FCV2004 ; The 10th Korea-Japan Joint Workshop on Frontiers of Computer Vision, pp.25-30 (2004)

Extraction and Elimination of parallel obstacles for image restoration

Hiroshi Suzuki†, Toru Tamaki†, Masanobu Yamamoto‡
†Graduate School of Science and Technology, Niigata University, JAPAN
‡Dept. of Information Engineering, Fac. of Engineering, Niigata University, JAPAN
{hsuzuki@vision.ie,tamaki@ie}.niigata-u.ac.jp

Abstract We propose a technique for extracting and remove a fence region in a still image. The proposed method uses the fact that the fence region consists of periodic and parallel straight lines in a space. Characteristic of parallel and periodic lines in an image is used for the fence parallel to the image plane. Property of a projection of periodic parallel lines in a space is used for the other case. The extracted region is removed and interpolated by a disoccluding technique. Experimental results show the validity of this technique by applying to real images.

1 Introduction

Taking a picture in outdoor by a digital camera, a scene that people want to keep in a photograph is often occluded by various obstacles, such as branches, trees and fences. Especially fences settled in a zoo or a ballpark can not be taken away, so that we always see animals and players behind the diamonds of a fence. This problem will happen when a mobile robot wanders in a restricted area surrounded by fences, or cameras of a surveillance system are placed in a cage protecting them from outsiders. These cases require to see and recover the occluded scene, and there are two related techniques; detecting the occluding objects and disocclude them.

Disocclusion, or fill-in process, has been studied in computer vision inspired by psychology studies. Once given regions to be filled-in in a still image, these computational approaches recover the regions of missing data by PDE [1], variational approach [2], autocorrelations [3] and level lines [4]. Sequence restoration, or movie correction, is also developed [5] to detect and recover regions on a movie film damaged by spots, dusts, scratches and so on. Special effects in recent films require the technique by another reason of removing undesired objects, such as wires. However, because of the wide range of applications, the methods do not take account of detection and segmentation of objects to be removed. Some of them use temporal information[6] for sequence restoration, but the others require manual operations to specify the regions to be processed.

Detecting and removing objects occluding a scene or desired objects have been proposed for a specific task; removing glasses [7] of a face image using models of faces and glasses, text regions from TV program[8] with colors and strokes of letters, and water-drops on a camera using multi cameras [9]. These methods can not be applicable to other situations because of its dependency on their targets, while more general segmentation techniques (clustering, snake and level sets, etc.) are rarely used for a known object in a scene.

In this paper, we propose a method for extracting a fence occluding a scene, and remove regions of the fence and recover the occluded background scene. We use characteristics of the fence that is approximately comprised of parallel lines in a three-dimensional space. For a fence parallel to the image plane, the proposed method extracts parallel and periodical lines in an image. To recognize a fence that is inclined to a camera and perspectively projected on the image plane, results from the projective geometry is used to locate the projected lines. The lines of a fence in the image are masked to recover the regions by disocclusion techniques.

This paper is organized as follows; in section 2, the method for the parallel case is described, and the case that a fence is not parallel to the image plane is explained in section 3. Experimental results are shown in section 4, then discussions and conclusions are given in section 5.

2 Localizing a fence parallel to the camera

In this section, we describe a method to localizing parallel lines in a scene. First we consider a fence that is parallel to the image plane so that parallel lines in three-dimensional space are projected to parallel lines in the image plane.

