Memoirs of the Faculty of Engineering, Okayama University, Vol.40, pp.44-52, January, 2006

Generating Dense Point Matches Using Epipolar Geometry

Yasuyuki SUGAYA* Kenichi KANATANI
Department of Computer Science
Okayama University
Okayama 700-8530 Japan

Yasushi KANAZAWA
Department of Knowledge-based Information Engineering
Toyohashi University of Technology
Toyohashi, Aichi 441-8580 Japan

(Received December 7, 2005)

Dense point matches are generated over two images by rectifying the two images to align epipolar lines horizontally, and horizontally sliding a template. To overcome inherent limitations of 2-D search, we incorporate the "naturalness of the 3-D shape" implied by the resulting matches. After stating our rectification procedure, we introduce our multi-scale template matching scheme and our outlier removal technique using tentatively reconstructed 3-D shapes. Doing real image experiments, we discuss the performance of our method and remaining issues.

1. Introduction

Two approaches exist for 3-D reconstruction from images: one is based on feature tracking through a video stream, a typical method being the Tomasi-Kanade factorization [15, 21]; the other is to directly match feature points in separate images [9, 14]. Here, we consider the latter.

The basic principle for point correspondence detection is first applying a feature extraction filter to the two images separately and then matching those points that have similar neighborhoods. However, this usually produces many incorrect matches, or "outliers". So, we impose various constraints such as the epipolar equation, homographies, and global consistencies and reject those that do not satisfy them as outliers [16, 25].

If we thus keep rejecting questionable matches, the number of remaining matches decreases. Often, they concentrate on a particular portion of the image, and we cannot reconstruct a detailed 3-D shape of the scene as a texture-mapped polyhedron having these points as its vertices.

The standard technique for overcoming this is to compute the *fundamental matrix* from the resulting matches and search for new matches along *epipolar lines*, which we hereafter abbreviate as *epipolars*. This is a well known procedure for stereo vision, and many efficient searching techniques have been studied. Among them is to transform the images so that epipolars become horizontal and have common heights, which makes template scanning very easy. Such an image transformation is called *image rectification*.

The concept of image rectification was first proposed by Ayache et al. [2, 3, 4], who assumed the cameras were calibrated. Since then, various techniques for simplification [22] and extension [5, 6] have been proposed. Hartley [8] presented a comprehensive theory of image rectification for uncalibrated cameras, and many variants for simplification [1], parameter optimization [10, 17], and extension to trinocular views [24] have been proposed.

All these techniques are to warp the images according to homographies (projective transformations), so some parts of the images may be mapped to infinity. To prevent this, Roy et al. [20] introduced cylindrical coordinates, and Pollefeys et al. [23] used polar coordinates. Oram [19] generalized them into a hybrid system.

In this paper, we first present a new rectification procedure using homographies. This is very close to existing methods [1, 8, 10, 17], but the computation is simpler, and its geometric meaning is clearer.

In order to increase matching accuracy, we use templates of multiple sizes and determine the correspondence by hierarchical search and majority voting. Then, we remove questionable matches by imposing global consistency.

However, *this type of 2-D search is inherently limited*, because we cannot remove wrong matches that satisfy the epipolar constraint and have similar neighborhoods. Removing such matches inevitably requires *3-D information* that tells us that the implied 3-D shape of the scene is "unnatural" in some sense.

In this paper, we present a technique for outliers using tentative 3-D shapes. Doing real image experi-

*E-mail sugaya@suri.it.okayama-u.ac.jp

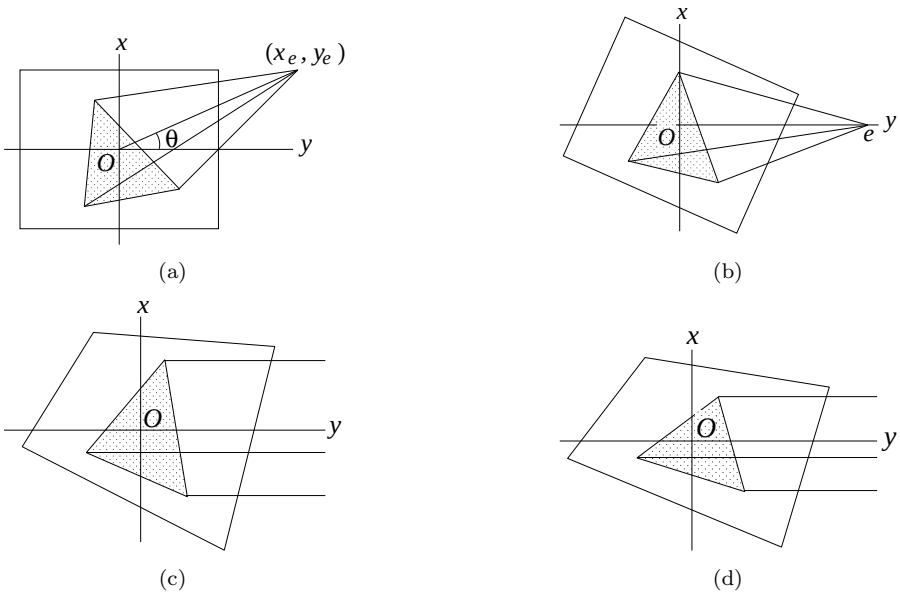

Figure 1: Rotating the image ((a)→(b)). Rectifying epipolars ((b)→(c)). Adjusting heights ((c)→(d)).

ments, we discuss the performance of our method and remaining issues.

