

Methods to Improve Stereo Matching Algorithm for Object Segmentation

JeongMok Ha*, Hong Jeong*, JunSeo Lee**

* Department of Electronic & Electrical Engineering, POSTECH (Pohang University of Science and Technology),
77 CheongAm-Ro, Nam-Gu, Pohang, Gyungbuk, Korea

** Department of Electronic & Communications Engineering, CNU (Chungnam National University),
99 Daehak-Ro, YuSeong-Gu, Daejeon, Korea

jmokka@postech.ac.kr, hjeong@postech.ac.kr, jsride@gmail.com

Abstract— Up to these days, there are huge amount of stereo matching algorithms and its improved versions. Not sufficient algorithms are proposed to be used in real systems like object recognition not considering time computing and complexity of the algorithm, but only for improving Pixel Error Rate (PER) in disparity image. We used some image processing methods for not only improving PER but also making disparity images good to be used for object segmentation. We used Trellis algorithm which is Dynamic Programming based stereo matching algorithm. Our proposed algorithm shows better results than original Trellis algorithm in qualitative and quantitative manner. Also this algorithm could be implemented in hardware system; it could work in real-time system.

Keywords— Stereo vision, Stereo matching, Object Segmentation, Trellis algorithm, Occlusion ignorance

I. INTRODUCTION

The desire of the human always wants to make human-like robot or computer that can understand and recognize object, situation, and more. For this reasons, computer vision has been studied for a long time. Based on this, the areas related to segmentation, tracking, recognition, and detection has been studied from many researchers.

Stereo vision is also well studied computer vision area. Unfortunately, the researches have been done only for improving quality of the depth map images. It is time to use stereo matching algorithm in another area which I already wrote above, and we need to mix them for practical usage.

With the development of the stereo matching algorithm, gesture recognition or pose estimation is also developed. We could use 3D data for finding motion not just for left, right, up, and down, but also forward and backward ([14], [15]). So from this moment, we could recognize all directional gestures.

In vehicle tracking, previous techniques only use 2D information, so it is not accurate and it makes errors several times ([12], [13]). If stereo matching is well improved, the distance between car and car is known exactly. It means that it is easy to use depth information in tracking parts, recognition, and automatic vehicle system.

To do recognition or tracking, object segmentation should be implemented before we use those techniques. Object

segmentation and background subtraction are basic techniques for other techniques. In our research we proposed some methods to improve stereo matching algorithm for object segmentation, such as block matching, left align for matching index, and occlusion ignorance. We explained original trellis algorithm first, after then we explained our proposed algorithm.

II. TRELLIS ALGORITHM

We used trellis algorithm to make real-time stereo matching system ([1]-[2]). Trellis algorithm could be changed into parallel architecture so it could be implemented in hardware system or GPU based programming so it can be implemented in real-time system.

Trellis algorithm is improved version of DP algorithm ([3], [5]). It separates matching node and occlusion node unlike DP algorithm. If matching node and occlusion node are divided into each other, the whole system becomes stable and left and right image pixels are matched well. That is the reason for choosing trellis algorithm.

Before we discuss our proposed algorithm, we should know about the original trellis algorithm, and how it works. In this section we deal with basic properties of the trellis algorithm and procedure of the algorithm.

A. Model of the Trellis Algorithm

Trellis algorithm uses raster scan-line procedure. Raster scan-line is a technique that extracts one line by one line. And the algorithm is applied independently in line by line. To use this raster scan-line technique we assume that left and right stereo images are well rectified. So we used rectified images for our experiments.

What we say “rectified” is the geometric that focal length of the left and right, the two image planes have coplanar, the optical axes of the two images are parallel, and the epipolar lines are the same. All those conditions are satisfied, we could start implementing trellis algorithm.