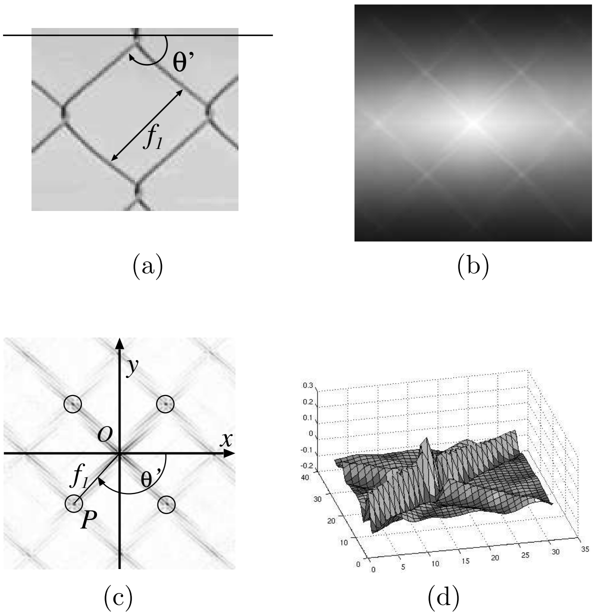


Fig. 1: The interval f_1 and the angle θ' of a fence. (a) A fence image I represented by f_1 and θ'. (b) The autocorrelation I_a. (c) Four peaks in the autocorrelation indicated by the circles. (d) Correlation plotted with shaded surface around one of the peaks.

2.1 Angles and interval of diamonds of a fence

Usually there are two sets of parallel lines crossing each other in an image of a fence. They are identified by the angles to the horizontal line and the intervals between lines (see Fig.1(a)). These two parameters, θ' and f_1, are estimated by using spatial autocorrelation of the image. The autocorrelation $I_a(x, y)$ of an image $I(x, y)$ is represented by

$$I_a = \mathcal{F}^{-1}\{|\mathcal{F}\{I\}|^2\} \qquad (1)$$

where $\mathcal{F}$ means the Fourier transform. In Fig.1 (b), autocorrelation I_a calculated from the edge image of Fig.1(a) as $I(x, y)$ is shown with brighter pixels indicating high correlation. Usually the correlation decreases as it goes away from the origin. To remove the bias, the Laplacian filter is applied to the autocorrelation I_a. In Fig.1(c), except around the origin with strong correlation, four prominent peaks are easily found. They correspond to the two groups of parallels lines because of the symmetry of the power spectrum, so two sets of a peak and an angle are detected. The distance $\overline{PO}$ between the origin O and one of the peaks P (see Fig.1(d)) determines the interval of the parallel lines, f_1, and the angle between $\overline{PO}$ and the x axis is identical to θ', the angle of parallel lines in the original image.

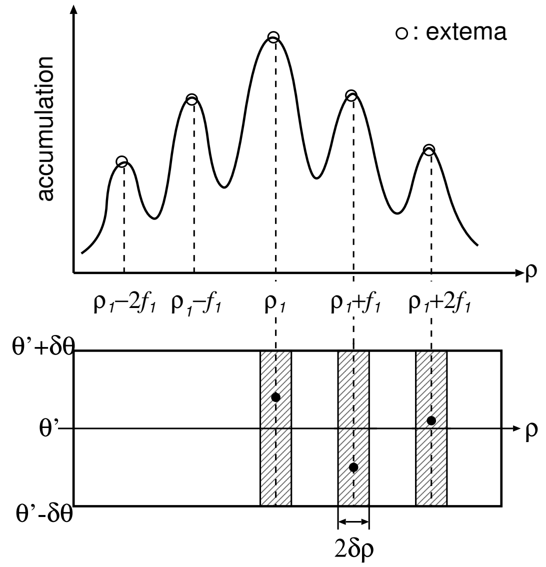


Fig. 2: (Top) Profile of accumulation for $\theta = \theta'$ is drawn with maximum and extrema. (Bottom) Search space for extrema are represented by the gray areas.

2.2 Detecting lines by Hough transform

Based on the rough estimates of the interval and the angle, the θ-ρ Hough transform is applied to the edge of a fence image. Instead of exhaustive voting in the θ-ρ parameter space, we limit the vote of the Hough transform for θ to $\theta' \pm \delta\theta$ (here we use $\delta\theta = 3$[deg] according to preliminary experiments).