2. Rectification Procedure

Our rectification procedure is as follows (Fig. 1):

1. Input eight or more correspondences over the two images.
2. Compute the fundamental matrix from them.
3. Compute the epipoles of the images from it.
4. Rotate the images so that the epipoles are on the horizontal axis.
5. Apply homographies to the two images and map the epipoles to infinity in the horizontal direction.
6. Apply a homography to the second image so that corresponding epipolars have the same height.

We now describes the details of each step.

2.1 Input correspondences

The input correspondences are supplied either by an automatic matching algorithm [16, 25] or by hand. This depends on applications, so our system regards the initial correspondences simply as input.

2.2 Fundamental matrix computation

Since the number of input correspondences may be small, we use a statistically optimal algorithm called *renormalization*[1] [11, 14] for computing the fundamental matrix with high accuracy.

Here, we need to specify the image coordinate system. In our system, we define the image origin $(0, 0)$ at the frame center and take the x-axis *upward* and y-axes *rightward* so that we can imagine the z-axis extending away from the viewer, defining a right-handed xyz coordinate system. We identify the camera optical axis with that hypothetical z-axis and imagine that the image is at distance f_0 (pixels)[2] from the camera lens center (the *viewpoint*).

Remark 1. Many people overlook the fact that *the numerical value of the fundamental matrix depends on the coordinate system involved*. For example, the value will be different from ours if the upper-left corner of the image frame is taken as $(0, 0)$. □

Remark 2. The camera model introduced here is *hypothetical*. For a real camera, the intersection of the optical axis with the image plane (the *principal point*) may not be at the frame center; the *aspect ratio*, the ratio of the horizontal and vertical intervals of the photo cell array, may not be 1; the *skew angle*, the angle made by the horizontal and vertical axes, may not be exactly 90°. But *these are irrelevant as far as image analysis is concerned*. Precise *camera calibration* is necessary only when we want to reconstruct the correct 3-D shape of the scene from images. □

[1] The source code is publicly available at `http://www.img.tutkie.tut.ac.jp/`

[2] As far as image analysis is concerned, the value f_0 is arbitrary. In our system, we set $f_0 = 600$.

2.3 Epipole computation

Let $\boldsymbol{F}$ be the computed fundamental matrix, and let $\boldsymbol{e}$ and $\boldsymbol{e}'$ be the unit eigenvectors of $\boldsymbol{F}^\top$ and $\boldsymbol{F}$, respectively, for eigenvalue 0. Since eigenvectors have sign indeterminacy, we chose the sign so that the third component is nonnegative (the sign is irrelevant if the third component is 0).

The points on the image plane in the direction of the vectors $\boldsymbol{e}$ and $\boldsymbol{e}'$ placed at the viewpoint are called the *epipole*.

Remark 3. Even in the presence of image noise, the fundamental matrix $\boldsymbol{F}$ is computed subject to the constraint $\det \boldsymbol{F} = 0$, so $\boldsymbol{F}^\top$ and $\boldsymbol{F}$ both eigenvalue 0. In practice, we simply compute the unit eigenvectors $\boldsymbol{e}$ and $\boldsymbol{e}'$ of positive semi-definite symmetric matrices $\boldsymbol{F}\boldsymbol{F}^\top$ and $\boldsymbol{F}^\top\boldsymbol{F}$, respectively, for the smallest eigenvalue. □

Remark 4. As is well known [9], the viewpoint of the second image is in the direction of $\boldsymbol{e}$ when viewed from the viewpoint of the first image, and the viewpoint of the first image is in the direction of $\boldsymbol{e}'$ when viewed from the viewpoint of the second image, However, these directions are relative to the *hypothetical cameras we are assuming*; they may not coincide with the physical directions. In order to let them coincide, we need precise camera calibration. Here, we do not do 3-D reconstruction, so we can arbitrarily assume the camera model. □

2.4 Image Rotations

The epipole (x_e, y_e) is given from the eigenvector $\boldsymbol{e} = (e_1, e_2, e_3)^\top$ as follows:

$$x_e = f_0 \frac{e_1}{e_3}, \qquad y_e = f_0 \frac{e_2}{e_3}. \tag{1}$$

We rotate the image around the image origin $(0, 0)$ so that this point is on the y-axis. By this rotation, each point (x, y) is mapped to a point $(\tilde{x}, \tilde{y})$ such that

$$\tilde{x} = x\cos\theta - y\sin\theta, \qquad \tilde{y} = x\sin\theta + y\cos\theta, \tag{2}$$

where θ is the angle of the vector $(x_e, y_e)^\top$ measured from either the positive or the negative side of the y-axis. After this rotation, the epipole is at $(0, e)$, where

$$e = x_e \sin\theta + y_e \cos\theta. \tag{3}$$

The second image is rotated similarly.

