PAPER

A Dynamic Geometry Reconstruction Technique for Mobile Devices Using Adaptive Checkerboard Recognition and Epipolar Geometry

Vinh Ninh DAO†a), ***Nonmember and*** **Masanori SUGIMOTO**†b), ***Member***

SUMMARY This paper describes a technique for reconstructing dynamic scene geometry using a handheld video projector–camera system and a single checkerboard image as a structured light pattern. The proposed technique automatically recognizes a dense checkerboard pattern under dynamic conditions. The pattern-recognition process is adaptive to different light conditions and an object's color, thereby avoiding the need to set threshold values manually for different objects when the scanning device is moving. We also propose a technique to find corresponding positions for the checkerboard pattern, when displayed by a projector, without needing any position-encoding techniques. The correspondence matching process is based on epipolar geometry, enabling the checkerboard pattern to be matched even if parts of it are occluded. By using a dense checkerboard pattern, we can construct a handheld projector–camera system that can acquire the geometry of objects in real time, and we have verified the feasibility of the proposed techniques.

key words: *checkerboard, pattern recognition, pattern matching, geometry reconstruction, handheld projector–camera*

1. Introduction

Recent advances in 3D scanning technology have made convenient and cost-effective systems possible. In particular, techniques using active structured light are well known for their accuracy and speed. However, during the past few decades, 3D scanning with these techniques has usually been aimed at static capture. Acquisition of 3D geometry in a dynamic environment remains an open research issue. Following the recent development of miniature projectors, ubiquitous projection that can present any surface as a display [4], [14] is becoming practicable. Therefore, when real-time 3D scanning systems using such miniature projectors become more readily available, we will be able to utilize them for adaptive projection and the augmented reality applications around us [4], [5].

There are "one-shot" geometry-reconstruction techniques that can acquire the 3D shape of an object from a single captured image [25], but they usually impose some conditions on the scanning surface, such as surface smoothness or constant color reflectance. If the conditions do not hold because of ambiguous colors or abrupt color changes caused by surface complexity or scanning device movement, the correspondence will be lost. Among the commonly used structured light patterns, a checkerboard pattern has not been used for geometry-acquisition purposes because it is difficult to encode the positional information in this pattern. Until now, checkerboard-recognition techniques have been proposed mainly for camera-calibration purposes [23], [36], where a checkerboard on a planar surface is used. Consequently, these techniques are usually sensitive to any distortion, occlusion, and noise in the pattern.

This paper proposes a novel technique to recognize a dense black and white checkerboard pattern for one-shot geometry-acquisition purposes (Fig. 1). The checkerboard pattern can be recognized automatically in dynamic conditions, with the internal edges being extracted adaptively. We exploit the directional properties of the feature points, the feature edges, and the graph topology to eliminate noise in the detected checkerboard pattern. This paper also presents a pattern-matching technique based on epipolar geometry that requires no position-encoding technique, with the corresponding positions being identified even if parts of the pattern are missing because of occlusion. This method enables the feature points to be extracted with high accuracy, and this enables us to easily find the corresponding position for each patch in a dense checkerboard pattern. Our experimental evaluation shows that the epipolar constraints of locally connected feature points are sufficient to solve for the correspondence robustly and effectively.

The remainder of the paper is organized as follows. The next section is about related work in checkerboard-recognition and shape-acquisition techniques. In Sect. 3, we present our method for recognizing the checkerboard pattern adaptively. This checkerboard-recognition technique

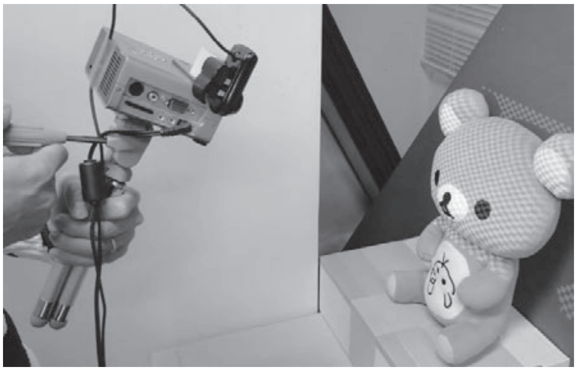


Fig. 1 A handheld geometry acquisition system using a checkerboard pattern and a mobile projector–camera system.

Manuscript received May 13, 2010.
Manuscript revised September 13, 2010.

†The authors are with the Department of Electrical Engineering and Information System, The University of Tokyo, Tokyo, 113–8656 Japan.
a) E-mail: dao@itl.t.u-tokyo.ac.jp
b) E-mail: sugi@itl.t.u-tokyo.ac.jp

DOI: 10.1587/transinf.E94.D.336

involves three steps, namely feature-point detection, feature-edge detection, noise elimination and a labeling process. Section 4 describes a pattern-matching technique for the checkerboard pattern in which we show that the simple epipolar geometry of connected feature points and feature edges is sufficient to find the corresponding positions of segments of the checkerboard pattern. In Sect. 5, experimental results are discussed. Finally, the conclusion of the paper is given in Sect. 6.

2. Related Work

2.1 Checkerboard Recognition

The checkerboard-recognition process has usually started from a corner-detection process. Then, the geometric relations between the corners are analyzed to recognize the checkerboard pattern. There are several approaches to the detection of crossing corners. A common approach to finding the checkerboard corners is to use universal corner detectors such as those proposed by Harris [34] or Smith and Brady [30]. However, these detectors require some manual setting of thresholds, and usually produce too many redundant corners that are not specific to the checkerboard crossing corner.

Alternatively, the checkerboard pattern can be detected as an array of regular rectangles or grid lines. Wang *et al.* [11] and Bouguet [36] recognize the checkerboard's corners as the intersections of horizontal and vertical lines. However, this approach can find the corners only when the checkerboard pattern is on a planar surface. Rufli *et al.* [8] proposed a method for recognizing a checkerboard pattern as a collection of rectangular areas, which resembles the widely used *cvFindChessBoardCorner* function in the OpenCV library [23], but is more robust with respect to distortion and light-condition changes by using adaptive-threshold techniques to generate the binary image and by using image morphing at different levels to achieve the best detection.

Recently, Sun *et al.* [6] proposed using a crossing-point descriptor, generated by using neighboring pixels, to recognize the corners of a distorted checkerboard pattern. This method requires minimal space to detect the crossing corner robustly without the manual setting of thresholds. However, it requires the checkerboard pattern to be fully detected for correspondence matching, with this requirement being very difficult to satisfy in geometry-reconstruction conditions that have occluded, discontinuous, or mosaic surfaces.

2.2 Edge Extraction

In most of the geometry-reconstruction techniques that use light stripes or light grid patterns, edges are the target of the pattern-extraction process because they contain dense information about the object geometry. Pure feature-edge detection has been the subject of much research in image processing, and various techniques have been proposed [30], [35]. However, these techniques are universal, and detect many false edges that are not specific to the inner edges of the checkerboard pattern. In existing checkerboard-recognition techniques, edges are used to provide information about the connections between the crossing points [6], [17]. However, these edges are approximated by straight lines for detection and noise-filtering purposes, but not for geometric information.

In shape-acquisition techniques that use light stripes or grid patterns, color may be used to identify horizontal and vertical stripes [1], [7], [13], [29]. Edges are detected by following the peak response of catoptric light on different color channels, which is heavily dependent on the quality of the object's surface and the light source. Therefore, the thresholds for edge continuity and edge detection require manual setting for different conditions. In that research, noise was either neglected [1] or dominated by over-constraint conditions [7], even though many false edges may have appeared depending on the object being scanned.

2.3 Correspondence Matching

The research into structured light techniques for acquiring 3D geometry has been well investigated and summarized by Forest and Salvi [18] and Blais [16]. Most techniques using structured light patterns can be categorized as either time encoded or space encoded, with the latter being more suited to capturing dynamic objects. Space-encoded techniques can be divided further into techniques that use local spatial encoding of color [2], [20], [32] and techniques that use projector–camera geometry constraints to solve the correspondence of the feature points [1], [7]. Our work belongs to this second category, and we will therefore discuss some of the more relevant work here.

Correspondence-matching techniques that use epipolar geometry have been widely used in one-shot shape acquisition systems. Maruyama and Abe [31] proposed using light slits with random cuts. Tateishi *et al.* [9] used a multi-spot laser projector to acquire a sparse 3D geometry of an object at extremely high speed. These studies solved the correspondence for each point by using its corresponding epipolar line. This technique is easily implemented on a hardware device to archive the high speed performance. However, the density of the pattern is strictly limited, because if there are more than two laser spots on an epipolar line, the correspondence cannot be resolved. Watanabe *et al.* [10] improved the measurement density by using massively parallel processors to track the corresponding laser spots on epipolar lines for a dense array (33 × 33). They need an initialization phase to find the corresponding position for each laser spot, and if the tracking is lost, they cannot solve the corresponding problem for each image.