The purpose of the trellis algorithm is obtaining disparity between left and right matching points. Finding matching

point is key point in trellis algorithm and whole stereo matching algorithms. So in trellis algorithm it compares whole left image pixels and right pixels to find matching points. We used local line energy minimization to find the most similar pixels. In this case, the energy is defined as sum of the intensity difference between the left image and the right image.

If the height and width of the image are M and N , respectively, then the trellis algorithm is applied for M times and it compares intensities $N!$ times for one line. We defined left x -th line $\mathbf{f}_L(x)$ and right x -th line $\mathbf{f}_R(x)$ as

$$\begin{aligned}\mathbf{f}_L(x) &= [I_L(x,1) \ I_L(x,2) \ \cdots \ I_L(x,N)] \\ \mathbf{f}_R(x) &= [I_R(x,1) \ I_R(x,2) \ \cdots \ I_R(x,N)]\end{aligned}$$

where $I_L(x,n)$ and $I_R(x,n)$ are intensities of left and right images, x is the index of column, and n is index of row.

As we wrote above, trellis separates matching node and occlusion node. When we see trellis structure from left or right view, we couldn't see the occlusion nodes. However, in case of the center view, there is occlusion node which is different from matching node. Unlike matching node which is intensity difference between left and right image pixels, occlusion node has no information about the left and right image. Because trellis algorithm uses center projection model, it can be possible to separate them. If we separate them, the algorithm becomes stable. So we used this center-view trellis structure in trellis algorithm.

B. Energy Minimization Problem

As we wrote that trellis algorithm is used for solving local energy minimization problem. However, images which will be used for experiments are not clear enough to be used for trellis algorithm. That means there is noise in the images. So we assumed that there are noises and the noises are Additive White Gaussian Noise (AWGN). Images with this noise are defined by $\mathbf{g}_L(x)$ and $\mathbf{g}_R(x)$ which is corrupted version of the $\mathbf{f}_L(x)$ and $\mathbf{f}_R(x)$, respectively.

The purpose of the trellis algorithm is to estimate disparity between left and right images. Disparity $\mathbf{d}$ is estimated using Maximum A Posteriori (MAP) and estimated disparity is $\hat{\mathbf{d}}$ defined as,

$$\begin{aligned}\hat{\mathbf{d}} &= \arg \min_{\mathbf{d}} P(\mathbf{d} | \mathbf{g}_L, \mathbf{g}_R) \\ \hat{\mathbf{d}} &= \arg \min_{\mathbf{d}} P(\mathbf{g}_L, \mathbf{g}_R | \mathbf{d}) P(\mathbf{d}),\end{aligned}$$

where $P(\bullet)$ is the probability function and $P(\mathbf{g}_L, \mathbf{g}_R)$ term is removed by Bayes rules. Because there is exponential term, we used logarithm function which doesn't affect for energy minimization. The energy function is defined by

$$U(\hat{\mathbf{d}}) = -\log[P(\mathbf{g}_L, \mathbf{g}_R | \mathbf{d}) P(\mathbf{d})]$$

$$\begin{aligned}&= -\log[P(\mathbf{g}_L, \mathbf{g}_R | \mathbf{d})] - \log[P(\mathbf{d})] \\ &= U_c(\mathbf{d}) + U_p(\mathbf{d})\end{aligned}$$

where $U_c(\mathbf{d})$ and $U_p(\mathbf{d})$ are conditional energy function and prior energy function, respectively.

If we assume that energy functions are Gaussian function then we could get following easily,

$$U(\hat{\mathbf{d}}) = \sum_{i=1}^{2N} \left[\frac{1}{2\sigma} \Delta g(d_i) o(i + d_i) + \gamma \Delta d_i \right]$$

where γ is the real positive constant value for threshold and

$$\Delta g(d_i) = |g_L(a(d_i)) - g_R(b(d_i))|,$$

$$a(d_i) = \frac{1}{2}(i - d_i + 1),$$

$$b(d_i) = \frac{1}{2}(i + d_i + 1)$$

where $a(d_i)$ and $b(d_i)$ are left and right image indexes, respectively.