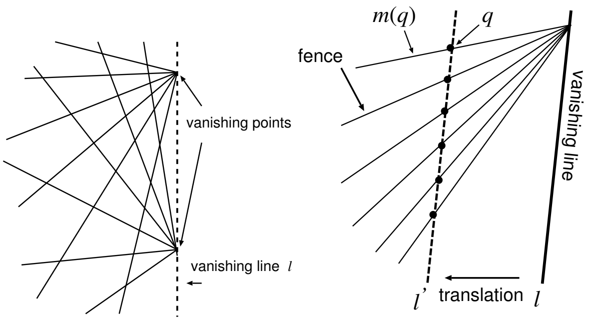
The parallel lines in an image are represented as a series of points in the parameter space at even intervals on a line along the ρ axis. In the accumulation of the Hough transform, the parallel lines are represented as extrema of broad peaks.

To find the peaks for each line precisely, we begin to search extrema (Fig.2) with the maximum ρ_1 for θ'. Then extremum is found in the local area that is f_1 away from ρ_1; $[\rho_1 + f_1 - \delta\rho, \rho_1 + f_1 + \delta\rho] \times [\theta' - \delta\theta, \theta' + \delta\theta]$. In the same manner, extrema are searched in the local areas $[\rho_1 + nf_1 - \delta\rho, \rho_1 + nf_1 + \delta\rho] \times [\theta' - \delta\theta, \theta' + \delta\theta]$ for $n = \pm 1, \pm 2, \ldots$. Here we use $\delta\rho = 5$.

The procedure above is applied to each of two parameter sets, then the extrema found in the parameter space are projected backward onto an image as lines as shown in Fig.9(b).

3 Localizing a fence not parallel to the camera

Since the method described above uses the parallelism of the lines, it is not applicable to finding lines of a fence not parallel to the image plane in



(a) (b)

Fig. 3: (a) Vanishing points and vanishing line. (b) Intersections of the translated vanishing line and the projected lines of a fence.

a space. This is more general, but the proposed method in this section is not an extension of the method in the previous section because the natures of the lines are rather different.

When parallel lines in a three-dimensional space are projected onto an image plane, the projections form a set of lines passing through a vanishing point on the image. If there are other sets of parallel lines in the space and all of them are on the same plane in the space, vanishing points made by projecting each of them make a vanishing line in the image plane (Fig.3(a)).

First the vanishing points are estimated, and then lines passing through each of them are determined in the accumulation of the Hough space.

3.1 Estimating vanishing points by LMeds

We describe the estimation of the vanishing points that correspond to the two groups of the parallels lines on a fence. Since the length of straight lines in the fence considered in this paper is limited, and the direction of the lines and the direction to which a fence is stretched are not identical, the vanishing points conformed by the lines do never appear in the image (an example is shown in Fig.5) and this makes the estimation instable. Moreover, finding the vanishing point based on edge of a fence image is not straightforward because of the existence of many clutters in the background behind the fence. Although an elaborated algorithm to find vanishing points [10] has been proposed, we can here use an user-interaction and the information of lines in the parameter space of the Hough transform.

Now, the LMeds method estimates the vanishing points. For each vanishing point, the proposed method requires a user to specify a range of angles of the lines in the image by using a GUI tool. Then the range specifies an area in the parameter space

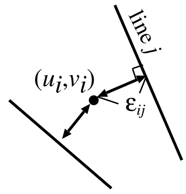


Fig. 4: Distance ε_{ij} between the intersection and lines.

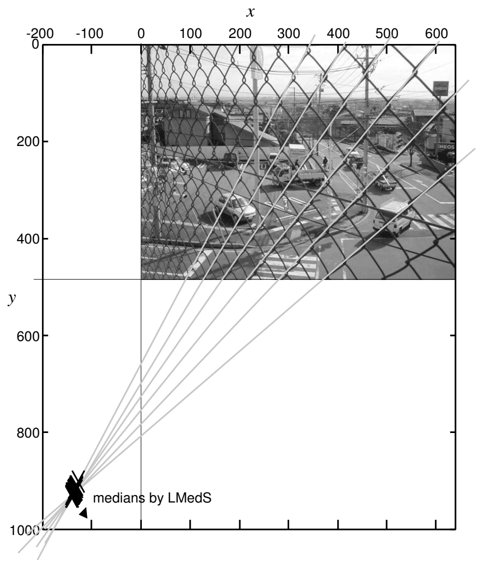


Fig. 5: Estimates of the vanishing point by LMeds method. Medians for each trial are plotted by $\times$, and several lines drawn manually are superimposed on the fence image.

of the Hough transform. In the area, n lines (i.e., n largest accumulated points) are chosen.