Koninckx and Gool [13] used epipolar geometry to encode vertical stripe patterns, but they also used distinguishable color stripes to support the correspondence-matching process. The use of color makes the pattern easily affected by surface color and light conditions. In addition to stripe patterns, 2D grid patterns have been investigated. Hu and Stockman [33] suggested using a grid pattern and showed that the problem can be simplified greatly. The observed grid in the camera is matched to the projected pattern by exploiting a number of geometric and topological constraints. Proesmans *et al.* [29] also used a grid pattern with a belief-propagation method to reduce noise. However, these methods impose the conditions that the object surface must be continuous and that most of the pattern must be recovered successfully for pattern matching to be archived.

Recently, Furukawa *et al.* [7] proposed using a grid pattern comprising horizontal and vertical colored lines. The constraint of coplanarity of the horizontal and vertical projection planes was used to solve for the correspondence. However, the process of performing a singular-value decomposition on a large and very sparse matrix is very time consuming. Similarly to Furukawa *et al.*, Ulusoy *et al.* [1] also used a grid pattern with horizontal and vertical colored lines. To resolve the correspondence problem, Ulusoy *et al.* used an epipolar-geometry approach, which resulted in a shorter calculation time. Ulusoy *et al.* also proposed using De Bruijn spacing patterns instead of the random spacing patterns proposed by Furukawa *et al.* for more robust evaluation results. By using a nonuniform spacing pattern [1], [7], the density of the pattern will be decreased, and the detail of the surface geometry will not be well constructed where the space between the grid lines is sparse.

3. Checkerboard Recognition

In this section, we propose an automatic checkerboard-recognition technique that is robust with respect to dynamic conditions such as illumination, color, surface distortion, and discontinuity. The recognition process analyzes the checkerboard pattern as a mesh of feature points connected by feature edges. In this process, no threshold value is required for crossing-point detection. The thresholds for identifying the feature edges' continuity are calculated adaptively and dynamically based on connected feature points. This automatic checkerboard-recognition technique enables a handheld 3D acquisition device to be used in dynamic conditions.

We use three steps to detect the checkerboard pattern. First, crossing feature points are detected and refined to sub-pixel accuracy. In the second step, the relatively vertical and horizontal feature edges are detected by extending the neighboring pixels of the detected feature points in the directionally filtered edge images. After this processing, noise is inevitable. In the third step, we propose a priority-based propagation method to label the feature points, and then apply topological constraints to eliminate noise at the graph level. Finally, feature points and feature edges are labeled by their relative position to prepare for the checkerboard-matching process.

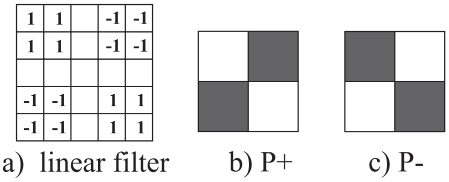
3.1 Feature Point Detection

The feature-point-detection process is essential to the robustness of the projection pattern because it is used in the search for corresponding positions in the checkerboard pattern. In the first step, the regions of each crossing point are approximately detected using the method proposed by Sun *et al.* [6]. This method detects a checkerboard crossing-point by using a descriptor, generated by linear extension of its neighboring points. The intensity of the neighboring pixels in a circle around a crossing point will vary from low to high and high to low twice. This method detects all points that have four alternating black and white areas surrounding them. Consequently, it requires no manual setting of thresholds related to light intensity. To increase the positional accuracy of the feature points and to eliminate wrongly detected crossing points at the pixel level, adjacent points are grouped and the crossing-point position is calculated by taking the mass center of all points in the same group.

Next, we apply the radial saddle-point approximation algorithm to refine the detected crossing points to sub-pixel accuracy. We use the method described in the OpenCV library [23] to find the sub-pixel radial saddle point iteratively. This method does not require the radial saddle point to be inside the initial pixel position, and it gives good convergence after several iterations. Finally, we use a linear filter, as shown in Fig. 2(a), to categorize the feature points into two categories, namely P+ and P–, depending on their directional characteristics (Figs. 2 (b) and 2 (c)). A feature point *P* is classified into the P+ category if its convolution integral with the linear filter results in a *plus* response, and classified into the P– category if its convolution integral with the linear filter results in a *minus* response. The linear filter in Fig. 2 (a) work correctly when categorizing feature points that are relatively aligned in the horizontal or vertical directions. By aligning the posture of the projector and the camera such that the projection area appears to be relatively parallel to the camera images, we can make sure that the feature points are always classified properly.

3.2 Feature Edge Detection

Our aim is to extract automatically the step edges that are the boundaries of the black and white areas in the checkerboard pattern with high accuracy. We present two techniques for eliminating unrelated feature edges automatically



1	1		-1	-1
1	1		-1	-1
-1	-1		1	1
-1	-1		1	1

Fig. 2 Categories for feature points: a) linear filter for the feature-point categorization, b) P+, and c) P–.

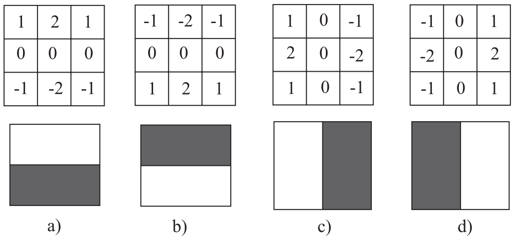


Fig. 3 Edge filters and corresponding categories for feature edges: a) H–, b) H+, c) V–, and d) V+.

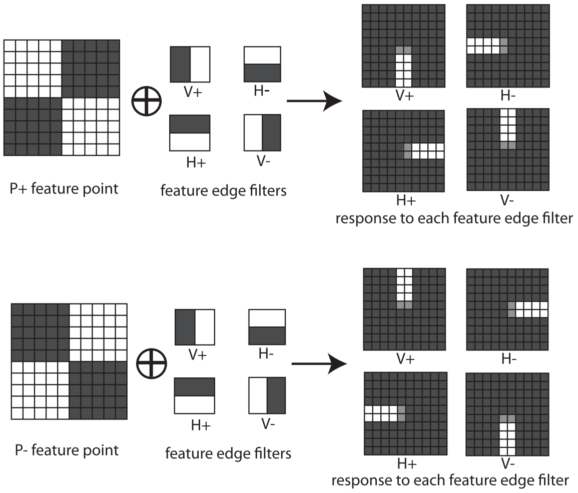


Fig. 4 Feature edges and their relative positions to feature points.

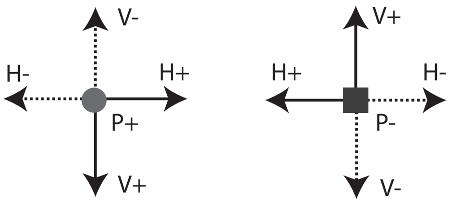


Fig. 5 Neighboring relations between the feature edges and the feature points.

and for changing the edge-detection threshold adaptively, so that separated edges are detected separately and each edge is detected continuously in a dynamic environment.

To limit the detected edges to only those belonging to the checkerboard pattern, we note that there are exactly four edges connected to each feature point, so that we can detect feature edges based on their connection with the detected feature points. The feature edges are distorted by the geometric projection, but their directions will remain relatively aligned to the horizontal or vertical axis when epipolar lines are relatively aligned to the vertical or horizontal axis. To detect the feature edges, we first use 3 × 3 Sobel filters (Fig. 3) to differentiate the image in four different directions. Based on the direction of the filters, the edges in the images are divided into four different categories, namely H+, H–, V+, and V–.

The feature edges must be at positions where the convolution integral between the filters and the image is locally maximal. However, because of the symmetry property, the values given by a convolution integral of the original image with edge filters will be zero or very small at the feature-point positions (Fig. 4). We can search for feature-edge to feature-point connections by looking for pixels in the neighborhood of the feature points. From a feature point, at a distance d equal to half of the size of the edge filter, the convolution integral values are not affected by the symmetric geometry of the feature point, and its maximal value should be the starting point of an edge. In the case of the 3 × 3 Sobel filters, a distance of two pixels is sufficient to decide the connectivity between the feature points and the feature edges.

