

Evaluation of Boundary Detection Methods Based on a Pair of Images

Khalid A. Al-Shalfan

Allmam University, Riyadh, Saudi Arabia
kshalfan@ccis.imamu.edu.sa

ABSTRACT: Local discontinuities in image intensity fall into three categories: points, lines, and edges. Discontinuous in image amplitude are important primitive characteristics of an image that carry information about object borders. Detection methods of image discontinuities are principal approaches to image segmentation and identification of objects in a scene. This leads to locating intersection boundaries that represent a 3D point in the scene correspond to an object point. In this paper some of edge detect filters have been evaluated based on detecting image intersection points and then show the ones that correspond to polygonal object planes for a variety of types of subject. The proposed algorithm starts up with obtaining image corresponding object intersection points after been detected. Epipolar rectification then applied on the pair of images using the detected corresponding object intersection points. This technique is illustrated with a pair of uncalibrated images of several scenes, utilizing the epipolar constraint.

KEYWORDS: Boundary Detection, Image Segmentation, Edge Detect Filters, Image Processing.

I. INTRODUCTION

Local discontinuities in image intensity fall into three categories: points, lines, and edges. Discontinuous in image amplitude are important primitive characteristics of an image that carry information about object borders. A camera maps a world point on to a 2-D image point. Extracting features of mapped objects becomes a powerful means in computer vision for sensing the environment and has been widely used in object detection and robotics. The well-known method based on Edge detects filters which search for borders between different colors and so can detect local discontinuities in image intensity.

Most of deferent the edge detect filters are based on gradient calculation methods and give the border lines. Figure 1 represents color intensity variations, where the left is a slow color gradient which is not a border. While the right is a quick variation which is an edge. In order to the find out the color variation speed whit in an image, we need to calculate gradient of this edge. This can be done by applying one of deferent the edge detect filters, i.e. the first derivative. We have to decide that a border is detected when gradient is more than a threshold value as shown in Figure 2 (the exact border is at top of the curve, but this top varies according to borders). In most cases, threshold is under top and border is thick.

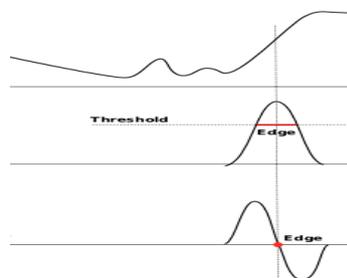


Figure 1: Gradient calculation methods.

One of the most used edge detect filters is the Laplacian edge detection which uses the second derivative as shown in Figure 3. It renders a thin border (one pixel wide) because the top of the curve is now at zero and clearly identified.



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

However, this derivative gives several zeros corresponding to small ripples, resulting in false edges. So image blurring need to be applied first before applying edge filters, which flattens small ripples in signal and so prevents false edges.

An object intersection point is a region in an image whose neighbourhood exhibits a significant two-dimensional structure. The fact that it is associated with a single point (and thus highly localized), while the nearby region has considerable 2-D structure, makes intersection points a better candidates for correspondence matching.

The detection of object intersection point has been looked at in the field of mobile robot navigation research. Yeung and et al. [1] investigate corner-based fixation and direction control for docking a fixated object using only visual parameters, specifically, the rotational component of log-polar optical flow. Detecting object intersection point has been tackled in different ways in the literature. Points lying on a single 3-D plane in the scene can be detected based on the approach that exploiting quantities that remain unchanged under perspective projection and can be directly computed from the employed image features.

Such quantities that are invariant under perspective viewing are referred to as projective invariants [2]. Meer et al. [3], for example, employed projective and permutation invariants to obtain representations of coplanar point sets that are insensitive to both projective transformations and permutations of the labelling of the set. Lourakis [4] proposed a method for determining the correspondence of two sets of coplanar points and lines. The method exploits results from projective geometry and determines the correspondence between images that are related by an arbitrary projective transformation. The problem of extracting the intersection points of a polygonal object plane can be also tackled utilizing the epipolar constraint, as it is proposed in this paper.

II. DETECTION OF OBJECT INTERSECTION POINTS

Many approaches have been developed to extract image intersection points and determine their associated attributes. One approach is boundary based intersection point detectors, where the image is segmented before extracting the boundaries of the objects and expressing them as chain code. Intersection points are then detected along the boundaries. The main weakness of this approach lies in the lack of reliable methods of image segmentation [5] as presented by Mokhtarian and Suomela [6]. Another approach is based on the differential analysis of the raw grey-scale image [7]. This can be either template based [8] or gradient based [9]. The template-based method detects the similarity between a given template at a specific angle and the image data in a sub window to find the intersection points. The use of templates at multiple orientations makes the method computationally expensive. The gradient based detectors rely on measuring the curvature of an edge that passes through a neighbourhood. The strength of the intersection point response depends on the edge strength and the edge direction rate of change.

