ISPACS 2006

2006 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS2006)
Yonago Convention Center, Tottori, Japan

Block-based Vanishing Line and Vanishing Point Detection for 3D Scene Reconstruction

Yi-Min Tsai, Yu-Lin Chang, and Liang-Gee Chen

DSP/IC Design Lab,
Department of Electrical Engineering and Graduate Institute of Electronics Engineering,
National Taiwan University, Taipei, Taiwan, R.O.C.
Email: lgchen@video.ee.ntu.edu.tw

***Abstract*— This paper presents a robust and reliable block-based method for vanishing line and vanishing point detection. The proposed method provides assistance in construction of a depth map, which is a necessity of 3D scenes. Moreover, the method focuses on the fundamental image structural element analysis, and divides them into six successive steps. Furthermore, complicated mathematic calculation and approximation are replaced by six block-based estimation algorithms. Also, the method is a regular algorithm design which is suitable for VLSI design on future 3D applications.**

1. Introduction

In nowadays, the 3D image processing has become a trend in the related visual processing field. However, for a two dimension (2D) to three dimension (3D) conversion, depth map is constructed by estimating relative depth for each pixel, and thus there exist an assigned value for each point in a depth map. In a 3D video generation flow, depth map plays an important role, such as the second step in Fig. 1. The depth map gives the audience stereo vision by the third step – Stereo video from depth, such as "Depth Image Based Rendering [1]."

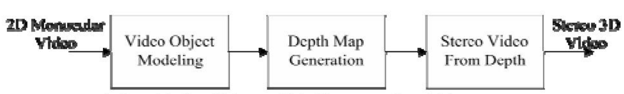


Fig. 1 The 2D-to-3D Conversion Diagram

S. Battiato et al summarizes the of generation of a depth map [2], which is based on several steps, including gradient planes generation, depth gradient assignment, consistency verification of detected region, and finally depth map generation. Vanishing point (VP) [3] is the point which possesses the furthest depth in an image, and things "converge" at this point through human brain. In other words, a vanishing point is the intersectional point of the vanishing lines in the image plane. As for the depth gradient assignment in the S. Battiato's work[2], vanishing line construction and vanishing point extraction are the primary works.

Moreover, with the help of vanishing line and vanishing point detection, image classification are processed easier according to vanishing line distribution, and then such a classification is utilized to construct stereo 3D scene with reliability. In other words, it provides a general idea of image space structure. 3D stereoscopic image pairs are then generated form depth map with S. Battiato's method [2].

The vanishing line and vanishing point detection can be divided into two kinds. One is the image-based method, and the other is the characteristic-based method. Image-based methods depend on the relation between object's position and characteristics in an image. For instance, a VP is assumed to locate in the middle of an image, which is a central point, the closer the objects are toward the central point, the smaller the objects are shown on the image. Hence, collecting information related with objects and the relationship may determine the trend of the vanishing line toward the vanishing point. Even so, image-based method [3] is not robust enough on finding the vanishing point because the estimated results still depends on the numbers of iterative training and adjustment, and acceptable results consume enormous time. Furthermore, erroneous detections may occur with high probability due to vague object relation such as sparse object distribution, lack of objects components in an image.

Another kind is the characteristic-based method [4][5] that mostly depends on the mathematic model analysis such as probability distribution analysis model or geometric mapping analysis model. With correct mathematic modeling, it describes the position formulas of a vanishing point. However, characteristic-based methods usually need complicated mathematic calculation approximation steps with tedious work.

A novel block-based vanishing line and vanishing point detection algorithm is proposed in this paper. The proposed method will be described in the section 2. And the experiment results will be shown in the section 3. Finally, section 4 gives the conclusion of the methods.

2. The Proposed Method

Our proposed method provides a simple but reliable method for VL and VP detection. We primarily focus on the fundamental image structure analysis instead of mathematically sophisticated estimation. An image object, which is an element of image content, is composed of points (pixels in images) and lines basically. Fundamentally, the line sections which compose of the image contents are edge segments. VLs are the extended edges in a sense, and a VP is usually on the extended edge but not on the edge of objects. In addition, a VP is also regarded as the intersectional point of VLs. Accordingly, we would focus on the detection of "lines" and find out essential vanishing lines of image structure and then a vanishing point can be detected from these vanishing lines easily and robustly. However, because of detection errors, VLs may not intersect at a point exactly; then a tolerable approximating area as a vanishing region (VR) is defined. And then, VR is composed of the intersectional points from vanishing line pairs. Similar to VP, a VR represents a region with the largest depth.