Consider a feature point $P(x_p, y_p)$ and a feature edge E that connects to the feature point P at one end $E_1(x_1, y_1)$. Their relative positions will satisfy the conditions of Eqs. (1) and (2):

$$E \in H\pm \Rightarrow \begin{cases} x_1 = x_p + \sigma d \\ y_p - 1 \le y_1 \le y_p + 1 \end{cases} \tag{1}$$

$$E \in V\pm \Rightarrow \begin{cases} x_p - 1 \le x_1 \le x_p + 1 \\ y_1 = y_p + \sigma d \end{cases} \tag{2}$$

Here, parameter σ is the direction relation between the feature point and the feature edge, where σ equals 1 if the feature point and the feature edge are of the *same sign* and σ equals -1 if the feature point and the feature edge are of *different signs*. Their neighboring relations are expressed in Fig. 5. The edge's starting point E_1 is decided by taking the pixel with the maximum response (Eq. (3)) in the range expressed by Eqs. (1) and (2).

$$(\hat{x}_1, \hat{y}_1) \equiv arg \max_{x_1, y_1} I(x_1, y_1) \tag{3}$$

where $I(x, y)$ is a convolution integral value between an original image and appropriate edge filters. The edge-detection process is continued by extending the detected pixel $E_n(x_n, y_n)$ in an edge E in the direction specified by σ in Eqs. (4) and (5).

$$E \in H\pm \Rightarrow \begin{cases} x_{n+1} = x_n + \sigma \\ y_n - 1 \le y_{n+1} \le y_n + 1 \end{cases} \tag{4}$$

$$E \in V\pm \Rightarrow \begin{cases} x_n - 1 \le x_{n+1} \le x_n + 1 \\ y_{n+1} = y_n + \sigma \end{cases} \tag{5}$$

By extending to the adjacent pixels with maximum response in the filtered image in the corresponding direction for each feature edge, we obtain the feature edges connecting to each feature point (Eq. (6)).

$$(\hat{x}_{n+1}, \hat{y}_{n+1}) \equiv arg \max_{x_{n+1}, y_{n+1}} I(x_{n+1}, y_{n+1}) \tag{6}$$

with the condition of Eq. (7):

$$I(x_{n+1}, y_{n+1}) \ge \alpha I(x_1, y_1) \tag{7}$$

Here, α is used to decide the continuity of the detected feature edge. The edge-extension process will finish when the response of the next pixel is smaller than the threshold, or the next pixel has been labeled with another edge. An edge connecting two feature points will have two different continuity thresholds at its ends. In this case, the edges may be extended from different starting points and then merged into one if they belong to the same feature point category (e.g. P+ or P–). In our implementation, the threshold for edge continuity is taken to be half the value at its starting point ($\alpha = 0.5$). From our experiments, this value is adequate for edge extraction under most conditions.

Finally, the pixel-scale edges are refined to sub-pixel accuracy. In this research, we adopt sub-pixel edge estimation based on a quadratic approximation to find local derivative maxima in the filtered image. In our implementation, this method is adequate, calculating sub-pixel edges rapidly and robustly. However, many sub-pixel edge-detection techniques have been proposed, and they could be used to achieve possibly better results.

3.3 Noise Filtering and Labeling Scheme

During the detection processes for feature points and feature edges, various types of error may occur, resulting from surface mosaic colors, occlusion, or discontinuity. For the feature points, errors involve false detection and misdetection. For the feature edges, errors involve disconnection and false detection. Misdetected feature points and discontinuous feature edges will result in holes in the reconstructed geometry, but do not affect the accuracy of the correspondence-searching process. On the other hand, falsely detected feature points and feature edges will result in inaccurate geometry.

An edge can connect with at most two feature points at a time at its ends, so any feature points that connect with the feature edge in the middle are considered noise. This kind of noise may occur as a result of the adaptive feature-edge detection process, and can be filtered out by these constraints. The average length of the edges that connect with two different feature points is also used to limit the proper length of the edges that connect with only one feature point. However, there may still be errors caused by occlusion or discontinuity on the object's surface, and the error may spread during the labeling process.

For our implementation, we propose using topological constraints to eliminate noise during the labeling process. This approach is not new, having been used to recognize a checkerboard pattern by Chang *et al.* [17] and by Proesmans *et al.* [29] to eliminate noise in a grid pattern. However, by categorizing the feature points and the feature edges, we can filter out the noise more efficiently.

In Fig. 6, the labeling process that starts from a random origin point $(0, 0)$ may result in inconsistency at points $(3, 1)$, $(3, 2)$, and $(3, 3)$. These errors can be detected by checking the consistency of the loops in the groups. However, in the example shown in Fig. 6, it is not obvious where the error is. The feature points at positions $(1, 0)$ and $(2, 0)$ may be falsely detected, or the edges between feature points $(0, 1)$–$(3, 1)$, $(0, 2)$–$(3, 2)$, and $(0, 3)$–$(3, 3)$ may be in error because of edge collapse. Edge collapse is rare and falsely detected feature points are more common. This is because falsely detected feature points can be noise or any crossing corner in the image itself that does not belong to the checkerboard pattern. On the other hand, the edge collapse only occurs on some specific mosaic patterns or partly occluded surfaces. We solve the problem of the inconsistency by setting a priority for the detected feature points based on their number of neighbors, and then start the labeling process with points of high priority, moving towards points of lower priority (Fig. 7). This method takes the precedence of eliminating noise corners over false edges, and therefore may result in false 3D reconstruction results. This issue is discussed further in Sect. 5.3.

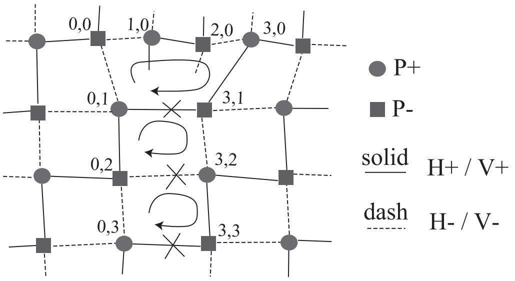


Fig. 6 Inconsistent labeling.

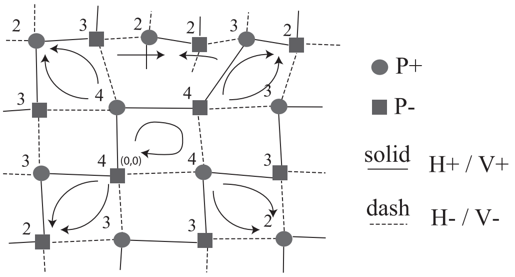


Fig. 7 Propagation based on the feature point's priority, where the number on each feature point is its number of neighboring points.

The origin for the labeling process is taken to be at the middle of the points of highest priority, which is labeled $(0, 0)$. The neighboring points of the labeled point are then searched and put into a queue in order of their priority, and the labeling process continues from the point of the highest priority in the queue. Feature points of lower priority can then be labeled in more than one way. If such a point (of low priority) is labeled inconsistently, it is interpreted as noise.

4. Checkerboard Matching

To resolve a correspondence problem, we use the epipolar geometry of the projector–camera system. Epipolar geometry has been used to solve the correspondence problem in computer vision for a long time ago. However, it is difficult to solve the correspondence for a dense pattern when the crossing points are detected at low accuracy. In this research, we propose a method of using the epipolar geometry of groups of feature points to solve for the correspondence. We also prove the uniqueness of the correspondence result for the group of feature points if the group is sufficiently large.

4.1 Epipolar Geometry and Correspondence Matching

In a rigidly attached and calibrated projector–camera system, a point $P_i(x_i, y_i, 1)$ in the projection pattern is mapped to a point $P'_i(x'_i, y'_i, 1)$ in a camera image (Fig. 8). For the calibrated camera, when the distance from the projector–camera system to the screen changes, the trajectory of the mapped point P'_i is an epipolar line $l'_i(x'_i, y'_i, z'_i)$. The epipolar line l'_i in the camera's image is calculated from the position $P_i(x_i, y_i, 1)$ in a projection pattern by the equation:

$$l'_i = FP_i \tag{8}$$

and conversely:

$$l_i = F^T P'_i \tag{9}$$

Here, F is a fundamental matrix of the rigidly attached projector–camera system and it is calculated by the calibration process. The ambiguity is that not only point P_i but also all points lying on the corresponding epipolar line l_i in the projection pattern will map to the same epipolar line l'_i in the camera's image (Fig. 8). This ambiguity is inevitable when the checkerboard pattern is dense.