Remark 5. The angle θ is conveniently computed by

$$\theta = \begin{cases} \tan^{-1}(e_1/e_2) & |e_2| \geq |e_1| \\ \pi/2 - \tan^{-1}(e_2/e_1) & |e_1| > |e_2| \end{cases}. \tag{4}$$

Then, the epipole is mapped onto the positive side of the y-axis ($e > 0$) if the epipole (e_1, e_2) is in the half-plane $e_1 + e_2 \geq 0$ and the negative side ($e < 0$) if $e_1 + e_2 < 0$. □

Remark 6. Here, we are assuming that $e_3 \neq 0$. If $e_3 = 0$, the epipole is at infinity in the direction of $(e_1, e_2)^\top$ from the image origin, so $e = \pm\infty$. However, if we determine the angle θ by eq. (4), we can rotate the image irrespective of whether $e_3 = 0$ or not. □

Remark 7. We cannot use eq. (4) if $(x_e, y_e) = (0,0)$. Here, we are assuming that for both image the epipole is not inside the image frame, so $(x_e, y_e) \neq (0, 0)$ and hence $e \neq 0$. We discuss this assumption later in further details. □

2.5 Rectifying epipolars

As is well known [9], the point p' that corresponds to a point p in one image is on its epipolar in the other image, and all epipolars pass through the epipole of that image. Hence, if we apply a homography that maps the epipole to infinity, all the epipolars become parallel to each other. For this, we apply the following homography that maps $(0, e)$ to $(0, \pm\infty)$:

$$\hat{x} = \frac{\tilde{x}}{1 - \tilde{y}/e}, \qquad \hat{y} = \frac{\tilde{y}}{1 - \tilde{y}/e}. \tag{5}$$

The second image is transformed similarly.

Remark 8. A homography is a first order rational mapping with eight parameters [9], but we obtain eqs. (5) if we demand that

1. point $(0, e)$ be mapped to $(0, \pm\infty)$,
2. points on the x-axis (including the image origin) do not move, and
3. the rate of expansion of the y-axis be 1 at the image origin.

□

Remark 9. Note that the mapping (5) can be computed irrespective of whether $e_3 = 0$ or not; we only need to let $\tilde{y}/e$ in the denominator be 0 if $e_3 = 0$, in which case $e = \pm\infty$. Recall that we are assuming $e \neq 0$ (see Remark 7). □

2.6 Height adjustment

After the above procedure, all epipolars in each image are parallel to each other. Now, we expand/compress the second image vertically so that the corresponding epipolars have the same height. For this, we apply the following homography:

$$\bar{x}' = \frac{a\hat{x}' + b}{c\hat{x} + 1}, \qquad \bar{y}' = \frac{a\hat{y}'}{c\hat{x}' + 1}. \tag{6}$$

The coefficients a, b, and c are determined as follows. Let $\{(\hat{x}_\alpha, \hat{y}_\alpha)\}$ and $\{(\hat{x}'_\alpha, \hat{y}'_\alpha)\}$, $\alpha = 1, ..., N$, be the positions of the input N corresponding feature points $\{(x_\alpha, y_\alpha)\}$ and $\{(x'_\alpha, y'_\alpha)\}$ after the procedure described so far. We determine a, b, and c so that

$$\hat{x}_\alpha \approx \frac{a\hat{x}'_\alpha + b}{c\hat{x}'_\alpha + 1}, \qquad \alpha = 1, ..., N. \tag{7}$$

Eliminating the denominators, we minimize

$$J = \sum_{\alpha=1}^{2} \Big(a\hat{x}'_\alpha + b - \hat{x}_\alpha(c\hat{x}_\alpha + 1) \Big)^2. \tag{8}$$

After differentiation, we obtain the normal equation

$$\begin{pmatrix} \sum_{\alpha=1}^N \hat{x}'^2_\alpha & \sum_{\alpha=1}^N \hat{x}'_\alpha & -\sum_{\alpha=1}^N \hat{x}_\alpha \hat{x}'^2_\alpha \\ \sum_{\alpha=1}^N \hat{x}'_\alpha & N & -\sum_{\alpha=1}^N \hat{x}_\alpha \hat{x}'_\alpha \\ -\sum_{\alpha=1}^N \hat{x}_\alpha \hat{x}'^2_\alpha & -\sum_{\alpha=1}^N \hat{x}_\alpha \hat{x}'_\alpha & \sum_{\alpha=1}^N \hat{x}^2_\alpha \hat{x}'^2_\alpha \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_{\alpha=1}^N \hat{x}_\alpha \hat{x}'_\alpha \\ \sum_{\alpha=1}^N \hat{x}_\alpha \\ -\sum_{\alpha=1}^N \hat{x}^2_\alpha \hat{x}'_\alpha \end{pmatrix}, \tag{9}$$

from which we obtain a, b, and c. If the input matches were exact, eq. (7) should hold with equality. Since this is not the case in general (e.g., they can be specified at most to integer pixel values), we measure the accuracy of the resulting rectification by

$$h = \sqrt{\frac{1}{N} \sum_{\alpha=1}^{N} (\hat{x}_\alpha - \bar{x}'_\alpha)^2}, \tag{10}$$

where $(\bar{x}'_\alpha, \bar{y}'_\alpha)$ is the corrected position of $(\hat{x}'_\alpha, \hat{y}'_\alpha)$.