These methods are more sensitive to noise than the template methods. Luo and Hancock in [10] presented a intersection point detection method that exploits some features of the template and the curvature based methods. In particular, they aimed to detect intersection points via direct topographic analysis. An intersection point detector technique that defines an intersection point as two half edges and uses a more recent edge detection approach based on the derivatives of Gaussian smoothing operators is given in Mehrotra and Nichani [8]. Another approach called SUSAN (Smallest Univalued Segment Assimilating Nucleus) proposed by Smith and Brady in [11] allows image edges, lines and junctions to be accurately and quickly found. It is based on the summation of pixels within a region and is less sensitive to noise than the gradient based methods. Visual inspection of the results showed that the SUSAN method compared to the others was best with regard to accuracy and fastness. Consequently this approach was adopted in the author's system for intersection point's detection and correspondences. The rest of this section describes the implementation of the SUSAN intersection point detector.

2.1. SUSAN Operator

It is important to apply a reliable intersection point detection procedure, because noise tends to cause false intersections. A feature of the SUSAN intersection point detector is its ability to find intersection points by direct analysis of the image intensities rather than their derivatives, thus removing the need for smoothing to reduce noise. This makes it faster than most of the other intersection point detectors, and it also provides better localization of the intersection points regardless of the mask size used. The basis for the SUSAN principle is the use of an approximately circular mask with an area of n_{max} pixels and the nucleus at an intersection point feature, to give an isotropic response, moving over each pixel in the image as the center or nucleus of the mask. The intensities are said to be similar when

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

their difference is less than a defined threshold t . This threshold can be used to control the quantity of the output without significantly affecting the quality of the results. The response function used in comparing intensities is

$$c(\mathbf{r}, \mathbf{r}_0) = \exp\left(-\frac{I(\mathbf{r}) - I(\mathbf{r}_0)}{t}\right)^6 \quad (1)$$

Where, \mathbf{r}_0 is the position of the nucleus in the image, \mathbf{r} is the position of any other point within the mask, $I(\mathbf{r})$ is the intensity of any pixel, and t is the difference threshold. With the response function of equation (1), the area of the SUSAN with nucleus \mathbf{r}_0 is

$$n(\mathbf{r}_0) = \sum_r c(\mathbf{r}, \mathbf{r}_0) \quad (2)$$

For there to be an intersection point at \mathbf{r}_0 , $n(\mathbf{r}_0)$ must be less than half of its maximum possible value n_{max} . The number of pixels $n(\mathbf{r}_0)$ is compared with a geometric threshold g , normally set to $n_{max}/2$. This gives a decision function.

$$R(\mathbf{r}_0) = \begin{cases} g - n(\mathbf{r}_0) & \text{if } n(\mathbf{r}_0) < g \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The function is non-zero at all candidate points for intersection points. Reducing the geometric threshold g restricts the candidate points to those with sharper angles. The center of gravity of the SUSAN is calculated at each candidate point, according to the following formula:

$$\bar{r}(\mathbf{r}_0) = \frac{\sum_r r c(\mathbf{r}, \mathbf{r}_0)}{\sum_r c(\mathbf{r}, \mathbf{r}_0)} \quad (4)$$

For any candidate point to be a genuine intersection point, \mathbf{r} must be distinct from \mathbf{r}_0 . Thus candidate points are rejected if $|\mathbf{r} - \mathbf{r}_0| < 2$. As a final test, all the points lying on the line from the nucleus \mathbf{r}_0 , through the center of gravity \mathbf{r} , to the edge of the mask are tested. If all these pixels are in the SUSAN, the nucleus is accepted as a intersection point; otherwise it is rejected.

