, h_c the camera height, v_l the horizontal vanishing line $a_1x + a_2y + a_3 = 0$, $v = [\ v_x \quad v_y\ ]^T$ the vertical vanishing point, h_o the object height, o_f the object foot point, o_h the object head point, and $d(A, B)$ the distance between a point A and a line B.

3.2. Object Depth Estimation and Occluded Region Detection

The proposed algorithm uses object depth information to detect an occluded region. To estimate the depth of an object, the 2D image coordinate is projected onto the reference plane in the 3D space using a projective matrix. Since the object foot points should be on the ground plane, the proposed algorithm uses the ground plane as the reference plane, which means that the ground plane is considered as the XY plane because the camera height is calculated as the distance between the ground plane and the camera. Using an object foot point in the 2D image, the object foot point on the ground plane in the 3D space can be calculated. To detect the foot point in the 3D space, a foot point in the 2D image is inversely projected onto the 3D space using a projective matrix as

$$\mathrm{X} = \left(\mathrm{P}^T\mathrm{P}\right)^{-1}\mathrm{P}^T\mathrm{x}_f \tag{5}$$

where x_f represents the foot point in the 2D image, matrix P the projective matrix, and X the inversely projected coordinate of x_f. Inversely projected coordinate X is normalized by the Z-axis value to detect the foot point in the 3D space as

$$\mathrm{X}_f = \frac{\mathrm{X}}{\mathrm{Z}} \tag{6}$$

where Z represents the Z-axis value of X, and X_f the foot point on the ground plane in the 3D space.

An object depth is estimated by computing the distance between the object and camera. However, the foot point appears in the finite position in the input image. For this reason, the proposed algorithm uses the nearest foot point as a pivot point for the object depth estimation. The estimated depth is then normalized using the farthest distance. If an object is far enough from the camera, depth of the object foot point is assumed to be the Y-axis coordinate since the camera pan angle is equal to zero and the pivot point is on the ground plane. If the object depth is equal to the object foot point depth, the object depth is calculated as

$$d = \frac{\left|Y_f - Y_p\right|}{d_F} \tag{7}$$

where d represents the object depth, Y_p the Y-axis value of the pivot point, Y_f the object foot point, and d_F the farthest distance. Figure 6 shows the proposed object depth estimation model, where d_N represents the nearest distance, (X_p, Y_p, Z_p) the pivot point, and (X_f, Y_f, Z_f) the object foot point.

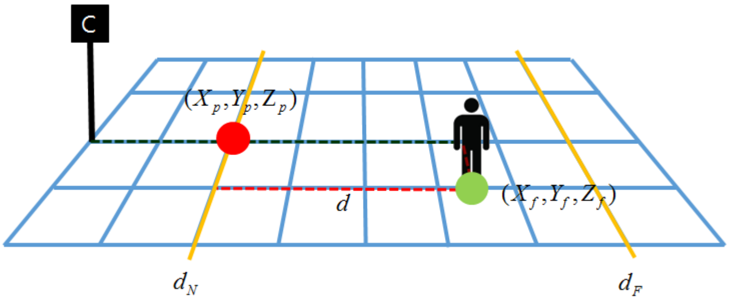


Figure 6. The proposed object depth estimation model.

The proposed algorithm detects the object occlusion using the estimated object depth. The depth of the same object in adjacent frames slowly changes. On the other hand, if the object is occluded, the estimated depth of the object rapidly changes. Based on the observation, object occlusion is detected as

$$O = \begin{cases} \text{true}, & \text{if } |d_{t-1} - d_t| \geq T_O \\ \text{false}, & \text{otherwise} \end{cases} \tag{8}$$

where O represents the object occlusion detection result, d_t the depth of the object at time t, and T_O the threshold value for the object occlusion detection. We can only estimate depth from standing human objects whose feet lie on the reference plane assuming that each object has a uniform depth.