C. Procedure of the Algorithm

Trellis algorithm is Viterbi-based energy minimization algorithm. It has forward process and backward process to find path which makes energy function minimum. In forward process, this algorithm adds all intensity difference between left and right images and finds the minimum cost. In backward process, it finds optimal path which makes energy function minimum using computed minimum cost in forward process.

Whole procedure is based on what we wrote above, however, according to the sort of nodes, how to compute the node cost and find optimal path is different. Following 4 steps are whole procedure of the trellis algorithm. $U(i, j)$ is the accumulated costs where i is the site of the $U(i, j)$ and j is the disparity of the $U(i, j)$, and $P(i, j)$ is the optimum path.

First, in "Initialization" step, set up all cost to infinity value except for the first cost d_0 .

$$U(0, j) = \begin{cases} 0 & j = 0 \\ \infty & \text{o.w.} \end{cases}$$

Second, in "Recursion" step, compute the matching cost and find the minimum cost and the path of the each nodes for $i = 1$ to $i = 2N$.

In case of the matching node when $i + j = \text{odd}$,

$$\begin{aligned}U(i, j) &= U(i-1, j) + \Delta g(d_i) \\ P(i, j) &= j.\end{aligned}$$

In case of the occlusion node when $i + j = \text{even}$,

$$U(i, j) = \min_{\alpha \in [-1, 1]} U(i-1, j+\alpha) + \gamma$$

$$P(i, j) = \arg \min_{\alpha \in [-1, 1]} U(i-1, j+\alpha) + \gamma.$$

Third, in “Termination” step, finish computing and finding the best path

$$U = U(2N, 0)$$

$$\hat{d}_{2N} = P(2N, 0)$$

Fourth, in “Backtracking” step, find the optimum path for energy minimization.

$$\hat{d}_{i-1} = \hat{d}_i + P(i, j)$$

III. PROPOSED METHODS TO IMPROVE TRELLIS ALGORITHM

In this section we will suggest some methods to improve trellis algorithm for object segmentation. We used block matching for computing matching cost, we used left align system, and we used occlusion ignorance for making stereo matching system more convenient to be used object segmentation and background subtraction.

A. Block matching

The original trellis algorithm only compares one left image pixel and one right image pixel. Of course when left and right images are perfectly rectified to each other, it is possible to use only pixel by pixel matching. However, in real situation it is impossible to rectify stereo images perfectly to be used in real situation. So we should use not pixel by pixel matching but another method.

There are two kinds of solving this problem. One is improving rectification technique; another one is finding solution in other parts like trellis parts. We proposed block matching for overcome this problem. Pixel matching compares only one pixel to another one pixel. It means that pixel matching compares line by line independently so it is cause of streak noise.

In pixel matching case, matching cost $D(i, j)$ is calculated by

$$D(i, j) = |I_L(x, y) - I_R(x', y')|,$$

where i and j is site and disparity of the trellis structure respectively.

Block matching compares left image block and right image block. Block is group of the pixels so it means that block matching compares left pixel group and right pixel group.

In block matching case, matching cost $M(i, j)$ is calculated by

$$M(i, j) = \sum_{a=-n}^n \sum_{b=-m}^m |I_L(x+a, y+b) - I_R(x'+a, y'+b)|$$

where n and m are the width and height of block. Block could be both isotropic (when $n = m$) and anisotropic (when $n \neq m$). It depends on images size, resolution, the characteristic of the image, and so on.

Also it is possible to allow weight in block matching cost. If the distance a and b are nearer to (x, y) , weight function $w(c, d)$ is larger than far one.

B. Left Align for Matching Index

There are two ways to use depth information in recognition area, one is only using depth information for the feature, and another one is using both depth information and 2D information in the same time for the feature. Actually when we use only depth information, we assumed that depth information is very accurate enough to be used, that means that stereo matching algorithm is very powerful and robust in various situation. However, we used dynamic programming based algorithm, improved version of the trellis algorithm, it is not accurate enough but can be implemented real-time system. So we have to use both 2D information and depth information simultaneously.