Remark 10. A homography has eight parameters as mentioned earlier, but we obtain eqs. (6) if we demand that

1. horizontal lines be mapped to horizontal lines,
2. points on the y-axis be mapped to points on the y-axis, and
3. the rate of expansion be the same for the x-axis and the y-axis at the image origin

□

Remark 11. The above procedure is slightly different from those described in the literature [1, 8, 10, 17]. Hartley [8] showed that the homography that horizontally aligns epipolars are analytically given from the fundamental matrix $\boldsymbol{F}$, and existing methods all follow this strategy. Hence, no height adjustments are necessary. However, the homography determined by Hartley's method has three degrees of indeterminacy. In fact, epipolars remain the same position if the second image is

- horizontally translated,
- horizontally expanded/compressed, and
- sheared (or skewed) in such a way that the x-axis is slanted while the y-axis is fixed.

Existing methods differ in how to fix them using some kind of optimization. Our method is equivalent to fix them so that

1. the origin of the first image is fixed,
2. the horizontal position of the origin of the second image is fixed, and
3. the aspect ratio (the ratio between of vertical and horizontal rates of expansion/compression) and the orthogonality are preserved at the origin of each image.

Thus, not only do we need no optimization but also the geometric properties of the resulting images are clear, while they are often difficult to grasp for existing methods. In this respect, our method is considered to be more suitable. □

Remark 12. Theoretically, we should adjust the height so as to minimize eq. (10) (or its square). Here, we use the least-squares minimization of eq. (8) for computational simplicity. This is optimal if the expansion/compression is uniform (i.e., $c = 0$), and this should be sufficient in practical applications, where c is usually very small.

□

3. Template Matching

3.1 Image rectification

For the first image, the composition of all the mappings is written in terms of vectors and matrices in the form

$$\hat{\boldsymbol{x}} = Z[\boldsymbol{GRx}], \quad \boldsymbol{x} = \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}, \quad \hat{\boldsymbol{x}} = \begin{pmatrix} \hat{x}/f_0 \\ \hat{y}/f_0 \\ 1 \end{pmatrix} \tag{11}$$

$$\boldsymbol{R} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}, \qquad \boldsymbol{G} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\gamma & 1 \end{pmatrix} \tag{12}$$

$$\gamma = \frac{e_3}{e_1 \sin\theta + e_2 \cos\theta} \tag{13}$$

where $Z[\,\cdot\,]$ denotes scale normalization so that the z component is 1, and θ is the angle computed by eq. (4). For the second image, we obtain

$$\bar{\boldsymbol{x}}' = Z[\boldsymbol{KG}'\boldsymbol{R}'\boldsymbol{x}'] \tag{14}$$

$$\boldsymbol{x}' = \begin{pmatrix} x'/f_0 \\ y'/f_0 \\ 1 \end{pmatrix}, \quad \bar{\boldsymbol{x}}' = \begin{pmatrix} \hat{x}'/f_0 \\ \hat{y}'/f_0 \\ 1 \end{pmatrix} \tag{15}$$

$$\boldsymbol{K} = \begin{pmatrix} a & 0 & b/f_0 \\ 0 & a & 0 \\ cf_0 & 0 & 1 \end{pmatrix} \tag{16}$$

where $\boldsymbol{G}'$ and $\boldsymbol{R}'$ are the matrices defined by eqs. (12) and (13) for the second image.

Remark 13. For actual computation, we define two empty image frames and copy the intensity (or color) values of the original images via the inverse mappings of eqs. (11) and (14):

$$\boldsymbol{x} = Z[\boldsymbol{R}^\top \boldsymbol{G}^{-1} \hat{\boldsymbol{x}}], \quad \boldsymbol{x}' = Z[\boldsymbol{R}'^\top \boldsymbol{G}'^{-1} \boldsymbol{K}^{-1} \bar{\boldsymbol{x}}'] \tag{17}$$

□

Remark 14. The value γ in eq. (13) is simply f_0/e, but this expression allows us to compute it irrespective of whether $e_3 = 0$ or not. □

Remark 15. As mentioned in Remark 7, we are assuming that the epipole is outside the image frame for both images. If it is inside the image frame, the part that approaches the epipole from the image origin is mapped infinitely far away while the part that passes through it comes back from the opposite direction. We can avoid such a singularity by using cylindrical or polar coordinates around the epipole [19, 23, 20]. However, this anomaly occurs when the camera motion is nearly along the optical axis, and such a camera motion significantly reduces the accuracy of 3-D reconstruction [23]. In practice, we should avoid such a camera motion. Here, we assume that camera motion is more or less horizontal and output a warning message if the epipole is within a distance[3] L from the origin in either image. The warning condition is

$$|\gamma| \geq \frac{f_0}{L}. \tag{18}$$

□

Remark 16. If the mappings (17) are applied to pixel positions, the computed positions are between pixels in general. Our system estimates the intensity value there by bilinear interpolation. □

3.2 Multi-scale template matching

For generating point correspondences over the rectified images, we first extract feature points from the first original images, using the Harris operator[4] [7] and map them onto the first rectified image by eq. (11). Then, we search for their corresponding positions on the second rectified image by horizontal template matching: we cut out a square region around each feature point in the first image as a template and look for the position in the second at which the template matches with the least residual. The detected positions are inversely mapped onto the original second image by eq. (17).

The difficulty is the choice of the template size. If the two images look almost the same, the matching becomes robust by using a large template. If not, we should use a small template, but we do not know which is the case in advance.