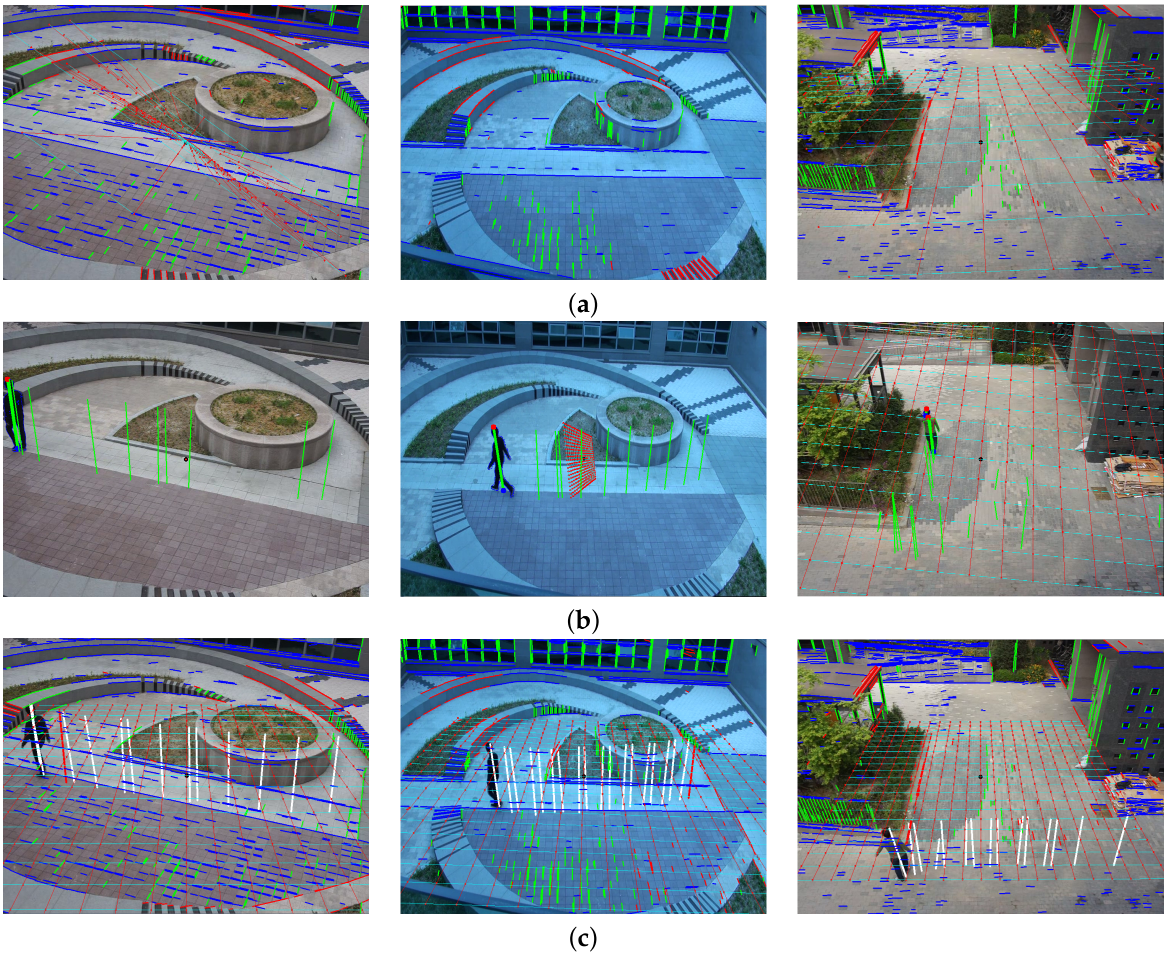
4. Experimental Results

This section shows the results of the proposed automatic camera calibration and object occlusion detection algorithms. For the experiment, test video sequences of resolution 1280 × 720 were acquired using a camera installed at 2.2 to 7.2 m high. In addition to the in-house test sets, Vision and Autonomous System Center's (VASC) stereo dataset is also used to compare the performance of the proposed method with existing stereo matching-based methods [16].