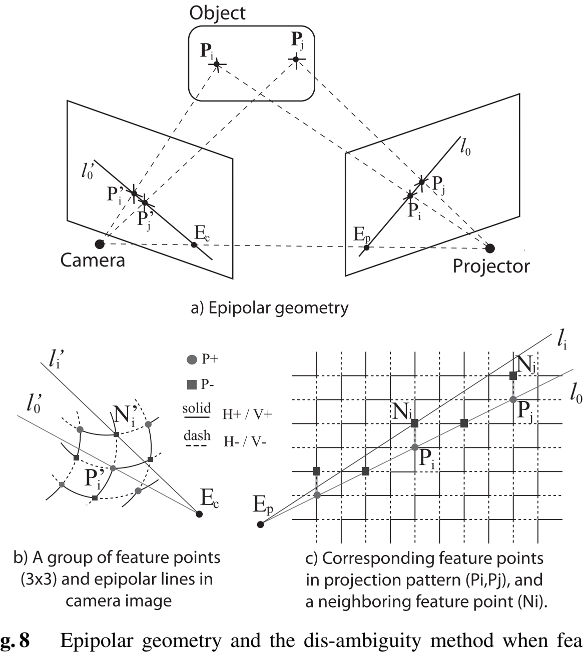
In a checkerboard pattern, this ambiguity will be resolved if there is a connection clue between at least two feature points in the camera image of different epipolar lines. Consider the case of Fig. 9, where multiple points $\{P_1, P_2, \ldots, P_n\}$ of the same feature point category on the projection pattern lie on an epipolar line l_0. Under ideal conditions, this means that the Euclidean distance from those points to the epipolar line l_0 should be 0 (Eq. (10)).

$$d(P_1, l_0) = d(P_2, l_0) = \cdots = d(P_n, l_0) = 0 \tag{10}$$

The corresponding feature points $\{P'_1, P'_2, \ldots, P'_n\}$ in the camera image will lie on the corresponding epipolar line l'_0, meaning that their correspondence will not be distinguishable by using individual epipolar lines. However, the ambiguity is resolved if their neighboring points on different epipolar lines are identified. If a feature point P'_i in the camera image has a neighboring feature point N'_i on a different epipolar line l'_i, then its corresponding epipolar line l_i in the projector image is defined by using Eq. (9). Because of the grid structure of the checkerboard pattern, the line that goes though the feature points $\{P_1, P_2, \ldots, P_n\}$ and the line that goes through their neighboring feature points $\{N_1, N_2, \ldots, N_n\}$ will be parallel together as shown in Fig. 8 (c). As the feature points $\{P_1, P_2, \ldots, P_n\}$ are on the epipolar line l_0, then the epipolar lines going through their neighboring points $\{N_1, N_2, \ldots, N_n\}$ will be separated, if the epipole on the projector image e_p is not at infinity. By taking the neighboring point N_i is the point with the smallest distance to the corresponding epipolar line l_i (Eq. (11)) as the corresponding feature point of N'_i (Fig. 9 (c)), we can consequently resolve the ambiguity for the feature point P_i via its connection to the feature point N_i:

$$\hat{j} \equiv arg \min_{0 \le j \le n} d(N_j, l_i) \tag{11}$$

In general, because of noise or other external factors, the condition in Eq. (10) does not always hold. In a dense projection pattern, when the number of feature points increases, the resolution angle (and the distance) between

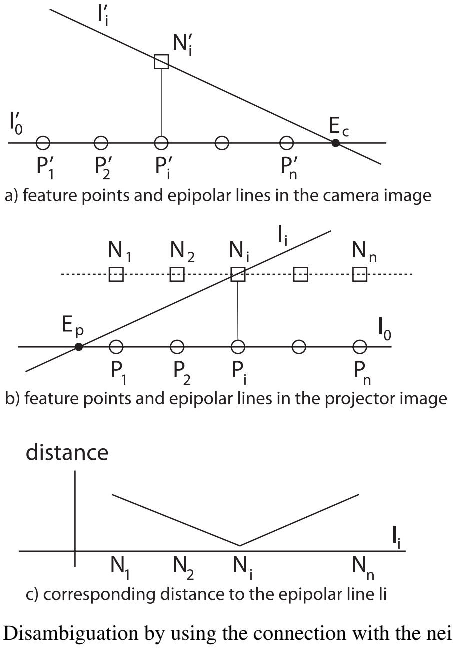


a) Epipolar geometry

b) A group of feature points (3x3) and epipolar lines in camera image

c) Corresponding feature points in projection pattern (Pi,Pj), and a neighboring feature point (Ni).

Fig. 8 Epipolar geometry and the dis-ambiguity method when feature points lie on the same epipolar line.



a) feature points and epipolar lines in the camera image

b) feature points and epipolar lines in the projector image

c) corresponding distance to the epipolar line li

Fig. 9 Disambiguation by using the connection with the neighboring feature points.

corresponding epipolar lines becomes very small. Therefore, an epipolar line is considered to go through a feature point if its distance to the feature point is minimum or smaller than a tolerance value ϵ. To increase the accuracy of the correspondence-matching process, we can use all feature points in a group to find the corresponding position for that group. The distance to corresponding epipolar line (Eq. (11)) will reach the minimal value when it comes to the true correspondence. We can therefore sum the corresponding distances for all points in the same group to get a better-discriminated evaluation result based on the following procedure.

For the case of a group of $m+1$ connected feature points $\Omega'_t = \{P'_{t0}, P'_{t1}, \ldots, P'_{tm}\}$ (detected as a part of the checkerboard pattern) in the camera image, if the corresponding position of one feature point in the group is defined, the corresponding positions of other points in the group are consequently defined as discussed above. Without loss of generality, instead of finding the corresponding positions for all points in the group, we can find the corresponding position for a feature point P'_{t0} of the group Ω'_t. In our implementation, we use the feature point at the origin position $(0, 0)$ as shown in Fig. 7 to find the correspondence. We call the collection of candidate corresponding positions of feature point P'_{t0} is $\Phi_t = \{P^0_t, P^1_t, \ldots, P^n_t\}$, where $n+1$ is the number of candidate corresponding positions. These candidate corresponding positions are chosen by taking the crossing corners that have a distance to the corresponding epipolar lines l_{t0} of the feature point P'_{t0} smaller than the threshold ϵ (Eq. (12)).

$$d(P^i_t, l_{t0}) < \epsilon \tag{12}$$

The candidate corresponding points must be in the same category as the selected feature point P'_{t0} (Fig. 8 (c)). Assuming that, the corresponding point of the feature point P'_{t0} is $P^i_{t0} \in \Phi_t$, then the corresponding points for other feature points in the group Ω'_t are consequently $\Omega^i_t = \{P^i_{t0}, P^i_{t1}, P^i_{t2}, \ldots, P^i_{tm}\}$ (Fig. 10). The temporary corresponding positions P^i_{tj} ($j = 0, \ldots, m$) are derived from P^i_{t0}.

The sum of the distance D_{ti} of all temporary corresponding feature points P^i_{tj} in the group Ω^i_t to their corresponding epipolar lines is calculated by Eq. (13):

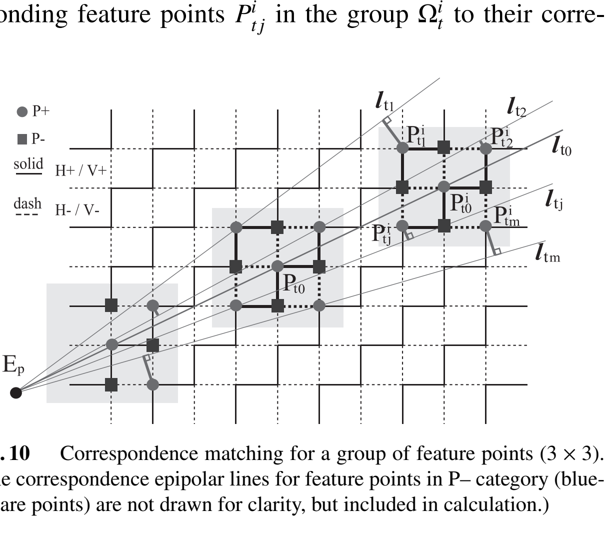


Fig. 10 Correspondence matching for a group of feature points (3 × 3). (The correspondence epipolar lines for feature points in P– category (blue-square points) are not drawn for clarity, but included in calculation.)

$$D_{ti} = D_t(P^i_{t0}) = \sum_{j=0}^{m} d(P^i_{tj}, l_{tj}) \tag{13}$$

where l_{tj} is the epipolar line corresponding to the feature point P'_{tj} in the group Ω'_t. l_{tj} is calculted from P'_{tj} via Eq. (9), and does not change for different candidate corresponding position Ω^i_t (Fig. 10). The sum of the distances D_{ti} indicates the error for different corresponding positions Ω^i_t for the group Ω'_t. We can therefore use D_{ti} as the evaluation function to find the corresponding position of Ω'_t. The average distance of each feature point to its corresponding epipolar line can be calculated by taking the average value of D_{ti} (Eq. (14)):

$$d_{ti} = D_{ti}/m \tag{14}$$

The corresponding position P_{t0} for the feature point P'_{t0} is taken to be the point at $\hat{i}$ that minimizes the sum of distances D_{ti} or the average distance d_{ti} (Eq. (15)):

$$\hat{i} \equiv arg \min_{0 \le i \le n} D_t(P^i_{t0}) \equiv arg \min_{0 \le i \le n} d_{ti} \tag{15}$$

When the corresponding epipolar line of a feature point in a group is determined based on values obtained through (Eq. (15)), corresponding epipolar lines of other feature points in the groups are determined accordingly.