For i th trial ($i = 1, 2, \ldots, q$), two lines are randomly selected from the n lines, and the intersection (u_i, v_i) of them are calculated. Distance ε_{ij} ($j = 1, 2, \ldots, n-2$) between the intersection and a line j, that is one of the rest of $(n-2)$ lines, is calculated (Fig.4). The median of the distances ε_i is

$$\varepsilon_i = \underset{1 \le j \le n-2}{\text{med}} \ \varepsilon_{ij} \tag{2}$$

The minimum of the distance ε_i for q trials is the estimate of the vanishing point $\boldsymbol{v} = (u, v)$;

$$\boldsymbol{v} = (u, v) = (u_{i^*}, v_{i^*}), \quad i^* = \underset{1 \le i \le q}{\operatorname{argmin}} \, \varepsilon_i \tag{3}$$

Here $n = 100$ and $q = 200$ are used.

Each vanishing point is estimated by the method above, and then the vanishing line passes the two vanishing points are computed.

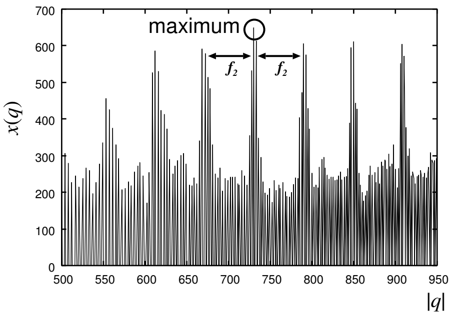


Fig. 6: Accumulation $x(\boldsymbol{q})$ of lines $m(\boldsymbol{q})$ on the line l'.

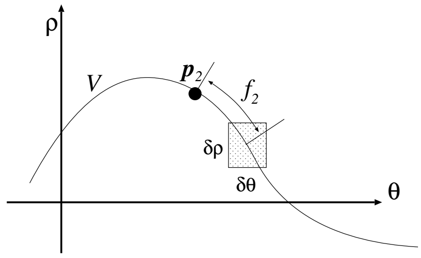


Fig. 8: The areas where the extrema is searched. The rectangle is the search area, and the distance from $\boldsymbol{p}_2$ along the curve V is f_2.

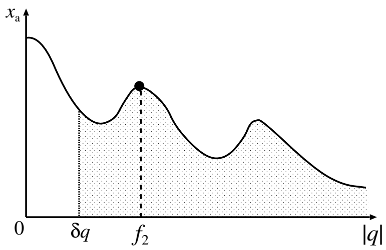


Fig. 7: Autocorrelation x_a of $x(\boldsymbol{q})$.

3.2 Accumulation on the vanishing line

Next, the lines of the fence in the image are determined based on the estimated vanishing line. Although n lines was chosen for LMeds method, many false lines (not on the fence) are involved because of the edge of the background in a scene. To find true lines (on the fence), we use a compound ratio of the fence lines.

Let $\boldsymbol{q}$ be a point on l' that is the translated vanishing line, and $m(\boldsymbol{q})$ be a group of lines that pass both $\boldsymbol{q}$ and one of the vanishing points, $\boldsymbol{v}$ (Fig. 3(b)). In the parameter space of the Hough transform, lines $m(\boldsymbol{q})$ are on a sinusoidal curve V (Fig.8) that represents lines passing through $\boldsymbol{v}$ in the image:

$$\boldsymbol{v} \cdot (\cos\theta, \sin\theta) = \rho \tag{4}$$

We write $x(\boldsymbol{q})$ as the accumulation in the Hough parameter space along the curve V. If a point $\boldsymbol{q}$ is on the fence, then $m(\boldsymbol{q})$ represents a line on the fence and $x(\boldsymbol{q})$ has a larger value than values not on the fence. Therefore, $x(\boldsymbol{q})$ should have periodical extrema corresponding to the fence lines. Fig.6 shows $x(\boldsymbol{q})$ with the horizontal axis representing a distance $|q|$ between $\boldsymbol{q}$ and a certain point on l'. The peaks at even interval represent the fence line on the image.