Figure 7 shows the result of three different methods for automatic camera calibration. A ground plane is drawn on the image using grid lines with a 0.5 m interval to show the accuracy of the camera calibration. The background structure-based method makes a poor calibration result because of insufficient, non-parallel line segments and random textures of natural objects as shown in Figure 7a. The moving object-based method degrades the calibration performance because of the incompletely detected moving object and only linear motion of the object as shown in Figure 7b. On the other hand, the proposed method significantly improves the accuracy of camera calibration because it uses neither incomplete background structures nor multiple object positions in the same line as shown in Figure 7c.

Figure 8 shows the result of object depth estimation using the proposed method. Figure 8b shows the calibration result with the superimposed ground plane. The estimated depths of moving objects are shown in Figure 8c.

Figure 9 compares depth estimation results using the stereo matching-based and the proposed methods. Figure 9a,b respectively show the left and right images of the "Toy" in the VASC stereo dataset. Figure 9c shows the stereo matching-based depth estimation result. Figure 9d shows guidelines to detect an region, where red lines represent the objects bottom boundary and blue lines the object's top boundary. Figure 9e shows the superimposed grid of the reference plane that is the camera calibration result. The calibration result using the proposed object depth estimation method is shown in Figure 9f. Although the stereo matching-based method generated many holes in textureless regions without features, the proposed method successfully estimated the continuous depth map.


(**a**)

(**b**)

(**c**)

Figure 7. Results of three different method for camera calibration: (**a**) background structure-based method; (**b**) moving object-based method; and (**c**) the proposed camera calibration method.

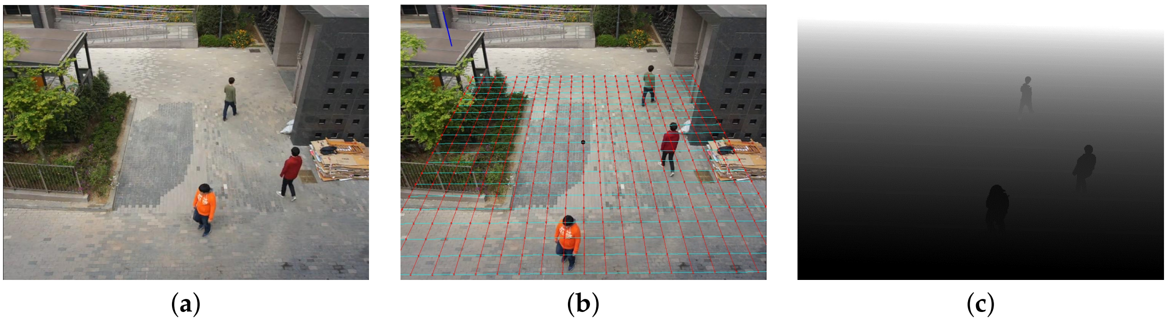
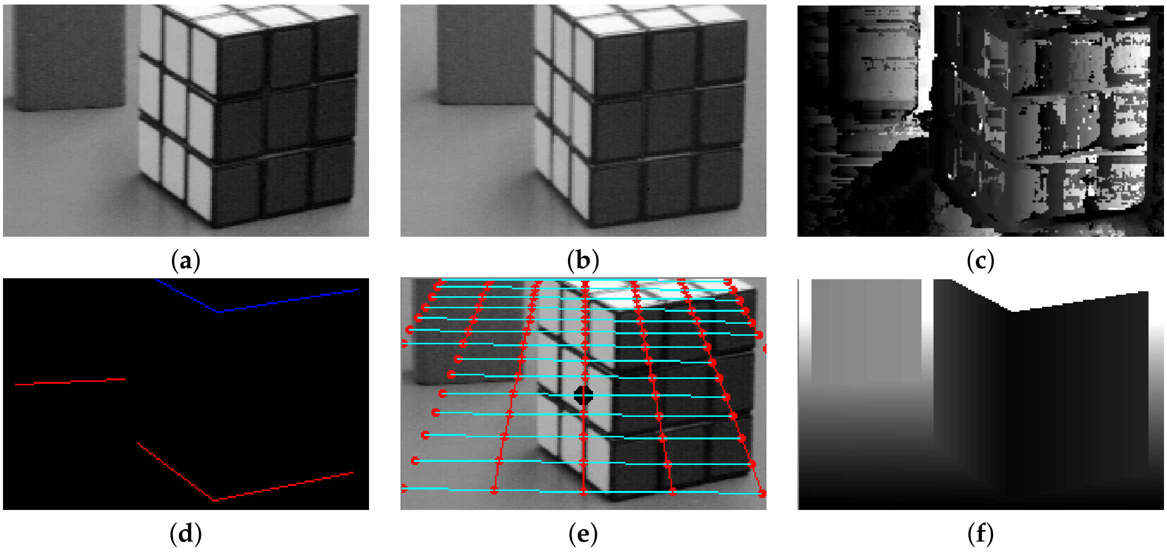