4.2 Conditions for Correspondence Matching

Epipolar geometry and the grouping of feature points can be utilized to resolve ambiguity, as described in Sect. 4.1. The accuracy of the matching process depends on four main factors: the accuracy of the feature-point detection, the accuracy of calibration process, the size of the groups, and the projector-camera system settings. Detecting the crossing points with high accuracy in accordance with high calibration accuracy result in more reliable corresponding evaluation values; enabling us to solve the correspondence problem for a dense checkerboard pattern.

On other hand, a larger group of feature points results in a more discriminating evaluation value. The sizes of the groups depend mainly on the condition of the object's surface. We restrict the corresponding positions of groups so that all feature points with high priority in a group must be inside the checkerboard pattern, and that feature points with high priority in different groups do not overlap each other. These restrictions enable us to reduce the number of candidate corresponding positions, which increase the performance of the corresponding matching in its accuracy and speed. In case there are several groups, the correspondence matching process is carried out by the order from larger to smaller groups. For the checkerboard pattern, we also limit the corresponding feature points to those in the same category (Fig. 8). This means that the correspondence of feature points in category P+/P– must be crossing points in category P+/P– in the projection pattern.

Finally, the condition for the existence of a minimum value in Eq. (11) is that the epipoles in the camera image and the projector image are not at infinity. This means that the projector and the camera must not be aligned parallel to each other in the direction of their axis. On the contrary, if the epipoles are closer to the image centers, the distances between different feature points to the epipolar lines will be more discriminating.

5. Configuration and Experiment

5.1 Configuration

We have implemented a handheld projector–camera system to investigate the efficiency of our proposal for one-shot geometry acquisition in a dynamic environment. In our implementation, a Logicool camera (QCAM-200V 640×480) and a Mitsubishi projector (DLP-PK20 800×600) are rigidly attached as shown in Fig. 11. The camera works at 30 fps. We calibrated the projector–camera system by extending the method for camera calibration proposed by Zhang [22] to the projector, and by using a calibration rig to measure the relative positions of the projector and the camera via a computer-vision technique. After the calibration process, the projector's intrinsic parameters, the camera's intrinsic parameters, and the external parameters relating the projector to the camera will have invariant values.

In our implementation, the camera center is located at $(108.5, 12.7, 81.4)$ mm relative to the projector center, and the Y-axis rotation is 12°. Because the focal length of the projector is longer than that of the camera, the projection area appears in the camera image at a resolution of approximately 320×240 pixels.

The working range of the scanning system is from 30 cm to 100 cm. The region in the camera image where the projection pattern appears (the region of interest) is a rectangle of 500×300 pixels. Using a computer with an Intel Core 2 Duo 2.4 GHz CPU and 2 GB RAM, our implementation achieved a geometry-reconstruction speed of 12 to 13 fps for a full 30×20 checkerboard pattern and 5 to 6 fps for a full 40×30 checkerboard pattern.

5.2 Experiment

To compare the robustness of the proposed checkerboard-recognition method with existing techniques, we carried out experiments with the three sample images shown in Fig. 12. These images were created by displaying a relatively sparse 20×15 checkerboard pattern from the projector and capturing the images statically via the attached camera. The first image (Fig. 12 (a)) is the planar surface of a wall. The second image (Fig. 12 (b)) is a corner of multiple planar surfaces of different colors and illumination-reflection conditions, and the third image (Fig. 12 (c)) is a doll that has a curved surface with high-contrast mosaic areas.

We compared our checkerboard corner-detection method with two other methods, namely the method proposed by Rufli *et al.* [8] and the widely used Harris corner detector [34] because many checkerboard-detection techniques use a Harris corner detector for the checkerboard pattern-detection process [15], [17]. The method proposed by Rufli *et al.* was implemented to detect black rectangular areas, so we converted the sample images into negative black and white images, as shown in Fig. 13. Both the Harris corner-detection technique and the checkerboard-detection technique proposed by Rufli *et al.* require some thresholds in the detection process. We chose the best detection results for these techniques to compare with our automatic checkerboard-detection technique.

The precision and recall for each detection technique are summarized in Table 1. The results for corner detection based on the Harris filter are shown in Fig. 13 (a) with a Harris-recommended threshold ($k = 0.01$ in [34]) and various quality factors (q in [23]). We can decrease the quality factor (q) for a better detection rate (recall), but overall, the Harris corner detector produces much noise unrelated to the checkerboard corners. The method proposed by Rufli *et al.* (Fig. 15 (b)) can detect checkerboard corners with quite high precision (Table 1). However, it requires a relatively uniform areas for the rectangular areas to be detected.

Figure 13 (c) shows the corner detection results for different sample images by using the detector by Sun *et al.* before the noise filtering process. The larger recall (Table 1) proves that the method proposed by Sun *et al.* can detect the checkerboard corners more efficiently than that proposed by Rufli *et al.*, because the former method requires less space for detecting a corner. However, when the checkerboard pattern is displayed on a non-uniform surface many corners are

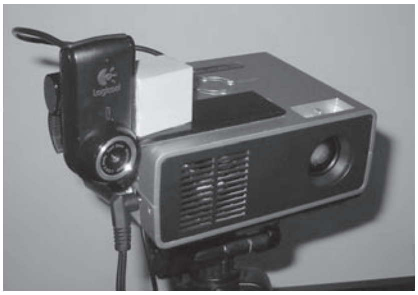


Fig. 11 The rigidly attached handheld projector-camera system.

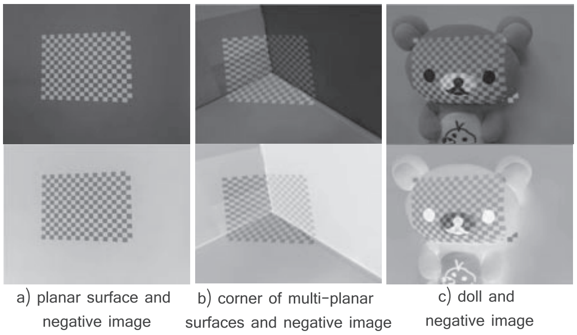
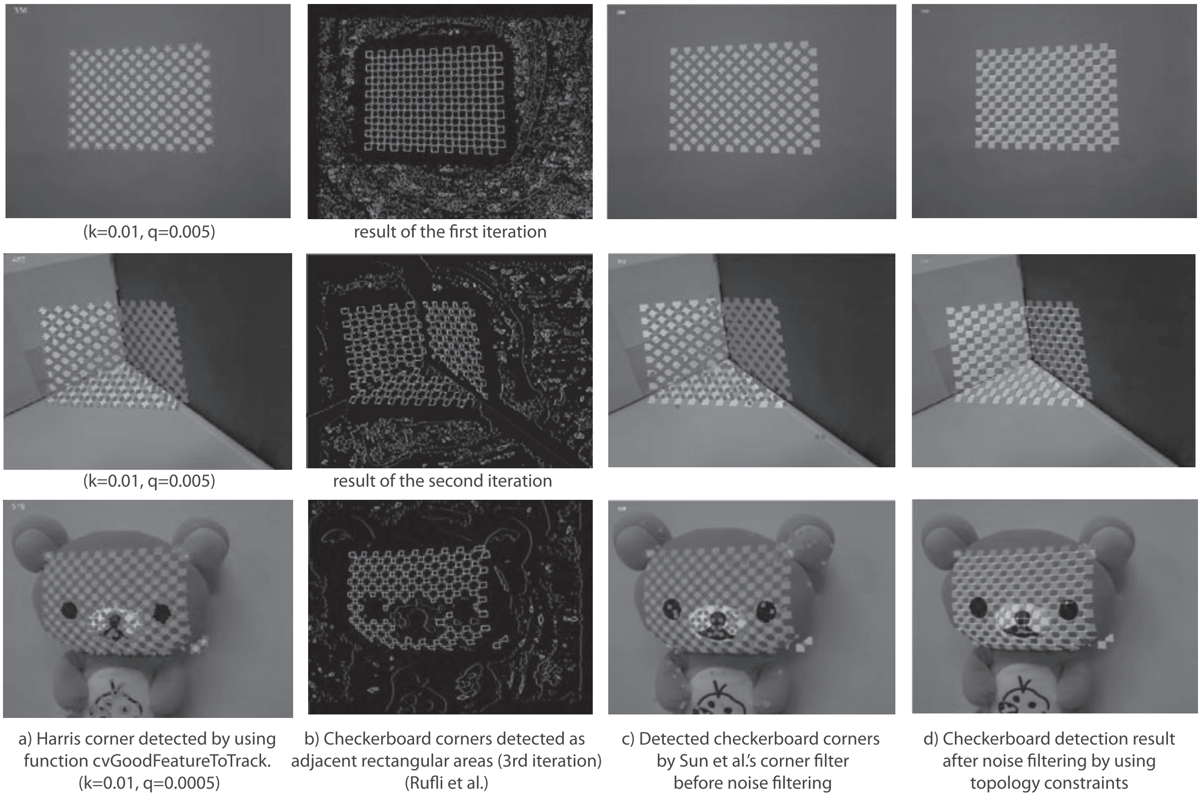

a) planar surface and negative image
b) corner of multi-planar surfaces and negative image
c) doll and negative image

Fig. 12 Sample images for experiment.


