

Enhancement of Trellis-Based Stereo Matching Algorithm Using Disparity of the Upper Scan-line

Yong Gyu Kang*, Won Suk Lee*, Hong Jeong*

* Department of Electronic and Electrical Engineering, Pohang University of science and technology,
790-784 san 31, Hyoja-dong, Nam-gu, Po-hang Gyungbuk, Korea.
ygkang@postech.ac.kr, wonsuklee@postech.ac.kr, hjeong@postech.ac.kr

Abstract— Stereo vision is the process of recovering 3D spatial information from a pair of 2D images. It is very difficult problem due to the fact that stereo matching problem tends to produce as large number of plausible solutions. Thus we need to restrict the solution space in some manner. Trellis-based stereo matching algorithm places hard constraints on the solution by considering the geometry of stereo imaging and by making assumptions about the real depth shape. In that algorithm, permitting the use of the highly parallel Viterbi algorithm (a special case of dynamic programming), real-time stereo vision system can be realized. However, by using a single scan-line for matching, there are too many stripe noises in the result. In this thesis, a new method which can reduce the stripe noise is proposed. Using the disparity information of the upper scan-line, the proposed algorithm achieves more exact disparity map. The proposed algorithm uses Average of Squared Difference (ASD) instead of Euclidean distance in calculating matching cost in order to alleviate the horizontal stripe noise. The experimental results show more precise disparity map compared with the existing algorithm, especially, at low-to-moderate SNR range. Random dot stereogram (RDS) is used to evaluate the quantitative experiment.

Keywords— Stereo matching, trellis, Dynamic programming

I. INTRODUCTION

In early vision, stereo matching has been one of the most important problems. The goal of stereo matching is reconstruction of 3D geometry from two images with different view point. Comprehensive survey is done by Scharstein and Szeliski [8] and by Brown and Hager [2] about stereo correspondence algorithms. In the taxonomy of [2], stereo matching algorithm can be categorized into two parts. One is local method and other is global method. A step of local method consists of matching cost computation, cost aggregation and disparity computation. In global methods, disparity optimization step is used instead of cost aggregation and disparity computation steps.

There are several matching cost calculation methods. Squared intensity differences (SD) and absolute intensity differences (AD) is the most common pixel based matching cost calculation method. Other methods are Normalized

cross-correlation (NCC), sum of-squared-differences (SSD), and binary matching cost.

Local and window-based methods aggregate the matching cost by summing or averaging over support region in cost aggregation step and simply choose the disparity at each pixel which has minimum cost value. To improve the result of window-based Local method, some techniques are used such as adaptive window [6] or shiftable window [9].

Global methods use bayesian methods which models discontinuities and occlusion. There are two methods depending on computation model; MRFs-based and dynamic programming (DP)-based. As the early work, DP is used by Baker and Binford [1] and Ohta and Kanade [7].

MRFs-based global matching methods get optimal disparity using optimization methods such as belief propagation or Graph Cut. These methods show good performance, but need much computation.

In classical DP method, the algorithm is suffered from "streaking" noise because optimization is done only in 1D scan-line structure. To solve this problem, several algorithms are proposed. For example, in 2-pass 1D DP approach [4], a 2D optimization is done enforcing piecewise smoothness constraint in the horizontal and vertical directions. In [3], advanced stereo algorithm is proposed which combines the strengths of region-based stereo and dynamic programming on a tree approaches using image segmentation such as mean shift algorithm.

The global methods show good result, but not much fast to get disparity results in real time. Therefore, Trellis-based stereo matching algorithm [10] is proposed to get disparity result in real-time. Trellis-based stereo matching algorithm suggests hardware architecture which is based DP algorithm. This algorithm is fast enough to get disparity image in real-time, but "streaking" noise problem is still remained. This paper suggests advanced trellis based algorithm to reduce "streaking" noise problem. In our paper, new method is proposed upper line scan method.

This paper is organized as follows. In Section II, we describe a brief review of the matching algorithm in form of trellis. In Section III introduces a technique to reduce stripe noise in DP. The experimental result is showed in Section IV. Finally, Section V concludes our algorithm and result.

II. TRELLIS-BASED STEREO

First, we brief the trellis-based stereo matching algorithm in [10]. Trellis-based stereo matching algorithm uses center referenced disparity geometry. Center-referenced disparity geometry has focal point in center of the focal points of left and right camera. Also, by new projection line, new nodes, called occlusion node, are created. Geometry of center referenced disparity is shown in Figure 1 and modified architecture in Figure 2. Filled circle is match node which denotes the correspondence of left and right point pixel, and white circle is occlusion node which represents the lack of matching point. In trellis algorithm, disparity changes only in occlusion node and matching node only represent data cost.

In the center-referenced disparity, there are some limitations to apply. First, disparity is larger than zero because of optical axis are parallel and zero disparity means infinite point. Second, only one disparity level change is allowed in occlusion node and, finally, the present and next stage's disparity level is same as past stage's level in matching node.