III. MATCHING THE DETECTED INTERSECTION POINTS

Automatic matching of features from stereo is a difficult task for computer-based image analysis [12]. Most existing approaches to matching intersection points in pairs of images fall into two main classes:

Correlation-based stereo, which is known as area based stereo, uses correlation among brightness patterns in the local neighbourhood of a pixel in one image with brightness patterns in a corresponding neighbourhood of a pixel in the other image. The area of search is restricted by the possible displacement, which depends on the angle between the two views and the maximum roughness of the surface. The area based techniques have a disadvantage in that they use intensity values at each pixel directly, and are hence sensitive to distortions as a result of changes in viewing position (perspective) as well as changes in the absolute intensity, contrast and illumination. Also, the presence of occluding boundaries in the correlation window tends to confuse the correlation-based matcher.

Feature based methods those use symbolic features derived from the intensity images rather than the image intensities themselves. Hence, these systems are more stable towards changes in contrast and ambient lighting. Also feature based methods allow for simple comparisons between attribution of the features being matched, and are hence faster than template matching methods [12]. Feature-based methods have some advantages over intensity-based correlation. Feature-based methods are less ambiguous since the number of potential candidates for correspondence is smaller. The resulting correspondence is also less dependent on photometric variations in images. In this work, both techniques were used. The feature based approach was used to match the initial correspondences detected as features for estimating the epipolar lines in the eight-point algorithm [13] while the correlation based approach was used to obtain the corresponding intersection points.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

This section describes the matching of intersection points detected using the SUSAN algorithm, discussed in the previous section, to obtain a set of correspondences. The correspondences are used with the eight-point algorithm [13] to find the fundamental matrix in order to rectify the images. Two lists of image intersection points are available for the matching procedure. The list for the first image contains $Nimg1$ number of intersection points, and the list for the second image contains $Nimg2$ intersection points. Intersection points in the first image are considered the reference correspondences and matched with intersection points in the second using normalized cross-correlation. This done by calculating the Pearson correlation coefficient for a region about the point (x, y) in the first image and (x',y') in the second image using this formula:

$$C_{coeff}(x',y') = \frac{\sum_{i=-\frac{I}{2}}^{\frac{I}{2}} \sum_{j=-\frac{J}{2}}^{\frac{J}{2}} ((g(x+i,y+j) - \bar{g})(g'(x'+i,y'+j) - \bar{g}'))}{\sqrt{\sum_{i=-\frac{I}{2}}^{\frac{I}{2}} \sum_{j=-\frac{J}{2}}^{\frac{J}{2}} (g(x+i,y+j) - \bar{g})^2 \cdot \sum_{i=-\frac{I}{2}}^{\frac{I}{2}} \sum_{j=-\frac{J}{2}}^{\frac{J}{2}} (g'(x'+i,y'+j) - \bar{g}')^2}} \quad (5)$$

where, g and g' denote intensities in the first image region and the second image region respectively; \bar{g} and \bar{g}' denotes the average value of g within the first image region, and g' in the second image region respectively; C_{coeff} ranges from -1 to +1 with 1 indicating the best match and -1 the worst; I and J are width and height of the correlation window. The matching procedure first of all takes the intersection points in the first image in the order they appear in the list and using a square correlation window cross correlates each with the intersection points listed in the second image in the order they appear. For each reference intersection point, the intersection point with the highest correlation coefficient is stored as the initial match. The final numbers of NO match matched intersection points are stored in a new list for further checks. Intersection points in the first image that has no match in the second image, because they are not visible in the second image, and vice versa, should have been eliminated from the list. The next step is to perform a geometrical check of the matched intersection points. The relative location of every point in the first image with respect to the neighbouring intersection points is determined by calculating distances and angles. The same calculations are made for the corresponding match, and this will be accepted as a true match if it has the same relationship with its neighbours. Best results are achieved with the eight-point's algorithm [13] if the set of correspondences used for the calculations are widely spread over the whole images. So the spacing between the matched intersection points is checked to be larger than an adjustable pre-defined distance, starting with matches having higher correlation coefficient value.

IV. IMAGE RECTIFICATION

The practical importance of epipolar geometry stems from the fact that the plane identified by P , O_l and O_r which is called an epipolar plane, intersects each image plane in a line, called an epipolar line, as shown in Figure 2. The images of an object point P on the image planes p and p' lying on corresponding epipolar lines. This important fact is known as the epipolar constraint. Image rectification is the process of generating quasi-stereo images from a pair taken from arbitrary camera positions. This is achieved by transforming each image of the pair to make all the epipolar lines parallel to the horizontal image axes, and to make corresponding epipolar lines collinear as shown in Figure 2. Received methods of rectification either require some knowledge of the camera calibration [14] or involve some decision-making to determine the optimal reformation [15].