(k=0.01, q=0.005)

result of the first iteration

(k=0.01, q=0.005)

result of the second iteration

a) Harris corner detected by using function cvGoodFeatureToTrack. (k=0.01, q=0.0005)

b) Checkerboard corners detected as adjacent rectangular areas (3rd iteration) (Rufli et al.)

c) Detected checkerboard corners by Sun et al.'s corner filter before noise filtering

d) Checkerboard detection result after noise filtering by using topology constraints

Fig. 13 Checkerboard detection results for different sample images using different methods.

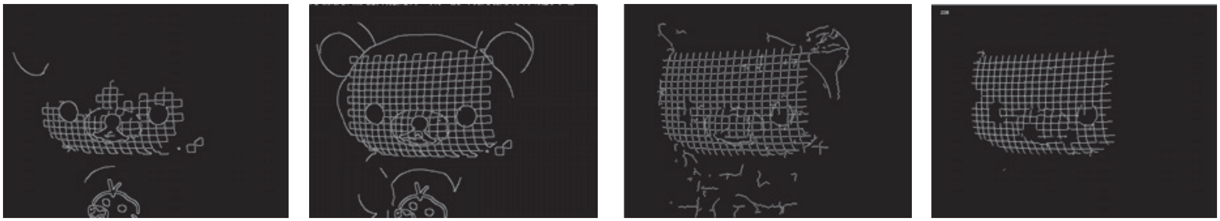
Table 1 Precision and recall of checkerboard detection techniques in term of the number of detected corners.

	Harris corner		Rufli et al.		Before noise filtering (Sun et al.)		After noise filtering	
Sample	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
Planar surface	79.17	100	88.67	100	100	100	100	100
Corner (Multi-planar)	50.65	87.97	94.80	82.33	81.09	95.11	99.20	92.86
Doll (Curved surface)	40.66	81.15	98.32	67.69	67.07	84.6	99.09	83.46

falsely detected, and therefore the detection precision deteriorates without the noise filtering. However, after the noise filtering process by using topology constraints (Fig. 13 (d)), the precision of the detected corners is improved remarkably (Table 1); The precision of the method with topology constraints is close to 100% for all the sample images while its recall is still better than that of the method proposed by Rufli *et al.*

In our algorithm, the accuracy of the detected feature edges depends directly on the accuracy of the feature points, because the edges are detected by adaptively expanding the detected feature points in different directions. The high accuracy of feature-point detection also means that the feature edges are detected with high accuracy. We compared the results for the edge-detection process with the widely used Canny edge-detection technique. The results for an example image are shown in Fig. 14. We can confirm that the Canny filter may not detect all the edges of the checkerboard pattern and may include many unrelated edges using the selected thresholds. On the other hand, our algorithm can detect the inner edges of the checkerboard pattern with high accuracy. The edges are detected separately, enabling them to be analyzed easily. It is worth noticing that the edge-detection process is self-adaptive and automatic, making it robust with respect to different light conditions and color changes.

To verify the accuracy of the proposed method, we measured a blank planar surface from different distances by using a 40 × 30 checkerboard pattern, and calculated the depth distances errors between each of the 40 × 30 points on the surface and a reference point on the projector. Figure 15 shows the averaged distance errors and standard deviations among these individual points. Each distance on the horizontal axis of the figure means the distance between the reference point in the projector-camera system and the planar surface. The measured depth distance and standard deviations of the individual points were calculated by using 200 sample images. Figure 15 proves that the proposed method could conduct stable distance measurements within the 100 cm distance; for example, the standard deviation at 80 cm was less than 4.5 mm. The standard deviation increases in proportion to the distance. However, the mea-



a) Canny edge [100,200]

b) Canny edge [50,100]

c) Edges detected by adaptively expanding from feature points

d) Detected edges after noise filtering

Fig. 14 Comparison between edge detection techniques: a, b) edges detected using a Canny filter, c) edges detected by adaptively expanding from feature points, and d) edges detected after noise filtering using topology constraints.

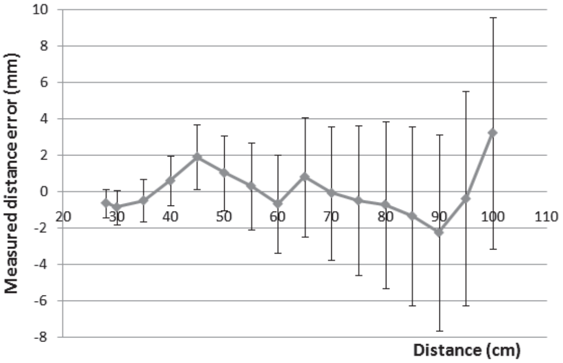


Fig. 15 Measured distance errors to a planar surface. The error bar is the standard deviation of measurement for each feature points.

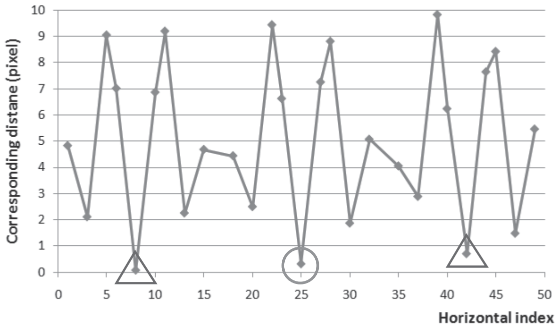


Fig. 16 Distances of candidate crossing points to an epipolar line and their ambiguity (the blue-circle means the true correspondence and the red-triangles mean the wrong correspondences due to noise).

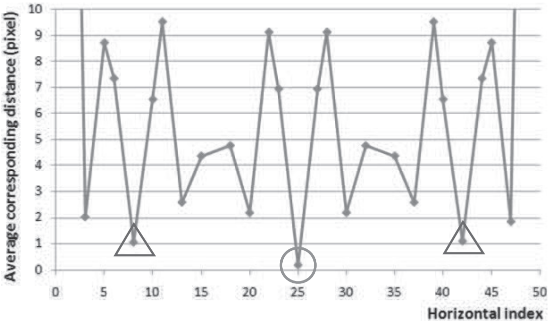


Fig. 17 Average of corresponding distance for groups of 5 × 5 feature points (influence of noise is suppressed).

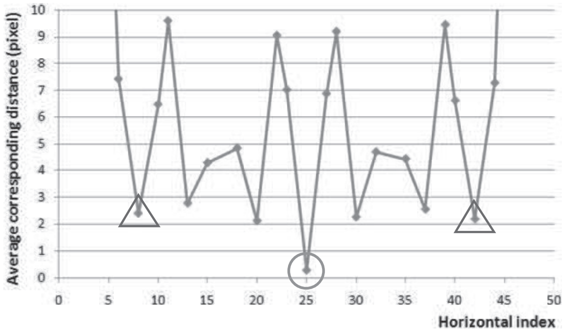


Fig. 18 Average of corresponding distances for groups of 10×10 feature points (influence of noise is more clearly suppressed).

sured distance error does not show the same tendency. This may be because of human errors when measuring the distance from the reference point to the planar surface.

To verify the convergence and correctness of the evaluation function (Eq. (13)) in the correspondence-matching process, we measured the value of the evaluation function for different group sizes at the center of a 50 × 40 checkerboard pattern. Figure 16 shows the distances of crossing points in the checkerboard pattern to the corresponding epipolar line of a single feature point. The candidate corresponding positions when the threshold value ϵ (Eq. (12)) equals 1 pixel are marked in circle and triangle areas. The true corresponding point is circled, but the ambiguity (points in triangles) is not resolvable by using only epipolar geometry. Figures 17 and 18 show the average corresponding distances (Eq. (14)) for all points in groups of 25 (5 × 5) and 100 (10 × 10) feature points. By measuring the correspondence for a group of connected feature points, the ambiguity about each feature point can be resolved as have discussed in Sect. 4.1. The sum of corresponding distances for larger groups discriminates better and converges faster, and is therefore more robust for the correspondence-matching process.