To do that we need to align 2D information index and depth information index because trellis algorithm is center-referenced model which is different from index of the 2D images. We changed depth information index into 2D information index form. The index of the 2D information is (i, j) , so we changed depth information index into (i, d_j) , then we could change depth information index into 2D information index easily.

C. Occlusion Ignorance

Not like DP algorithm, trellis algorithm uses both matching nodes and occlusion nodes. Matching nodes are computed based on the left and right image intensity difference. On the other hand, occlusion nodes are computed using uncertain values so it makes the depth result poor.

For the stereo matching algorithm parts, it is very important to improve PER which means that making accurate depth map image for whole image parts. However, we are focused on the application of the stereo matching area. In our case we want to use depth image for object segmentation, it is not that important to improve the depth result. In this reasons we could ignore the occlusion parts.

DP-based algorithms have huge weakness which is called streak noise (also called stripe noise). We could remove most of them using occlusion ignorance because most of the streak noises are created because of the occlusion parts. And it is very effective for object segmentation. When we use DP-based algorithm, the maximum disparity difference between neighbour pixels is 1, so wrongly matched parts and the streak noise are placed mostly on boundary area. If we ignore

occlusion parts, not only we could remove streak noise but also we can segment objects more easily.

IV. EXPERIMENTAL RESULTS

We used Middlebury images which are widely used in stereo vision area. Those images are already rectified so we could only focus on not the rectification part but stereo matching parts.

We used Tsukuba, Map, Teddy, and Venus images for this experimental result. Of course we need check the PER of the original trellis and proposed algorithm, but we more focused on the usage of the object segmentation. How well it can be separated from the background or other objects is the most important part of our research. This fact is hard to convert into the numbers like PER. We could only depend on the quantitative manner.

The intensity difference between object that we want to segment and the other object or the background is larger than the original trellis algorithm. It means that it is easier to segment the object from the messed up images. When we use both 2D information and depth information, it is easy to separate object boundary because it uses depth information which have separable information.

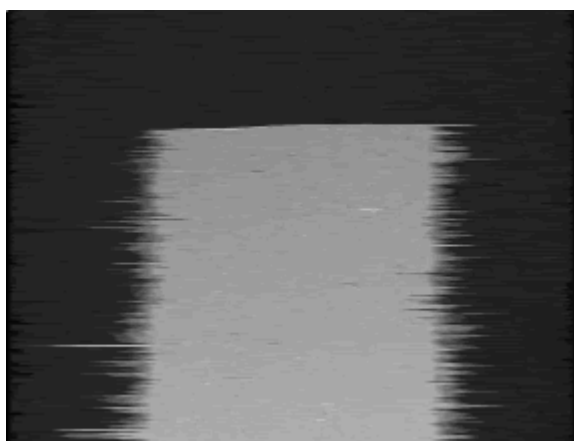


Figure 1. The map result image from the original trellis algorithm



Figure 2. The map result image from the left-aligned trellis algorithm

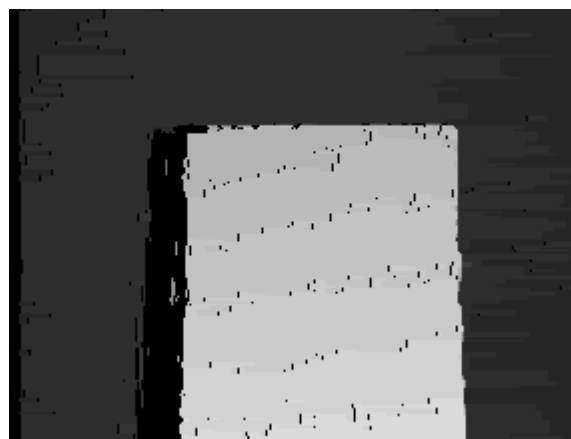


Figure 3. The map result image from the proposed algorithm

We showed just example of the map result images using original trellis algorithm (Figure 1), left-aligned algorithm (Figure 2), and the proposed algorithm (Figure 3). As we could see that original trellis algorithm makes too much streak noise and it is hard to segment the objects in the images, left-aligned algorithm makes index of the left image and index of the result image, and proposed algorithm makes more convenient for the segmentation.