(**a**) (**b**) (**c**)

Figure 8. The results of the proposed object depth estimation: (**a**) input image; (**b**) calibration result in the form of the superimposed grid representing the ground plane; and (**c**) the depth estimation result.

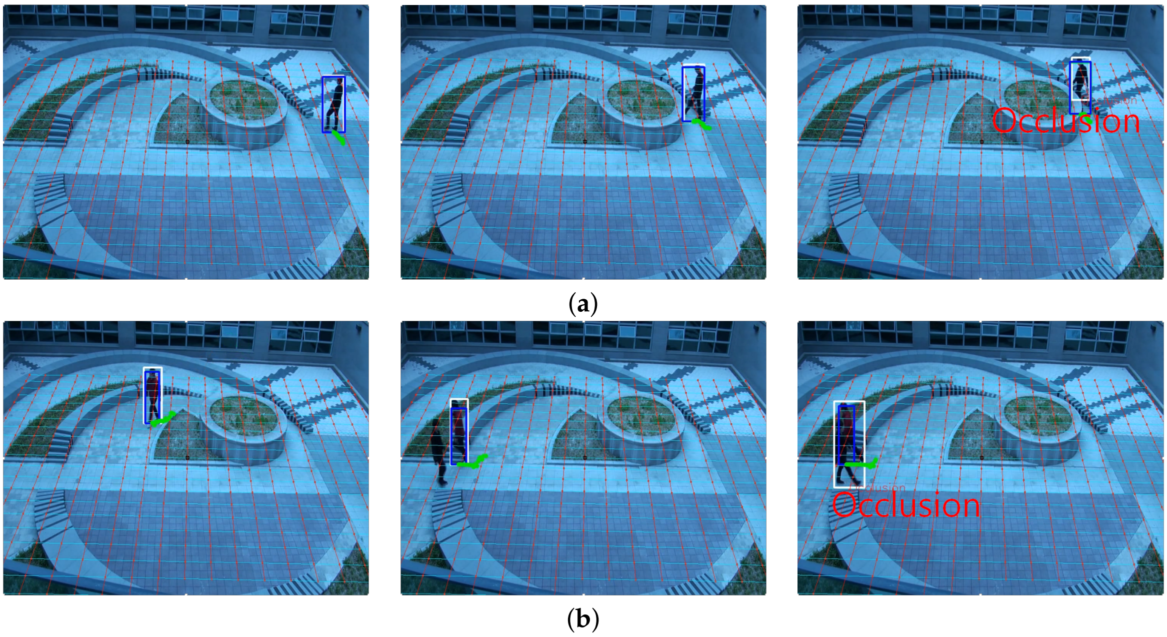


(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 9. Comparison of depth estimation results using the stereo-based and the proposed methods: (**a**) the left stereo image; (**b**) the right stereo image; (**c**) estimated depth map using the stereo matching-based method; (**d**) guide lines of the left image; (**e**) estimated ground plane of the left image; and (**f**) estimated depth map of the left image using the proposed method.