The proposed method is presented with multiple stages through the flow chart shown in Fig. 2 which contains procedures including edge detection, energy filter, block object finding, object combination and selection, dominant vanishing lines acquisition and VP detection. Most importantly, our method is a block-based algorithm rather than the pixel-based algorithms which are not as efficient and faster as the former. Our goal is to eliminate the disadvantages of the image-based methods and characteristic-based ones, and construct a more robust approach.

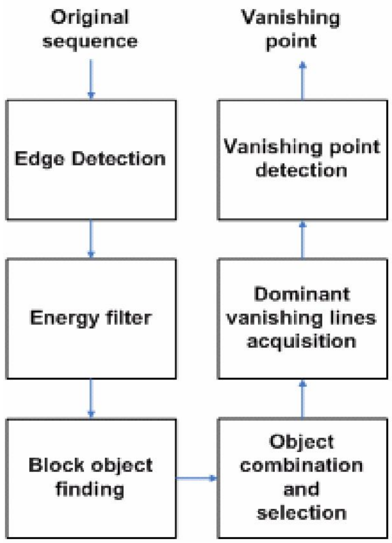


Fig. 2 The Flow Chart of the Proposed Method

In the following sections, these steps are discussed in detail:

2.1 Edge detection

In order to determine vanishing lines, from the previous concept of vanishing lines, the edge detection for images is a basic and significant step which is used to enhance the visualization of the edges. Robinson's detection operator [6] is preferred in the edge detection part. An adjustable threshold value is considered for optimization which maintains edge integrity and reduces noise interruption.

2.2 Energy filter

For edge segments detected by the edge detection process, there are two main forms of edge segments after processing. One kind is termed as "line-type" form, which is the necessary component for finding vanishing lines. And the other is termed the "mass-type" form, which is ineffectual due to its excessive pixels. The "Mass-type" segments cause line detection errors and increase the algorithm complexity. Thus, in order to eliminate the "mass-type" edge segments, a filter window is utilized, which is called the "energy filter" block with variable block size. For every pixel position (x, y) in a block, the total energy is represented as the following formula:

$$E = \sum_{x,y \in Block} I(x,y)^2 \qquad \textbf{(1)}$$

The total energy of a "mass-type" edge segment is higher than that of a "line-type" edge segment, so a threshold value is set up to make a judgment if an edge segment should be eliminated or not. The pseudo code of an energy filter is proved in the following codes:

```
//Energy filter
for (filter window traversed in the image){
  if (E>threshold)
    set I(x,y)==0;
  else    {
      // Keep the edge segment
  }
}
```

With the help of the proposed energy filter, blocks with high energy will be filtered out. And those blocks which have "line-type" edge segments would be left. After filtering, the image line structure is all composed of the low-energy "line-type" edge segments.

2.3 Block object finding

In order to catch all the edge segments in a filtered image, a square searching window with a searching block length and four block location points is utilized as shown in Fig. 3. We then define an edge segment within a searching window as a "block object." For a pixel at the middle position of an edge segment, we define it as a central reference point and we divide the searching window into eight searching parts around the central reference point, such as the black areas divided by the horizontal and vertical lines in Fig. 3. For each part, adjacent points which are connected with central reference point are marked pixel by pixel from the point closest to the reference point in outward orientation to the point farthest to the central reference point. Repeating the similar process from the upper left part in the clockwise orientation until all parts have been searched, and then a "small block object"(SBO) is constructed.

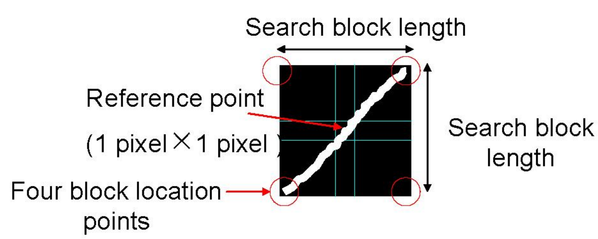


Fig. 3 Block Object Finding Diagram

A "small block object (SBO)" is a fundamental unit in the following step, and each SBO contains an edge segment with necessary information and characteristics. Linear regression line for an edge segment is thereupon calculated with regression line parameters β_1 and β_0 .The slope of the regression line is β_1 , and β_0 is the Y-intercept. A regression line for a particular edge segment block object can be represented as the following formulas:

$$y = \beta_1 x + \beta_0 \qquad (2)$$

where $\beta_1 = r \cdot \frac{S_y}{S_x} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$,

$$\beta_0 = \bar{y} - \beta_1 \cdot \bar{x}$$

$(x, y) \in$ edge segment within searching window

For each edge segment in SBOs, the linear regression line calculation is repeated identically, and the processing result is stored.