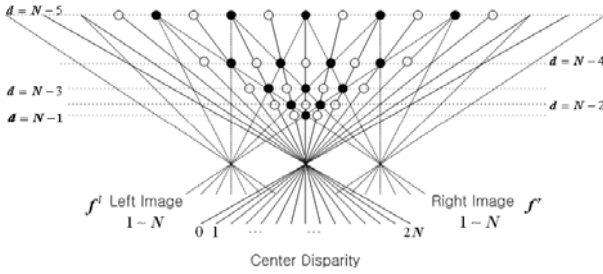


Figure 1. Center-reference disparity

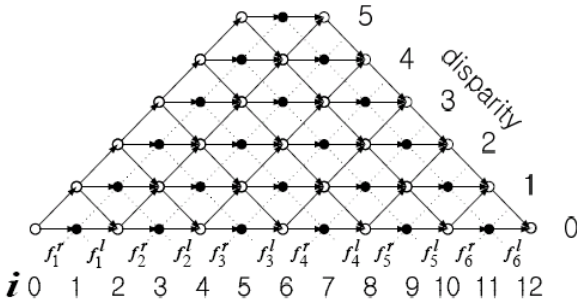


Figure 2. Modified trellis architecture of disparity

Trellis-based stereo matching is a maximum a posteriori (MAP) problem. Let g^l, g^r is camera image pair which pluses additive white Gaussian noise (AWGN). Estimation of disparity is represented as follows.

$$\begin{aligned}\hat{d} &= \arg \max_d P(d|g^l, g^r) \\ &= \arg \max_d P(g^l, g^r|d)P(d)\end{aligned}\quad (1)$$

By negative logarithm computation, (1) is represented as follow,

$$\begin{aligned}U_i(d) &= -\log\{P(g^l, g^r|d)P(d)\} \\ &= -\log P(g^l, g^r|d) - \log P(d) \\ &= U_c(d) + U_p(d)\end{aligned}\quad (2)$$

Two probability distributions, Conditional U_c and prior U_p , must be known in order to solve this problem. The expressions of probability distributions are solved in [10]. Final form of energy function is

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

where g is pixel difference of left and right image used in matching node and γ is occlusion node cost.

In trellis algorithm, Viterbi algorithm is used to find optimal path that minimize energy function. There are two different nodes; match node and occlusion node. Therefore, we have to use different cost calculation in each node. The algorithm is represented as:

- 1) Forward Initialization: At site $i=0$, all node costs are set to infinity except for disparity $j=0$.

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

- 2) Forward Iteration: At each step $i=1, \dots, 2N$, find the best path and best cost into each node or disparity j .

- (a) $i+j$ even (occlusion node):

$$U(i, j) = \min_{\alpha \in [-1, 1]} U(i-1, j-\alpha) + \gamma \alpha^2$$

$$P(i, j) = \arg \min_{\alpha \in [-1, 1]} U(i-1, j-\alpha) + \gamma \alpha^2$$

- (b) $i+j$ odd (match node):

$$U(i, j) = U(i-1, j) + \left(g_{\frac{1}{2}(i-j+1)}^l - g_{\frac{1}{2}(i+j+1)}^r \right)^2$$

$$P(i, j) = j$$

- 3) Backward Initialization : $i=2N$ and $j=0$

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

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

- 4) Backward Path Reconstruction: Find the optimal disparity path by tracing back the decisions.

$$\hat{d}_{i-1} = \hat{d}_i + P(i, \hat{d}_i), \quad i = 2N, \dots, 1$$

Result disparity image of trellis algorithm has "streaking" noise problem because of single scan-line matching methods. In section 3, we propose new algorithm to reduce "streaking" noise problem.

III. PROPOSED ALGORITHM

In trellis-based algorithm, "streaking" noise problem is generated because this algorithm uses single scan-line information. Motivation of our algorithm is using more information which comes from upper scan-line. If relation of

current pixel and upper line pixel is defined, we can use disparity information of upper pixel's disparity information to define current pixel's disparity.

Relation of current pixel and upper pixel is defined using pixel intensity. We assume that adjacent pixels which have similar intensity value represent the same object and same object has similar disparity values. Using this assumption, we can use upper line disparity information to decide current pixel's disparity value.

Let k is number of current scan-line. Our algorithm uses $k-1$ disparity information according to relation between upper and current line. If $k-1$ scan-line's i pixel value is similar with k scan-line's I pixel value, each pixel's disparity values are same.