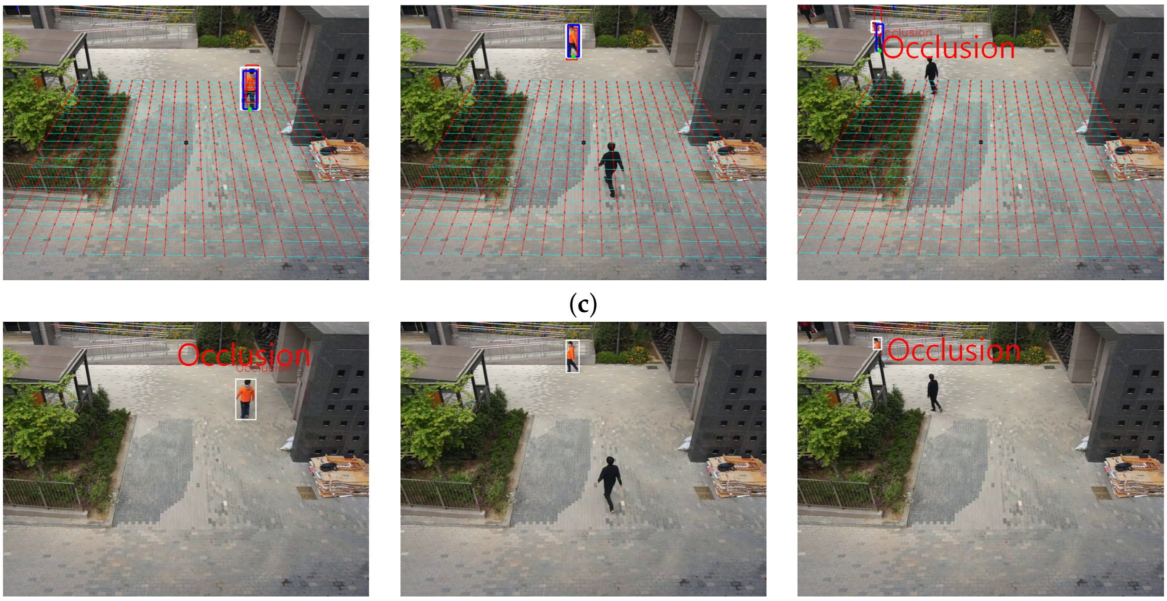
Figure 10 shows the detection result of an occluded object using the proposed occlusion detection algorithm with the threshold distance of 1.0 m. Figure 10a shows the detection result of the occluded object by a background structure. Figure 10b shows the detection result of the occluded object by another object, and Figure 10c shows the detection result in a different test video. The proposed method can successively detect occlusion in various test videos. As shown in Figure 10d detection of the y-axis value of an object foot position may results in erroneous detection of occlusion. However, the proposed method can correctly detect occlusion in the scene-invariant manner since it uses the depth in formation in the 3D space.



(**a**)

(**b**)

Figure 10. *Cont.*



(**c**)

(**d**)

Figure 10. Results of occlusion detection three selected frames in each video: (**a**) occlusion by background; (**b**) occlusion by another object; (**c**) result of occlusion detection detection in another video and (**d**) result occlusion detection without depth information.

5. Conclusions

In this paper, we presented a fully automatic object occlusion detection method by estimating the object depth from a single uncalibrated camera. The proposed algorithm can robustly calibrate a camera by combining the background structure line components and moving object information. In addition, object depth is estimated using a single RGB camera. As a result, the object occlusion is successfully detected by analyzing the object depth information. The proposed method can be applied to object detection and tracking in a multiple-view surveillance system.

The fundamental assumption of the proposed occlusion detection algorithm is that there is a single, flat ground on which all objects move around. If the ground is not flat or slanted, the estimated depth becomes inaccurate, and as a result, object detection may fail. In that case, the nonflat ground can be approximated by piece-wise flat one, and the slanting ground can be taken care of in the calibration process. In spite of the restrictions, the proposed method is suitable for a wide range of surveillance applications, such as multiple camera video tracking with object handover and normalized metadata generation-based video indexing and retrieval because of its economical implementation.