To find the interval, x_a, the autocorrelation of the accumulation $x(\boldsymbol{q})$ along the curve V, is calculated so that the interval $x(\boldsymbol{q})$ is represented as a peak in the autocorrelation x_a as shown in Fig.7. The distance f_2 between the origin and the peak corresponds to the interval of $x(\boldsymbol{q})$, and the peak is searched in a certain range where $|q|$ is greater than δq; here $\delta q = 10$ is used.

Then the extrema in $x(\boldsymbol{q})$ are searched by using f_2. First the maximum in $x(\boldsymbol{q})$ is transfered into the parameter space $\boldsymbol{p}_2$. Then the next extremum is searched around the area f_2 away from $\boldsymbol{p}_2$ along V. Since extrema may not be on the curve V due to noise, each extremum is searched in a $\delta\theta \times \delta\rho$ rectangle $nf_2 (n = \pm1, \pm2, \ldots)$ away from $\boldsymbol{p}_2$ (Fig.8).

The procedure above is applied to each of the two vanishing points, then the extrema are projected backward onto an image as lines as shown in Fig.10(b).

3.3 Disoccluding fence lines

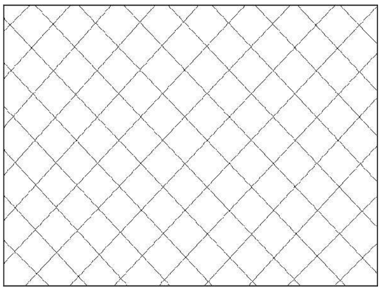
Now we can define areas to be disoccluded based on the fence lines in the image. Here the simplest way is used; just expanding the estimated lines drawn on the image (such as Fig.9(b) and Fig.10(b)). Then a disoccluding processing is performed to obtain a scene without a fence. In experiments shown in the next section, we implement the PDE based inpainting method [1].

4 Experimental results

We have tested the proposed method on twenty real images taken by a digital camera. Each image has an uncontrolled background and a fence which is closer to the camera and has no bend nor defect. In ten images a fence is parallel to the camera, and another ten images for the others. The images are of 640×480 pixels and processed within a minute by a linux-based PC with C++.

Fig.9(a) shows an image where a fence is parallel to the image plane, and estimated lines by the method in Sec.2 are shown in Fig.9(b). Almost all lines are extracted except a small segment

(a)

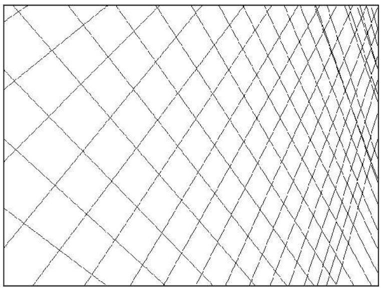


(b)

(c)

Fig. 9: Result for parallel case. (a) original image. (b) estimated fence lines. (c) fence removal result.

(a)



(b)

(c)

Fig. 10: Result for not parallel case. (a) original image. (b) estimated fence lines. (c) fence removal result.

Fig. 11: Some experimental results of line extraction.

in the left-bottom corner in the original image because of its small amount of accumulations in the Hough parameter space. The estimated lines are superimposed on the original image in Fig.12 in a large scale. The lines are not exactly on the wires of the fence, but the expansion before disocclusion can cover the difference and regions of the wire have been filled in. The disocclusion is shown in Fig.9(c). This result seems to be good for visual inspection, however, the performance depends highly on the disocclusion method used.

The images in Fig.10 show the result on the case where a fence is not parallel but inclined to the image plane. At the right-bottom area in Fig.10(b), lines are not extracted because the distribution of the lines are so dense that the accumulation has no prominent peaks around there. The twist of the wire in the three-dimensional space also affects the assumption that the fence can be approximated by straight lines. We can see the defect that some parts of the wires of the fence still remain Fig.10(c) because the assumption is also broken as the fence comes close to the camera. This may be reduced by more expanding the estimated lines to cover all the wire, but visible parts of the background will be less preserved.