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

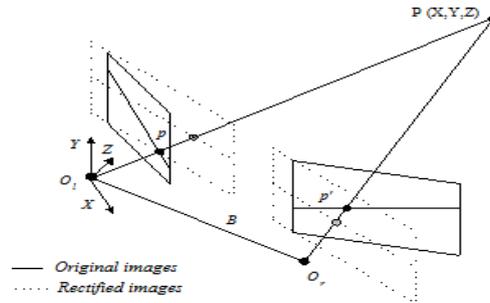


Figure 2: Epipolar rectification of a stereo pair of images.

To reconstruct the epipolar geometry without any information about the camera parameters, sufficient pairs of corresponding points in pixel coordinates are needed to estimate the fundamental matrix. There are a number of relatively complicated methods available to compute the fundamental matrix, but Hartley [15] gave a simple and stable method of deriving the fundamental matrix based on the eight-point algorithm proposed by Longuet-Higgins [13]. Considerable effort has been devoted to performing image rectification with general stereo geometry, but the work in this paper is based on an algorithm called Direct Rectification of Un-calibrated Images (DRUI) developed by Al-Shalfan et al. (2000) because it is simple, fast and direct [16]. The DRUI algorithm provides a general and unambiguous method for the rectification of stereo pair of images from an uncalibrated camera. In addition this algorithm directly estimates the fundamental matrix without knowledge of the camera calibration using the eight-point algorithm with number of corresponding intersection points.

V. OBJECT INTERSECTION POINT DETERMINATION

Based on the epipolar constraint, an intersection point of a polygonal object viewed in stereo images should have the same vertical co-ordinate in the left and right hand rectified images. The list of corresponding object intersection points produced in the previous sections is checked by this criterion. Any cases showing a significant vertical discrepancy are labeled as an object intersection point. Of the remainder, there is a difficulty when one edge of the object point is nearly horizontal in the rectified image, since it is then impossible to determine whether this intersection point represents points of a polygonal object or not. The remaining cases are taken to be object points.

VI. EXPERIMENTAL RESULTS

The system and each of its individual stage has been tested on different stereo image pairs, and the results show the method can usually determine object intersection points successfully. In this experiment a pair of 1524 by 1012 pixel uncalibrated images of an assembled artificial architectural scene were captured with a digital camera from viewpoints. The images are shown in Figure 3, and it is apparent from visual inspection that they contain a number of object corners of different polygonal planes due to the nature of the scene, which makes it ideal for such system. The objective was to test the effectiveness of the algorithm on stereo pair with a large shift between the two images.

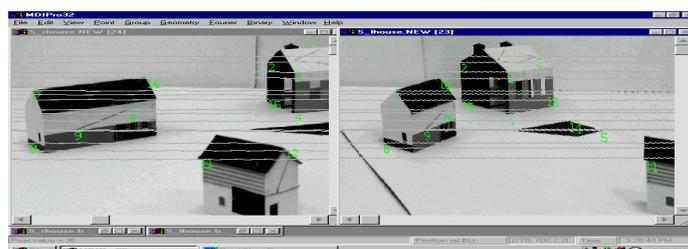


Figure 3: Rectified images of assembled artificial architectural scene.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

The numbered intersection points maps for the two rectified images in Figure. 3 are the result of applying the SUSAN detector described in Section 2. For the results, a circular mask of radius 3.4 pixel containing 37 pixels was used as suggested by Smith and Brady [11] with the intensity threshold t set to 25. Most of the prominent intersection points were detected successfully with high accuracy and with a strong feature attributes and located very precisely. Intersection points were matched on the basis of calculated the Pearson correlation coefficient as explained in Section 3. The minimum spacing separating the accepted matched points was set to 20 pixels to ensure an adequate spread of matched points over the images. From a collection of 19 detected intersection points in the left image and a collection of 18 in the right image, there are 16 correspondences which were successfully matched. Visual inspection verified all matches were correct. There are 11 object intersection points have been identified and the results are shown in Figure 3 and presented in Table 1.