Acknowledgments: This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2016-H8501-16-1018) supervised by the IITP (Institute for Information & Communications Technology Promotion), by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (B0101-15-0525, Development of global multi-target tracking and event prediction techniques based on real-time large-scale video analysis), and by the Technology Innovation Program (Development of Smart Video/Audio Surveillance SoC & Core Component for Onsite Decision Security System) under Grant 10047788.

Author Contributions: Jaehoon Jung and Inhye Yoon initiated the research and designed the experiments, and Joonki Paik wrote the paper.

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

References

1. Mei, X.; Ling, H.; Wu, Y.; Blasch, E.; Bai, L. Minimum error bounded efficient l1 tracker with occlusion detection. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 20–25 June 2011; pp. 1257–1264.
2. Zitnick, C.L.; Kanade, T. A cooperative algorithm for stereo matching and occlusion detection. *Pattern Anal. Mach. Intell. IEEE Trans.* **2000**, *22*, 675–684.
3. Sun, J.; Li, Y.; Kang, S.B.; Shum, H.Y. Symmetric stereo matching for occlusion handling. In Proceedings of the 2015 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 399–406.
4. Matyunin, S.; Vatolin, D.; Berdnikov, Y.; Smirnov, M. Temporal filtering for depth maps generated by Kinect depth camera. In Proceedings of the 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, Antalya, Turkey, 16–18 May 2011; pp. 16–18.
5. Lm, J.; Jung, J.; Paik, J. Single camera-based depth estimation and improved continuously adaptive mean shift algorithm for tracking occluded objects. In Proceedings of the 16th Pacific-Rim Conference on Advances in Multimedia Information Processing, Gwangju, Korea, 16–18 September 2015; pp. 246–252.
6. Huang, Z.; Boufama, B. A semi-automatic camera calibration method for augmented reality. In Proceedings of the IEEE International Conference on System, Man and Cybernetics, Yasmine Hammamet, Tunisia, 6–9 October 2002.
7. Lin, H.Y.; Chen, L.Q.; Lin, Y.H.; Yu, M.S. Lane departure and front collision warning using a single camera. In Proceedings of the 2012 IEEE International Symposium on Intelligent Signal Processing and Communications Systems, New Taipei, Taiwan, 4–7 November 2012; pp. 64–69.
8. Song, Y.; Wang, F.; Yang, H.; Gao, S. Easy to calib: Auto-calibration of camera from sequential images based on VP and EKF. In Proceedings of the 2014 International Conference on Innovative Computing Technology, Luton, UK, 13–15 August 2014; pp. 41–45.
9. Li, B.; Peng, K.; Ying, X.; Zha, H. Vanishing point detection using cascaded 1D Hough transform single images. *Pattern Recognit. Lett.* **2012**, *33*, 1–8.
10. Lv, F.; Zhao, T.; Nevatia, R. Camera calibration from video of a walking human. *Pattern Anal. Mach. Intell. IEEE Trans.* **2006**, *28*, 1513–1518.
11. Cipolla, R.; Drummond, T.; Robertson, D.P. Camera calibration from vanishing points in image of architectural scenes. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 13–16 September 1999; Volume 2, pp. 382–391.
12. Beardsley, P.; Murray, D. Camera calibration using vanishing points. In Proceedings of the British Machine Vision Conference, Leeds, UK, 22–24 September 1992; pp. 416–425.
13. Stauffer, C.; Grimson, W.E.L. Adaptive background mixture models for real–time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 2.
14. Lutton, E.; Maitre, H.; Lopez-Krahe, J. Contribution to the determination of vanishing points using Hough transform. *Pattern Anal. Mach. Intell. IEEE Trans.* **2002**, *16*, 430–438.
15. Liu, J.; Collins, R.T.; Liu, Y. Surveillance camera autocalibration based on pedestrian height distributions. In Proceedings of the British Machine Vision Conference, Dundee, UK, 29 August–2 September 2011; pp. 1–11.
16. Vision and Autonomous System Center's Image Dataset. Available online: http://vasc.ri.cmu.edu/idb/ (accessed on 28 July 1997).