In our system, we use templates of five sizes 33×33, 17×17, 9×9, 5×5, and 3×3 and apply Gaussian smoothing of sizes 17×17, 9×9, 5×5, 3×3, and 0×0 (i.e., no filtering), respectively, before the template matching (the smoothing is done over the original images). The standard deviation of smoothing is 8, 4, 2, 0.5, and 0, respectively. For determining correspondence, we use the following two strategies:

- **Hierarchical search**. Letting $s = 16$, we search for a corresponding pixel using the template of size $(2s+1) \times (2s+1)$. If we find one, we search nearby using the template of size $(2s+1) \times (2s+1)$. If the newly found pixel is apart from the previous one by s pixels or more, we judge that no correspondence exists. Otherwise, we halve s and repeat the same procedure until $s = 1$.

- **Majority voting**. We find for each feature point in the first image five corresponding points using the templates of five different sizes separately. Let $y_1 \leq \cdots \leq y_5$ be their positions. If the shortest of the intervals $[y_1, y_3]$, $[y_2, y_4]$, and $[y_3, y_5]$ is of four pixels or less, we decide the corresponding point to be at the average of the four positions in that shortest interval. Otherwise, we judge that no correspondence exists.

According to our experiments, both worked satisfactorily, and we were unable to decide which is universally better than the other. So, we let the user choose either as an option.

Remark 17. The similarity of images is usually measured by the *residual sum of square*

$$\mathrm{RSS} = \sum_{(x,y)\in\mathcal{T}} (I_1(x,y) - I_2(x',y'))^2, \tag{19}$$

where $\mathcal{T}$ is the template region, I_1 and I_2 are the intensity values of the first and second images, respectively, and (x', y') is the pixel to be compared

[3] In our system, we take L to be the image size.

[4] The source code is publicly available at `http://www.img.tutkie.tut.ac.jp/programs/`

with pixel (x, y) in the template region. If illumination changes occur between the images, this RSS may not give a low value for correct matching. This can be avoided by normalizing the intensity values of the two images in such a way that their mean is 0 and their variance is 1 inside the template region. This is mathematically equivalent to maximizing what is known as the *normalizing correlation*. However, this also deteriorates the matching performance, sometimes matching two points that should not be matched. So, we let the user decide as an option whether or not the intensity is normalized. □

Remark 18. The heights the feature points in the first rectified image are real numbers in general. Not rounding them to integers, we directly search along the corresponding horizontal lines in the second rectified image. The inter-pixel intensity value is computed by bilinear interpolation. □

Remark 19. According to our experiments, the effect of the Gaussian smoothing is very small, producing no appreciable differences. In some cases, however, it helps match points on object occluding boundaries with different backgrounds. □

Remark 20. The Gaussian smoothing is applied to all pixels including those near the frame boundary, in which case we use a truncated Gaussian kernel. Similarly, the template is scanned over all pixels including those near the frame boundary, in which case we evaluate the similarity in terms of per pixel. So, no anomalies occur near the frame boundary. □

3.3 Subpixel correction

Theoretically, corresponding points should exactly be on their epipolars. However, the original feature points can be located only up to integers even if no other noise sources exist, and the computed epipolars are not exact, either. On top of that, we search along the epipolars only at one-pixel intervals, so the matching accuracy is limited at most to one pixel.

We improve this by doing subpixel search: we translate the template around the detected position up, down, left, and right by distance h and move to the position that gives the smallest residual value. From there, we repeat the same procedure after halving the distance h until h is sufficiently small. The initial value of h should reflect the accuracy of the rectification, so start with the value given by eq. (10).

3.4 Global consistency

Some outliers may still remain after the above procedure. These can be easily found if we observe the "optical flow" (the horizontal line segments connecting corresponding positions by identifying the two frames). Very long segments are mostly due to mismatches, so we remove them. To be specific, we evaluate the mean μ and the standard deviation σ of the segment lengths and remove those outside the interval $[\mu - 2\sigma, \mu + 2\sigma]$ around μ.

4. Use of 3-D Information

Although almost all outliers are removed by the above procedure, there may still remain a few, for which

1. the epipolar constraint is satisfied, and
2. the neighborhoods of the two points look the same.

Such outliers occurs, for example, at a "T-junction" (the intersection of the boundaries of a nearby object and another object behind it); their appearances look as if the same, yet they are physically different parts of the scene. Theoretically, *it is impossible to remove such false matches by 2-D image processing alone*.

However, such false matches can be detected if we use 3-D information about the scene. If we do 3-D reconstruction from the detected matches, the 3-D positions computed from false matches have very large variations in their depths. If we display the 3-D shape as a polyhedron having the matching points as vertices, false matches usually result in marked "spikes". So, we remove such spikes by the following procedure.

We compute the fundamental matrix optimally from the given matches [11] and computing the depth Z of each feature point p by the method of Kanatani and Ohta [14]. Then, we remove those points that have negative depths (for one or both of the images). Recomputing the fundamental matrix from the remaining matches, we repeat the same procedure until no negative depths arise.

Next, we define a Delaunay triangular mesh with vertices at the feature points in the first image and represent the 3-D shape of the scene as polyhedral surface. We define for each vertex p the *discrete Laplacian* $L(p)$ by

$$L(p) = \frac{Z - (\text{average depth of incident points})}{(\text{average horizontal length of incident edges})}, \tag{20}$$

where the "incident points" means those points that are connected to the point p by the edges of the Delaunay triangulation, and the "incident edges" mean those edges. Then, the "horizontal length" of points (X, Y, Z) and (X', Y', Z') is defined to be $\sqrt{(X - X')^2 + (Y - Y')^2}$.