2.4 Object combination and selection

A VL passes through considerable amount of edge segments of objects in an image and carries essential information of an image structure. With the goal for determining the VLs, the SBOs whose characteristics are similar and correlative are combined into a "great block object"(GBO). Some constraints for the combination should be considered to make sure that two edge segments have similar regression line characteristics.

The first constraint shown in eq.4 depicts that the included angle θ between two regression lines of SBOs should be smaller than a threshold value angle $\theta_{threshold}$, and it can be represented in the form of trigonometric tangent as eq.3 and eq.4. For two regression line slopes m_1 and m_2, their relationship is shown as the following equation:

$$\tan\theta = \left|\frac{m_1 - m_2}{1 + m_1 m_2}\right| \qquad (3)$$

, where $m_1 m_2 \neq -1$

$$\tan\theta \leq \tan\theta_{threshold} \qquad (4)$$

A distance parameter d is defined as the distance from the intersectional point of two regression lines L_1, L_2 to the median of the two reference points corresponding to their "small block objects" with the regression lines L_1, L_2, as shown in Fig. 4. The second constraint is that the value of the distance d should be smaller than a threshold value distance $d_{threshold}$, which is depicted in the following equation:

$$d \leq d_{threshold} \qquad (5)$$

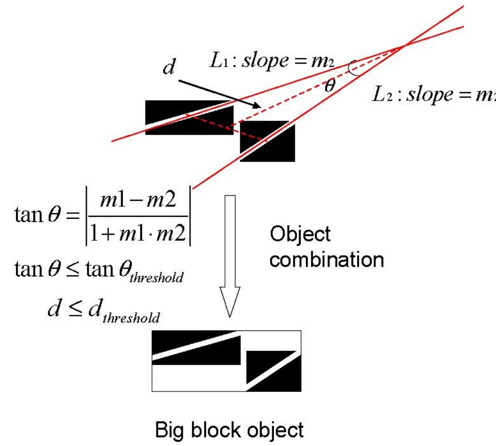


Fig. 4 The object combination scheme of two regression lines

After repeating the object combination process for all SBOs, the counts that each block object passes through the combination process is defined as N, and C_l is defined as the length between the upper left point and the upper right point within the searching window. C_w is defined as the length between the upper right point and the lower right point within a searching window.

Accordingly, a useful GBO selection is described according to the following inequality (6) (7):

$$N \geq N_{threshold} \qquad (6)$$

$$C_l^{\,2} + C_w^{\,2} \geq C^2{}_{threshold} \qquad (7)$$

This two constraints filter out the block objects which are of small size. Therefore, GBOs are left for further processing.

2.5 Dominant vanishing lines acquisition

After the object selection, there are several GBOs, and the regression lines of the GBOs are re-calculated. And then the vanishing lines which are approximately identical to these regression lines are obtained. However, for simplifying the future VP decision, few VLs with greater block object are selected to describe the image structure. The number of the selected VLs is about 2 to 5, depending on the complexity of image structure. These VLs are so called "dominant VLs". What comes after the dominant vanishing line acquisition is that we have to re-check the appropriateness of the dominant vanishing line by mapping it onto the original image. A real dominant vanishing line has numerous parts consistent with the edge segments.

2.6 Vanishing point detection

Finally, a VP is obtained by calculating the intersection point of each vanishing line pair. If the intersectional points are not all located at the same point, we then defined a subset as a vanishing region whose range covers all these intersection points. Surely, if the intersection points are precisely consistent in position, the meeting point is certainly a VP.

However, a VP usually appears on the extended line of edge rather than on the edge itself. Furthermore, a true vanishing point is acquired from at least two vanishing lines and they are line transformation from two originally parallel lines of actual scenes. These constraints are in the help of refining the correct vanishing point.