Some line extraction results shown in Fig.11 demonstrate the robustness of the proposed method, and suggest that the method for not parallel case has to be improved. Note that the parameters appeared in the description of the proposed method are used without any modification for all twenty images.

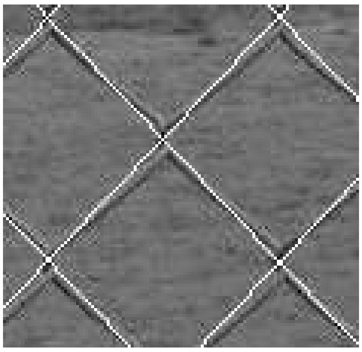


Fig. 12: Estimated lines superimposed on the image Fig.9(a).

5 Conclusions

In this paper, we have proposed a method to extract lines of a fence in the image, and remove them to recover the background occluded by the fence. By assuming that parallel lines in a space form a fence, its projection onto the image plane is modeled by two cases; the fence is parallel and not parallel to the image plane. Two difference methods for each case have been proposed, and experimental results shows the robustness of the proposed method and illustrated that the background scene occluded by a fence was recovered by disoccluding the line regions of the fence. The proposed method involving a lot of parameters to be tuned, however, the meanings of them are intuitively and easy to modify for a specific image. The proposed method does not yet judge whether a fence given in an image is parallel or inclined, and the automatic selection or the integration of two methods are the future work.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. of SIGGRAPH2000*, pp. 417–424, 2000.

[2] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, and G. Sapiro, "A variational model for filling-in gray level and color images," in *Proc. of ICCV2001*, vol. 1, pp. 10–16, 2001.

[3] T. Amano, A. Yamaguchi, and S. Inokuchi, "Image interpolation using bplp method on the eigenspace," *The IEICE Transactions on Information and Systems*, vol. J85-D-II, no. 3, pp. 457–465, 2002. http://hilbert.elcom.nitech.ac.jp/~amano/frames/demo.html#BPLP_Inter_e.

[4] S. Masnou and J.-M. Morel, "Level lines based disocclusion," in *Proc. of ICIP'98*, vol. 3, pp. 259–263, 1998.

[5] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing.* Springer, 2002.

[6] M. Haindl and J. Filip, "Fast restoration of colour movie scratches," in *Proc. of ICPR2002*, vol. 3, pp. 269–272, 2002.

[7] C. Wu, C. Liu, H. Y. SHum, Y. Q. Xu, and Z. Zhang, "Automatic eyeglasses removal from face images," in *Proc. of ACCV2002*, pp. 193–198, 2002.

[8] C. W. Lee, K. Jung, and H. J. Kim, "Automatic text detection and removal in video sequences," *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2607–2623, 2003.

[9] M. Kuramoto, A. Yamashita, T. Kaneko, and K. T. Miura, "Removal of waterdrops in images by using multiple cameras," in *Proc. of MVA2002*, 2002.

[10] A. Almansa, A. Desolneux, and S. Vamech, "Vanishing point detection without any a priori information," *IEEE Trans. on PAMI*, vol. 25, no. 4, pp. 502–507, 2003.

Suzuki, Hiroshi: received his B.E. in information engineering from Nigata University, Japan, in 2003. He is a graduate student at the Graduate School of Science and Technology, Niigata University, Japan.
Tamaki, Toru: received his B.E., M.S. and Ph.D degrees in information engineering from Nagoya University, Japan, in 1996, 1998 and 2001, respectively. Currently, he is a research associate at the Graduate School of Science and Technology, Niigata University, Japan.
Yamamoto, Masanobu: received his B.E. from Faculty of Engineering, Kyushu Institute of Technology in 1973, and M.S. from Graduate School of Science and Engineering, Tokyo Institute of Technology in 1975, and Ph.D degrees in engineering from Tokyo Institute of Technology in 1988. Currently, he is a professor at the Department of Information Engineering, Faculty of Engineering, Niigata University, Japan.