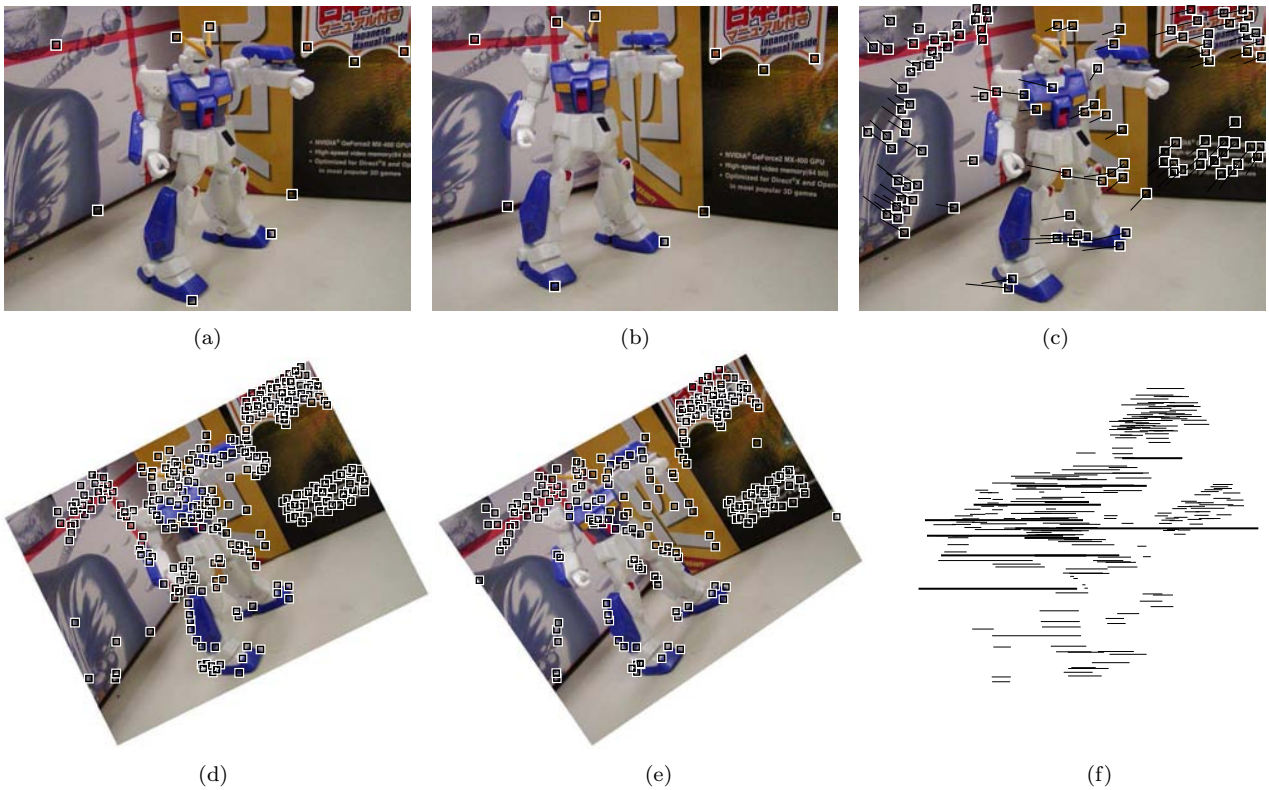

Figure 2: (a), (b) Input images and input correspondences. (c) Correspondences generated by automatic matching ("optical flow"). (d) Feature points in the first rectified image. (e) Detected corresponding points in the second rectified image. (f) "Optical flow" of the correspondence between (d) and (e). Thick line segments are removed by the global consistency judgment.

Setting a threshold[5] θ, we decide that a vertex p is a "spike" if

$$|L(p)| > \theta, \tag{21}$$

and if p is an extremal point, i.e.,

$$Z > \text{(maximum depth of incident points)}$$
$$\text{or}$$
$$Z < \text{(minimum depth of incident points)}.$$

After removing such "spikes", we recompute the fundamental matrix from the remaining matches and repeat the same procedure until no spikes appear.

Remark 21. The method Kanatani and Ohta [14] first computes the focal lengths of the two cameras and then computes the depths of the individual feature points. If the correspondences contain false matches, the computed fundamental matrix may not be accurate, resulting in imaginary focal lengths (i.e., some expressions inside root squares become negative), and elaborate methods were presented for avoiding this [13, 18]. Here, however, we do not need precise focal lengths. If we use wrong values, the reconstructed shape is a transformation of the true shape by a 3-D homography, known as *projective reconstruction* [9]. Still, the qualitative shape is preserved. The purpose of 3-D reconstruction here is tentative for removing "spikes", so it suffices to use an appropriate approximation to the focal lengths. □

5. Real Image Examples

Figs. 2(a),(b) show two input images. If we extract 300 feature points from them separately using the Harris operator and automatically match them by the method of Kanazawa and Kanatani[6] [16], we obtain 109 correspondences in Fig. 2(c), where the positions in the first image and their "optical flow" are shown. We can see that they concentrate on planar background parts.