3. Experiment Results

Here we present two sequence demonstrations with suitable threshold value in each process stage, and one sequence is outdoor scene in Fig. 5 and the other one is indoor scene in Fig. 6. Fig. 5 shows the original sequence after vanishing line detection. Obviously, the vanishing point is out of the image, far away from the ship, and it may be considered as the right case. Fig. 6 shows the indoor sequence with vanishing line detection, which is a typical case with the vanishing point located at the middle of the image.

The testing platform is a PC with AMD Athlon 2200+ 1.8GHz, 512MB RAM, on Win XP, Visual C++ .NET. And the detection procedure for 300 frames runs in 194 seconds, that is to say, 1.564 fps. If with vector acceleration hardware such as GPU or ASIC design, the real-time computing frame rate can be easily raised since the proposed algorithm is a regular block-based design, which fits into the vector calculation perfectly.

The two results show that the proposed algorithm achieves correct vanishing point detection for its following applications such as the depth map generation. And the computing time is better than the mathematic model analysis.

4. Conclusion

We have presented a fundamental detection algorithm based on the structural components analysis with robustness, and it is especially suitable for images with distinct object edges. The proposed method for vanishing line and vanishing point detection provides direct analysis from image structure without complicated math calculation. It utilized the energy of the edges to determine the dominant vanishing lines. And then the vanishing points are figured out from the most dominant vanishing lines. The proposed method is feasible for a particular image sequence without prior temporal information and guarantees that dominant vanishing lines are detected correctly with high probability and accuracy. The proposed block-based algorithm which still holds the regular block data flow feature is much faster, simpler and efficient. The feature makes the design suitable for VLSI implementation and vector acceleration hardware computing. Finally, the observation directly through the sequence from human view points is consistent with the experiment results. As for the 2D-to-3D conversion procedure, the proposed vanishing line and point detection gives great help for the overall scene knowledge, and the conversion proceeds more easily.

References

[1] Wan-Yu Chen, Yu-Lin Chang, Shyh-Feng Lin, and Liang-Gee Chen, "Efficient Depth Image Based Rendering With Edge Dependent Depth Filter and Interpolation, " in 2005 IEEE International Conference on Multimedia and Expo (ICME 2005), Amsterdam, Netherland

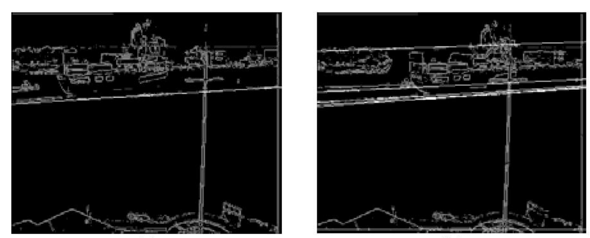
[2] S. Battiato, A. Capra, S. Curti, M La Cascia, "3D Stereoscopic Image Pairs by Depth-Map Generation", 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Proceedings. 2nd International Symposium on 6-9 Sept. 2004, Page(s):124 – 131.

[3] Virginio Cantoni, Luca Lombardi, Marco Porta, Nicolas Sicard, "Vanishing Point Detection: Representation Analysis and New Approaches", in the Proceedings of the 11th International Conference on Image Analysis and Processing, , Page(s):90 – 94., 26-28 Sept. 2001

[4] Jefferey A. Shufelt, "Performance Ealuation and Analysis of Vanishing Point Detection Techniques", IEEE transactions on pattern analysis and machine intelligence, VOL 21, NO. 3, March 1999.

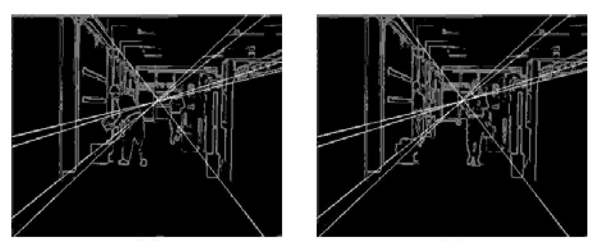
[5] Almansa, A.; Desolneux, A.; Vamech, S., "Vanishing Point Detection without Any A Priori Information", Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume 25, Issue 4, April 2003 Page(s):502-507.

[6] R. M. Haralick and L. G. Shapiro, Computer and Robot Vision, Vol. I, Addison Wesley, Reading, MA, 1992.



(a) (b)

Fig. 5 outdoor sequence-(a)1st frame;(b)275th frame



(a) (b)

Fig. 6 indoor sequence-(a)100th frame;(b)240th frame;