It is better to use other algorithms like median filtering because there are many holes in one object. Hole makes segmentation difficult because the fact that the depth in one object is different is bad for segmentation. In future work, filtering is needed like median filtering.

V. CONCLUSIONS

There are not much stereo matching algorithms for real world application system. We used trellis algorithm for making robust real-time stereo matching algorithm. We used block matching, left align for matching index, and occlusion ignorance. We did our experiments using middlebury images which are widely used in stereo vision area. Our proposed algorithm shows more accurate results than the original one and it is better to use our proposed algorithm for object segmentation.

ACKNOWLEDGMENT

This research was supported by the Brain Korea 21 Project.

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the “Core Technology Development for Breakthrough of Robot Vision Research” support program supervised by the NIPA (National IT Industry Promotion Agency).

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the “IT Consilience Creative Program” support program supervised by the NIPA (National IT Industry Promotion Agency).

REFERENCES

- [1] Hong Jeong and Yuns Oh, "Trellis-based parallel stereo matching", In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, July 2000.
- [2] Hong Jeong and Yuns Oh, "Fast stereo matching using constraints in discrete space," *IEICE Transactions on Information and Systems*, September 2000.
- [3] Jason Selzer Cheng Lei and Yee-Hong Yang. "Region-tree based stereo using dynamic programming optimization." In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2378-2385, 2006.
- [4] D. Scharstein and R. Szeliski. "A taxonomy and evaluation of dense two frame stereo correspondence algorithms." *International Journal of Computer Vision*, 47:7-42, 2002.
- [5] B.T.Choi C.Kim, K.M.Lee and S.U.Lee. "A dense stereo matching using two-pass dynamic programming with generalized ground control points." In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, USA, 2005.
- [6] J.G.Kim and H.Jeong, "Parallel relaxation algorithm for disparity computation," *IEEE Electronic Letters*, 33(16):1367-1368, July 1997.a
- [7] Myron Z. Brown and Gregory D. Hager. "Advances in computational stereo." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):993-1007, Aug 2003.a
- [8] O. Veksler. "Stereo matching by compact windows via minimum ratio cycle." In *Proc. Intl Conf. Computer Vision*, 2001.
- [9] Takeo Kanade and Masatoshi Okutomi. "A stereo matching algorithm with an adaptive window: Theory and experiment." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920-932, Sep 1994.
- [10] H. H. Baker and T. O. Binford. "Depth from edge and intensity based stereo." In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 631-636, Vancouver, Canada, 1981.
- [11] Y. Ohta and T. Kanade. "Stereo by intra- and inter-scan-line search." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI 7(2):139-154, Mar 1985.
- [12] Chao, T., Lau, B., Park, Y., "Vehicle detection and classification in shadowy traffic images using wavelets and neural networks", In: *Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS*, SPIE Proc. Vol. 2902, pp. 136-147, 1996.
- [13] Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B., Russell, S., "Towards robust automatic traffic scene analysis in real-time". Vol.1, *ICPR*, Israel, pp. 126-131, 1994.
- [14] A. Azarbayejani, C. R. Wren, and, A. P. Pentland, "Real-time 3-D tracking of the human body", in, *IMAGE'COM 96*, Bordeaux, France, May 1996.
- [15] Q. Cai, and, J. K. Aggarwal, "Tracking human motion using multiple cameras", in, *International Conference on Pattern Recognition*, 1996.