Figs. 2(d),(e) are rectification of Figs. 2(a),(b) using 10 point correspondences given by hand. We detected 300 feature points from the first original image using the Harris operator and mapped them onto Fig. 2(d) as shown there. Doing multi-scale template

[5] In our system, we set $\theta = 3$.

[6] The source code is publicly available at `http://www.img.tutkie.tut.ac.jp/programs/`

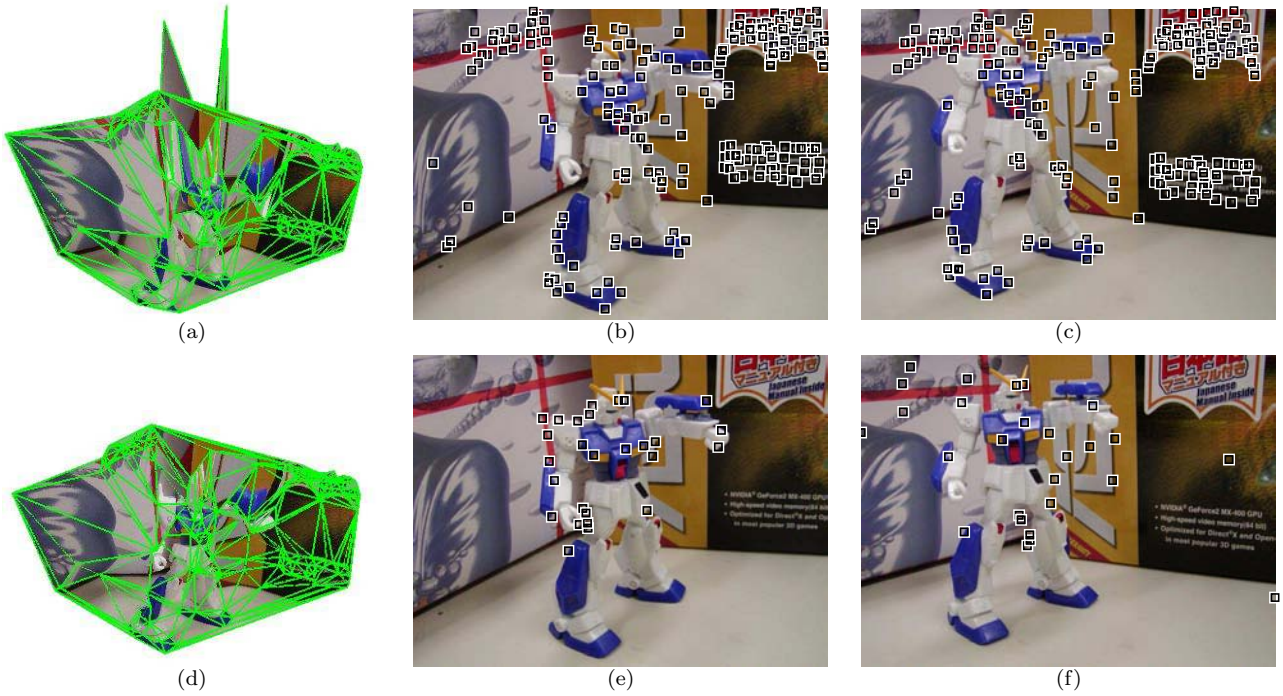

Figure 3: (a) Tentative 3-D reconstruction from the correspondences in Fig. 2. (b), (c) Final corresponding points using 3-D information. (d) Corresponding 3-D shape. (e), (f) Positions of removed feature points by our method.

matching described in Sec. 3.2 (we used hierarchical search), we found 255 corresponding points as indicated in Fig. 2(e). Fig. 2(f) shows their "optical flow". The thick line segments are judged to be mismatches by the global consistency judgment described in Sec. 3.4.

Fig. 3(a) shows the resulting tentative 3-D shape. Removing 16 matches that cause marked spikes (no negative depths arises in this case), we finally obtain 198 matches as shown in Fig. 3(b),(c). We obtain many matching points on the isolated object surface, too. Fig. 3(d) shows the resulting 3-D shape. Figs. 3(e),(f) show all the positions of feature points removed by our method.

We did many experiments using many other images and found that the removal of matches occur in the following cases:

1. Some points in the first image may do not have corresponding points in the second image: they are occluded by objects in front or outside the image frame.

2. Some points in a nearly periodical pattern are matched to similar but wrong points in the pattern.

3. Some points have similar neighborhoods by chance but they do not correspond to each other.

4. T-junctions are matched to the corresponding T-junctions in the other

5. Some correct matches are removed because large depth variations exit; they are removed by the global consistency judgment or regarded as causing spikes.

It has been well known that mismatches arise in Cases 1∼4. This is a major obstacle to 3-D reconstruction from images. In Cases 1∼4, it is difficult to remove such mismatches by 2-D image processing as long as they satisfy the epipolar constraint. In order to remove them, we must necessarily resort to 3-D information as we did here.

On the other hand, if we require some kind of "naturalness" of the scene, as we did here, some correct matches may also be removed (Case 5). This is an inevitable trade-off, and reducing such false negatives without increasing false positives is an important remaining issue.