Several experiments were conducted with deformable and rigid real objects to investigate the robustness of the proposed method. The surface of the doll (Rirakuma) can be scanned by a 40 × 30 checkerboard pattern as shown in Fig. 19. However, there remain holes in the areas of the eyes

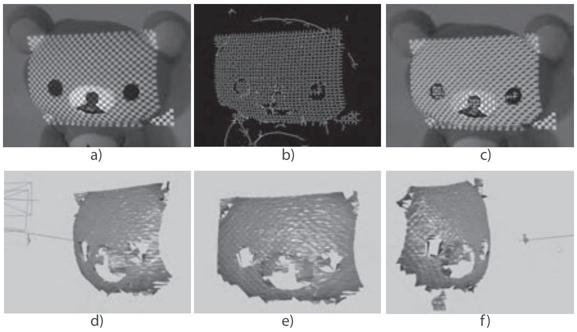


Fig. 19 Scanning a doll: a) doll under 40 × 30 projection pattern, b) extracted edges, c) recognized checkerboard pattern and d, e, f) different view of doll geometry.

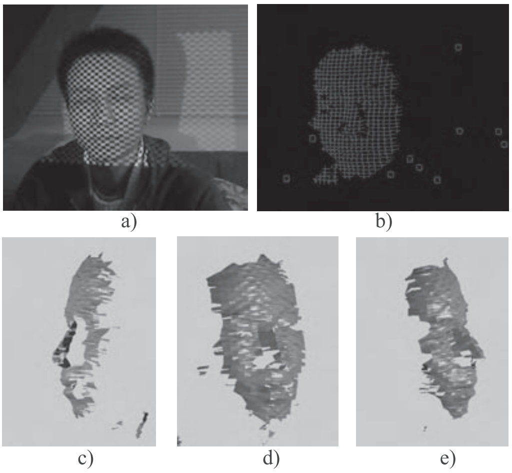


Fig. 20 Scanning a human face: a) human face under projection pattern, b) recognition results indexed by color, and c, d, e) different views of face geometry.

and mouth of the doll. The holes are partly because checkerboard detection has failed in those areas, and also because of limitations in the triangulation-creation process. In future work, we will seek a better method for triangulation to solve this problem. Similarly a human face can also be scanned with the same pattern, as shown in Fig. 20.

Although our current setup focuses on the mobility and the robustness of the system under dynamic conditions, and only a low-resolution Web camera is being used, the system can capture the geometry of a complex object such as a human hand (Fig. 21). The capture results for a colorful and high-contrast object such as the puppet shown in Fig. 22 (Kyoro-chan) and the surfaces with large discontinuity as shown in Fig. 23 may be evidence supporting the robustness of the detection technique.

5.3 Discussion about the Limitations of the Proposed Method

Currently, the proposed technique has several limitations mainly in the image processing process. They are the density of the checkerboard pattern, the surface distortion, the limited size of mosaic areas and the surface occlusion that results in edge collapses. The checkerboard recognition process requires several layers of neighboring pixels to identify a crossing corner. The resolution of the checkerboard pattern is limited by the minimum number of the layers to recognize a crossing point. Therefore this technique cannot ac-

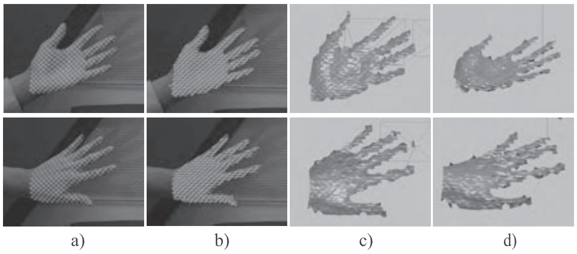


Fig. 21 Scanning an object with complex geometry: a) a hand under a 40 × 30 projection pattern, b) recognized checkerboard pattern, and c, d) hand geometry from different viewpoints.

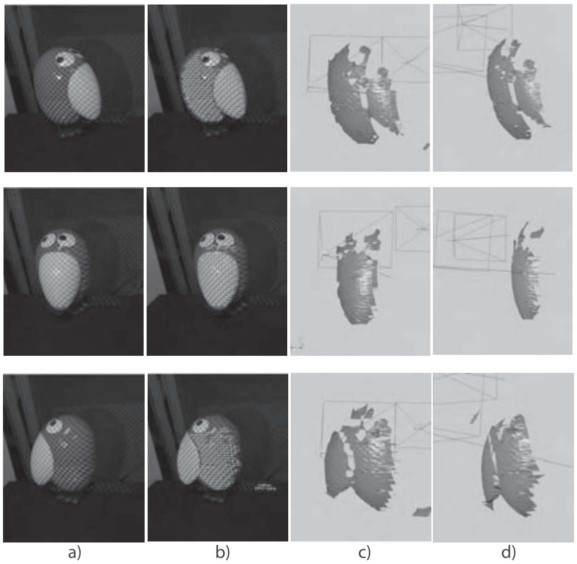


Fig. 22 High-contrast colorized object (Kyoro-chan): a) different views of the object, b) checkerboard recognition results, and c, d) reconstructed geometry in different views.

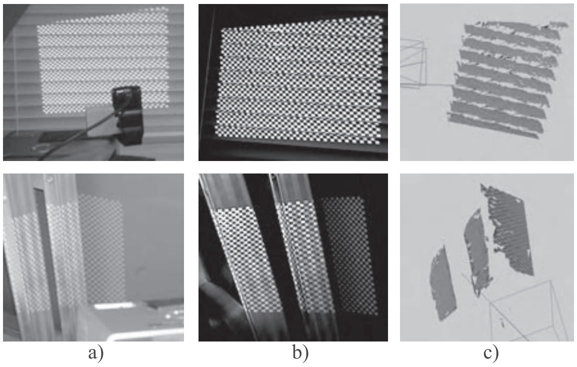


Fig. 23 Reconstruction of geometry of discontinuous surfaces: a) projected patterns, b) camera images, and c) reconstructed geometry.

quire the depth-map of an object for each pixel. Our current technique can reduce the number of pixels between crossing corners to three, if only two layers are used for the checkerboard recognition. However, an extremely dense checkerboard pattern is too sensitive to noise. Moreover, when a measured surface is distorted, the distance between crossing points may become smaller in the camera image. A sparse checkerboard pattern is therefore more robust to surface distortion, but it trades off against the resolution of acquisition data. In our current implementation a web-camera attached on a light weight projector is used. The area of the projected checkerboard pattern in the camera image is 320 × 240 pixels (QVGA), and the averaged space between the corners is 5 or 6 pixels. One method for increasing the density of the pattern to be successfully recognized is to use a high resolution camera. For example, if a camera whose resolution is 1600 × 1200 pixels is used, the density of the checkerboard pattern can theoretically be improved to 25 times as high as the current density.

Finally, in the labeling process (Sec. 3.3), to keep the consistency, we decided to ignore the collapsed edges and removed inconsistent feature points. However, this assumption is not always true. The edge collapse happens due to a surface mosaic pattern or occlusions as shown in Fig. 24 and can produce a failure 3D reconstruction result as shown in Fig. 25. The collapsed edges may wrongly connect different parts of the checkerboard pattern, and cannot be detected by using topological constraints. In Fig. 25, an edge collapse is detected through the feature points to which the corresponding matching is not applicable (circled in red). Furthermore, we can also detect the edge collapse based on the value calculated by using the evaluation function. Wrong connections of feature points result in wrong corresponding epipolar lines. For example, a detected feature point P, which is on an epipolar line l_P, is corresponding to an epipolar line $l_{P'}$. If the wrong connection is used, a large error in the evaluation function is obtained (Eq. (13)). By using a threshold for the distance of the corresponding positions to the corresponding epipolar lines, or using a threshold for the evaluation function, we can detect the false correspondence matching result caused by the edge collapse.

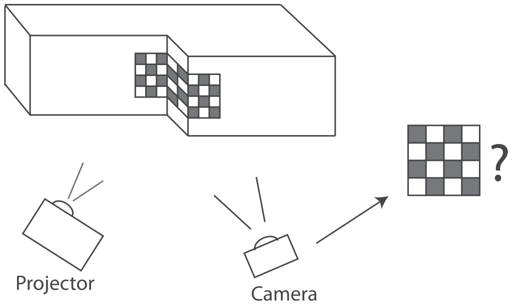


Fig. 24 Edge collapse caused by occlusion.

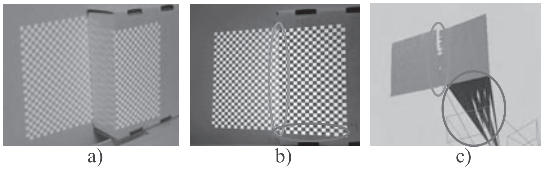


Fig. 25 A false reconstruction result caused by edge collapse: a) projected checkerboard pattern, b) captured checkerboard pattern with edge collapse and c) wrong reconstruction result. (Vertical eclipses show edge occlusion areas and horizontal eclipses show areas where the corresponding matching failed.)