The two images shown in Figure 3 were rectified on the base of 16 pairs of corresponding intersection points. The rectification algorithm DRUI depends on the estimated fundamental matrix. In order to measure the quality of the estimated fundamental matrix, the mean squared perpendicular distance between each point and the corresponding epipolar line is calculated. Mismatched correspondences are expected to have the largest squared perpendicular distance values. The average root-mean-square perpendicular distance between point and corresponding epipolar line is 0.10 pixels. The epipolar lines corresponding to the intersection points are superimposed on the images. The identified intersection points that belong to polygonal surfaces are listed in Table 1, with an indication of their location in the right and left images.

intersection point No.	Left Image		Right Image	
	The coordinates of the detected object intersection points		The coordinate of the detected object intersection points	
	x	y	x	y
1	326	239	241	238
2	296	250	180	241
4	326	192	244	197
6	141	199	159	188
7	42	219	89	198
8	37	164	72	141
9	81	174	129	161
11	217	142	408	176
12	313	149	198	197
15	295	207	182	202
16	157	233	149	219

Table 1: The coordinates of the identified object intersection points in left and right images.

VII. CONCLUSION

In order to identify the object intersection points, pairs of images of the scene have been considered and the image intersection points of a rectified stereo pair were extracted. Intersection point's detection was done using the SUSAN algorithm. The steps described in the intersection point's detection stage were applied successfully to each frame of the rectified image pairs, resulting in a list of the most significant (object) intersection points with associated attributes for each rectified image. Some intersection points were lost due to changing scene and camera geometry. Further operations were also included in order to obtain corresponding image corers. In the matching stage, sets of intersection points (with their associated geometrical and photometrical attributes) from left and right rectified images were compared and assigned as matching pairs if the attributes were sufficiently similar and certain constraints satisfied. Finally, the decisions about whether the intersection point is belong to an object plane or not were made based on the epipolar constraint obtained from the DRUI algorithm. This vision system has been tested using different types of scenes including outdoor and indoor scenes. The practical examples presented in the paper demonstrate the success of the techniques both individually, and collectively as an overall method for obtaining the object intersection points. In practice, many difficulties are encountered. Some parts of the scene may be occluded. Perspective distortion,



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 1, January 2017

transparency and changes in illumination can all cause the appearance of an object to vary, making it difficult to match its images in the two views.

VIII. REFERENCES

- [1] Y Albert, B Nick, Efficient Active Monocular Fixation Using the Log-Polar Sensor. International Journal of Intelligent Systems Technologies and Applications, Australasia 2005; 1: 157-173.
- [2] ML Joseph, Z Andrew, Geometric Invariance in Computer Vision MIT Press Cambridge Massachusetts, 1992.
- [3] M Peter, L Reiner, et al. Efficient Invariant Representations, International Journal of Computer Vision 1992; 26: 137-152.
- [4] LA I Manolis, H T Spyros, et al. Matching Disparate Views of Planar Surfaces Using Projective Invariants, Image and Vision Computing 2000;18: 673-683.
- [5] K Jack, P J Stephen, Corner Detection for Chain Coded Curves, Pattern Recognition 1995; 28: 843-852.
- [6] M Farzin, S. Riku, Robust image corner detection through curvature scale space, IEEE Trans. Pattern Analysis and Machine Intelligence 1998; 20: 1376-1381.
- [7] DM Tsai, Boundary-based corner detection using neural networks, Pattern Recognition 1997; 30: 85-97.
- [8] R Mehrotra, S Nichani, et al. Corner Detection, Pattern Recognition 1990; 23: 1223-1233.
- [9] S Ajit, Grey Level Corner Detection - a Generalization & a Robust Real-Time Implementation, Computer Vision Graphics and Image Processing 1990; 51: 54-69.
- [10] B Luo, A D J Cross, et al. Corner detection via topographic analysis of vector potential, Proc. 9th Int. Conf. on British Machine Vision 1998; 20: 635-650.
- [11] SM Stephen, J M Brady, SUSAN-A New Approach to Low Level Image Processing, International Journal of Computer Vision 1997; 23: 45-78.
- [12] C Gonzalez Rafael , E Woods Richard, Digital image processing, 2nd edn. Prentice Hall, Upper Saddle River, NJ, 2002.
- [13] HC Longuet-Higgins, A computer Algorithm for Reconstructing a Scene from Two Projections, *Nature* 1981; 293: 133-135.
- [14] F Andrea, T Emanuele, et al. Rectification with unconstrained stereo geometry. British Machine Vision Conf. (BMVC97), BMVA Press, Essex, UK, 1998, 400-409.
- [15] H Richard Z Andrew, Multiple view geometry in computer vision, (Cambridge University Press, New York, 2000).
- [16] KA Al-Shalfan, J G B Haig, et al. Ipsn, Direct Algorithm for Rectifying Un-calibrated Images, Electronics Letter 2002; 36: 419-420.