6. Concluding Remarks

We described a new procedure[7] for generating dense point matches over two images: we compute the fundamental matrix from given initial matches,

[7]The source code is publicly available at `http://www.suri.it.okayama-u.ac.jp/`

rectify the two images so that all epipolars are horizontal, and find dense matches by template matching. To increase the matching accuracy, we introduced multi-scale template matching and global consistency judgment.

Our rectification procedure is different from existing ones based on Hartley' theory [8]. Our method does not require any optimization procedure, and the geometric meaning is very clear.

However, such 2-D search is inherently limited. In order to overcome this, we introduced an outlier removal technique using a tentative reconstructed 3-D shape. Using real images, we confirmed the effectiveness of our method and discussed remaining issues.

Acknowledgments: This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology, Japan, under a Grant in Aid for Scientific Research C(2) (No. 17500112).

References

[1] K. A. Al-Shalfan, J. G. B. Haigh and S. S. Ipson, Direct algorithm for rectifying pairs of uncalibrated images, *Electronics Lett.*, **36**-5 (2000-3), 419–420.

[2] N. Ayache, *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*, MIT Press, Cambridge, MA, U.S.A., 1991.

[3] N. Ayache and C. Hansen, Rectification of images for binocular and trinocular stereo vision, *Proc. 9th Int. Conf. Pattern Recog.*, November 1988, Rome, Italy, pp. 11–16.

[4] N. Ayache and F. Lustman, Trinocular stereo vision for robotics, *IEEE Trans. Pattern Anal. Mach. Intell.*, **13**-1 (1991-1), 73–85.

[5] A. Fusiello, E. Trucco and A. Verri, Rectification with unconstrained stereo geometry, *Proc. British Mach. Vision Conf.*, September 1997, Essex, U.K., pp. 400–409.

[6] A. Fusiello, E. Trucco and A. Verri, A compact algorithm for rectification of stereo pairs, *Mach. Vision Appl.*, **12**-1 (2000-6), 16–22.

[7] C. Harris and M. Stephens, A combined corner and edge detector, Proc. 4th Alvey Vision Conf., Manchester, U.K., August 1988, pp. 147–151.

[8] R. Hartley, Theory and practice of projective rectification, *Int. J. Comput. Vision*, **35**-2 (1999), 115–127.

[9] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, U.K., 2000.

[10] F. Isgrò and E. Trucco, Projective rectification without epipolar geometry, *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, June 1999, Fort Collins, CO, U.S.A., Vol. 1, pp. 94–99.

[11] K. Kanatani, Optimal fundamental matrix computation: Algorithm and reliability analysis, *Proc. 6th Symp. Sensing via Image Information*, June, 2000, Yokohama, Japan, pp. 291–298.

[12] K. Kanatani and Y. Kanazawa, Automatic thresholding for correspondence detection, International Journal of Image and Graphics, **4**-1 (2004-1), 21–33.

[13] K. Kanatani, A. Nakatsuji and Y. Sugaya, Stabilizing the focal length computation for 3-D reconstruction from two uncalibrated views, *Int. J. Comput. Vision*, 2005, to appear.

[14] K. Kanatani and N. Ohta, Comparing optimal three-dimensional reconstruction for finite motion and optical flow, *J. Elec. Imag.*, **12**-3 (2003-7), 478–488.

[15] K. Kanatani and Y. Sugaya Factorization without factorization: Complete recipe, *Mem. Fac. Eng. Okayama Univ.*, **38**-1/2 (2004-3), 61–72.

[16] Y. Kanazawa and K. Kanatani Robust image matching preserving global consistency, *Proc. 6th Asian Conf. Comput. Vision*, January 2004, Jeju, Korea, Vol. 2, pp. 1128–1133.

[17] C. Loop and Z. Zhang, Computing rectifying homographies for stereo vision, *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, June 1999, Fort Collins, CO, U.S.A., Vol. 1, pp. 125–131.

[18] A. Nakatsuji, S. Takahashi, Y. Sugaya and K. Kanatani, Stabilizing the focal length computation for 3-D reconstruction from two uncalibrated views, *Proc. 6th Asian Conf. Comput. Vision*, January 2004, Jeju, Korea, Vol. 1, pp. 1–6.

[19] D. Oram, Rectification for any epipolar geometry, *Proc. British Mach. Vision Conf.*, September 2001, London, U.K., pp. 653–662.

[20] S. Roy, J. Meunier and I. J. Cox, Cylindrical rectification to minimize epipolar distortion, *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, June 1997, Puerto Rico, pp. 393–399.

[21] C. Tomasi and T. Kanade, Shape and motion from image streams under orthography—A factorization method, *Int. J. Comput. Vision*, **9**-2 (1992-10), 137–154.

[22] D. V. Papadimitriou and T. J. Dennis, Epipolar line estimation and rectification for stereo image pairs, *IEEE Trans. Image Process.*, **5**-4 (1996-4), 672–679.

[23] M. Pollefeys, R. Koch and L. Van Gool, A simple and efficient rectification method for general motion, *Proc. 7th Int. Conf. Comput. Vision*, September 1999, Kerkyra, Greece, Vol. 1, pp. 496–501.

[24] C. Sun, Uncalibrated three-view image rectification, *Image Vision Comput.*, **21**-3 (2003-3), 259–269.

[25] Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong, A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry, *Artif. Intell.*, **78** (1995), 87–119.