6. Conclusion

In this paper, we have proposed a one-shot 3D acquisition technique that uses a checkerboard pattern for structured light projection. The checkerboard pattern does not require any special position-encoding technique used by many existing studies. We have shown that, by using the local connectivity of simply categorized feature points and feature edges, the pattern can be analyzed robustly and is minimally affected by surface colors, discontinuity, or occlusion. The recognition technique is adaptive, and no manual setting of threshold values is required during the detection process, making this technique particularly suited to geometry acquisition in dynamic environments. Our checkerboard-recognition technique allows the recognition of dense and distorted patterns, which is particularly desirable for geometry-reconstruction systems. We have implemented a handheld scanning system by using a mobile projector and a Web camera to measure the geometry of a nearby object, and users can check the reconstructed geometry in real time.

In future work, we aim to merge 3D geometric data to create a 3D model in real time. It is an advantage of the handheld system that it can easily capture the geometry of an object from different angles. By merging the captured data, we can increase the accuracy of the model. We also plan to make the pattern adaptive to the environment by dynamically changing its horizontal and vertical line density. We believe that an adaptive pattern will produce a better 3D-scanned result in a dynamically changing environment.

References

[1] A.O. Ulusoy, F. Calakli, and G. Taubin, “One-shot scanning using De Bruijn spaced grids,” Proc. IEEE/3-D Digital Imaging and Modeling, 3DIM Workshop, pp.1–8, 2009.

[2] R. Sagawa, R. Furukawa, H. Kawasaki, Y. Ohta, Y. Yagi, and N. Asada, “Dense 3D reconstruction method using a single pattern for fast moving object,” Twelfth IEEE International Conference on Conputer Vison (ICCV 2009), 2009.

[3] V.N. Dao and M. Sugimoto, “A robust and dynamic scene geometry acquisition technique for a mobile projector-camera system,” SIGGRAPH ASIA ’09: ACM SIGGRAPH ASIA 2009 Sketches, pp.1–1, 2009.

[4] P. Mistry, P. Maes, and L. Chang, “Wuw — Wear ur world: A wearable gestural interface,” CHI EA ’09: Proc. 27th International Conference Extended Abstracts on Human Factors in Computing Systems, pp.4111–4116, 2009.

[5] K. Hosoi, V.N. Dao, A. Mori, and M. Sugimoto, “Mobile display-based manipulation: An intuitive and interactive technique for ma-

nipulating simple robots," Proc. ACM ACE 2008, pp.271–274, Yokohama, Japan, 2008.

[6] W. Sun, X. Yang, S. Xiao, and W. Hu, "Robust checker-board recognition for efficient nonplanar geometry registration in projector-camera systems," Proc. 5th ACM/IEEE International Workshop on Projector Camera Systems (PROCAMS), pp.1–7, 2008.

[7] R. Furukawa, H.Q.H. Viet, H. Kawasaki, R. Sagawa, and Y. Yagi, "One-shot range scanner using coplanarity constraints," Proc. IEEE Int'l. Conf. on Image Processing 2008, pp.1524–1527, Oct. 2008.

[8] M. Rufli, D. Scaramuzza, and R. Siegwart, "Automatic detection of checkerboards on blurred and distorted images," Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems. [IROS 2008], 2008.

[9] M. Tateishi, H. Ishiyama, and K. Umeda, "A 200 Hz small range image sensor using a multi-spot laser projector," Proc. IEEE Int. Conf. on Robotics and Automation, pp.3022–3027, 2008.

[10] Y. Watanabe, T. Komuro, and M. Ishikawa, "955-Fps real-time shape measurement of a moving/deforming object using high-speed vision for numerous-point analysis," Proc. 2007 IEEE International Conference on Robotics and Automation, pp.3192–3197, 2007.

[11] Z. Wang, W. Wu, X. Xu, and D. Xue, "Recognition and location of the internal corners of planar checkerboard calibration pattern image," Applied Mathematics and Computation: Special Issue on Intelligent Computing Theory and Methodology, vol.185, no.2, pp.894–906, 2007.

[12] J. Lee, S. Hudson, and P. Dietz, "Hybrid infrared and visible light projection for location tracking," UIST '07: Proc. 20th Annual ACM Symposium on User Interface Software and Technology, pp.57–60, 2007.

[13] T.P. Koninckx and L.V. Gool, "Real-time range acquisition by adaptive structured light," IEEE Trans. Pattern Anal. Mach. Intell., vol.28, no.3, pp.432–445, 2006.

[14] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines, "ilamps: Geometrically aware and selfconfiguring projectors," SIGGRAPH '06: ACM SIGGRAPH 2006 Courses, p.7, 2006.

[15] D. Chen and G. Zhang, "A new sub-pixel detector for x-corners in camera calibration targets," WSCG (Short Papers), pp.97–100, 2005.

[16] F. Blais. "Review of 20 years of range sensor development," J. Electronic Imaging, vol.13, no,1, pp.231–240, 2004.

[17] S. Chang, B. Alan, and F. Mark, "Automatic grid finding in calibration patterns using Delaunay triangulation," Tech. Rep., National Research Council of Canada, Aug. 2003.

[18] J. Forest and J. Salvi, "A review of laser scanning three-dimensional digitisers," Proc. IEEE/RJS Intfl Conf. Intelligent Robots and Systems, pp.73–78, 2002.

[19] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy, "Real-time 3d model acquisition," Proc. SIGGRAPH, pp.438–446, 2002.

[20] L. Zhang, B. Curless, and S.M. Seitz, "Rapid shape acquisition using color structured light and multi-pass dynamic programming," 1st IEEE International Symposium on 3D Data Processing, Visualization, and Transmission, pp.24–36, June 2002.

[21] R.I. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, ISBN: 0521623049, 2000.

[22] Z. Zhang, "A flexible new technique for camera calibration," IEEE Trans. Pattern Anal. Mach. Intell., vol.22, no.11, pp.1330–1334, 2000.

[23] G. Bradski, "The OpenCV library," Dr. Dobb's Journal of Software Tools, 2000.

[24] D.H. Chidiac, "Classification of image edges," Vision Interface '99, Troise-Rivieres, pp.17–24, 1999.

[25] J. Batlle, E. Mouaddib, and J. Salvi, "Recent progress in coded structured light as a technique to solve the correspondence problem: A survey," Pattern Recognit., vol.31, no.7, pp.963–982, 1998.

[26] D. Ziou and S. Tabbone, "Edge detection techniques — An overview," International Journal of Pattern Recognition and Image Analysis, vol.8, pp.537–559, 1998.

[27] J. Salvi, J. Batlle, and E.M. Mouaddib, "A robust-coded pattern projection for dynamic 3d scene measurement," Pattern Recognit., vol.19, no.11, pp.1055–1065, 1998.

[28] D. Caspi, N. Kiryati, and J. Shamir, "Range imaging with adaptive color structured light," IEEE Trans. Pattern Anal. Mach. Intell., vol.20, no.5, pp.470–480, May 1998.

[29] M. Proesmans, L. Van Gool, and A. Oosterlinck, "Active acquisition of 3D shape for moving objects," Proc. International Conference on Image Processing, vol.3, pp.647–650, 1996.

[30] S.M. Smith and J.M. Brady, "Susan — A new approach to low level image processing," Int. J. Comput. Vis., vol.23, pp.45–78, 1995.

[31] M. Maruyama and S. Abe, "Range sensing by projecting multiple slits with random cuts," IEEE Trans. Pattern Anal. Mach. Intell., vol.15, no.6, pp.647–651, 1993.

[32] P. Vuylsteke and A. Oosterlinck, "Range image acquisition with a single binary-encoded light pattern," IEEE Trans. Pattern Anal. Mach. Intell., vol.12, no.2, pp.148–164, 1990.

[33] G. Hu and G. Stockman, "3-d surface solution using structured light and constraint propagation," IEEE Trans. Pattern Anal. Mach. Intell., vol.11, no.4, pp.390–402, 1989.

[34] C. Harris and M. Stephens, "A combined corner and edge detection," Proc. Fourth Alvey Vision Conference, pp.147–151, 1988.

[35] J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Mach. Intell., vol.8, no.6, pp.679–698, 1986.

[36] J.-Y. Bouguet, Camera calibration toolbox for matlab, http://www.vision.caltech.edu/bouguetj/calib doc/

Vinh Ninh Dao was born in 1981. He received B.S. and M.S. degrees in the University of Tokyo in 2006 and 2008 respectively. He is doing Ph.D. research in the Interaction Technology Laboratory at the University of Tokyo.

Masanori Sugimoto received B.Eng., M.Eng. and Dr.Eng. degrees from University of Tokyo, Japan, in 1990, 1992, and 1995, respectively. Currently he is an associate professor at Department of Electrical Engineering and Information Systems, Graduate School of Engineering, University of Tokyo. His research concern includes human computer interaction, mobile and ubiquitous computing, human robot interaction, acoustic imaging, entertainment computing, and mixed reality.