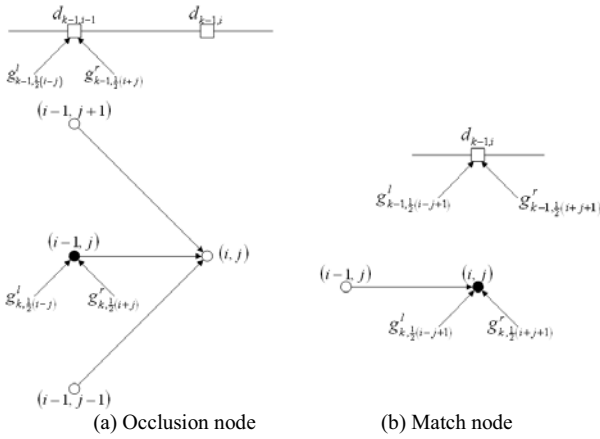


Figure 3. Modified structure in occlusion and match node

In occlusion node, match nodes image pixel intensity is compared in previous stage to compare relation with upper scan-line because there is no image pixel. After comparison, decrease same path's cost if compared pixel intensities are same. There are three paths reached in occlusion node and cost calculation formula in occlusion node is modified as follow,

$$U_k(i, j) = \min_{\alpha \in [-1, 1]} U_k(i-1, j-\alpha) + \gamma\alpha^2 - \delta(d_{k-1, i-1} - (j+\alpha)) \delta\left(g_{k, \frac{1}{2}(i-j)}^l - g_{k-1, \frac{1}{2}(i-j)}^l\right) \lambda \quad (3)$$

Where λ is weighting constant and δ is Dirac-delta function. The first delta function means difference of disparity of upper line and occlusion nodes, and second term means difference of pixel intensity which means correspondence of upper line and current line.

Modified cost calculation formula in match node is similar to occlusion node. Difference is match node has its pixel intensity. Modified formula is described as,

$$U_k(i, j) = U_k(i-1, j) + \left(g_{k, \frac{1}{2}(i-j+1)}^l - g_{k, \frac{1}{2}(i+j+1)}^r\right)^2 - \delta(d_{k-1, i-1} - j) \delta\left(g_{k, \frac{1}{2}(i-j+1)}^l - g_{k-1, \frac{1}{2}(i-j+1)}^l\right) \lambda \quad (4)$$

Where first delta function means difference of disparity of upper line and matching nodes, and second term means difference of pixel intensity which is similar in occlusion node. By this delta function, data cost, which related upper pixel's disparity, is reduced.

Instead of using absolute square difference of left and right pixel in single scan-line at second term of (4), we can use average of square pixel difference include $k-1$, $k+1$ scan-line. This reduces influence of pixel noise. The method called Average of Squared Difference (ASD) and us formula as follow,

$$ASD = \frac{1}{3} \sum_{\alpha \in [-1, 1]} \left(g_{k+\alpha, \frac{1}{2}(i-j+1)}^l - g_{k+\alpha, \frac{1}{2}(i+j+1)}^r\right)^2 \quad (5)$$

The algorithm is represented as:

- 1) Forward Initialization: At site $i=0$, all node costs are set to infinity except for disparity $d=0$.

$$U_k(0, j) = \begin{cases} 0 & j = 0 \\ \infty & \text{otherwise} \end{cases}$$

- 2) Forward Iteration: At each step $i=1, \dots, 2N$, find the best path and best cost into each node or disparity j .

- (a) $i+j$ even (occlusion node):

$$U_k(i, j) = \min_{\alpha \in [-1, 1]} U_k(i-1, j-\alpha) + \gamma\alpha^2 - \delta(d_{k-1, i-1} - (j+\alpha)) \delta\left(g_{k, \frac{1}{2}(i-j)}^l - g_{k-1, \frac{1}{2}(i-j)}^l\right) \lambda$$

$$P(i, j) = \arg \min_{\alpha \in [-1, 1]} U_k(i-1, j-\alpha) + \gamma\alpha^2 - \delta(d_{k-1, i-1} - (j+\alpha)) \delta\left(g_{k, \frac{1}{2}(i-j)}^l - g_{k-1, \frac{1}{2}(i-j)}^l\right) \lambda$$

- (b) $i+j$ odd (match node):

$$U_k(i, j) = U_k(i-1, j) + \left(g_{k, \frac{1}{2}(i-j+1)}^l - g_{k, \frac{1}{2}(i+j+1)}^r\right)^2 - \delta(d_{k-1, i-1} - j) \delta\left(g_{k, \frac{1}{2}(i-j+1)}^l - g_{k-1, \frac{1}{2}(i-j+1)}^l\right) \lambda$$

$$P(i, j) = j$$

- 3) Backward Initialization : $i=2N$ and $j=0$

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

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

- 4) Backward Path Reconstruction: Find the optimal disparity path by tracing back the decisions.

$$\hat{d}_{k, i-1} = \hat{d}_{k, i} + P_k(i, \hat{d}_i), \quad i = 2N, \dots, 1$$

IV. EXPERIMENTAL RESULT

Since the performance of this algorithm was tested on a variety of Middlebury test images [5] and synthetic images. Qualitative Evaluation is done using Middlebury test image; Baby and Aloe. PER ratio cannot be used because there is no center referenced ground truth image to compare with. The result image is shown in Figure 5, 6. Our algorithm shows that "streaking" noise problem is reduced compared with existing algorithm.

RDS is used for Quantitative Evaluation. RDS image have 256 grey levels, s^2 variance and 128 average intensities. We add zero mean and n^2 variance Gaussian noise to the RDS. Test was done with various images SNRs. Figure 4. shows the fraction of center-referenced disparity sites that are incorrectly estimated, pixel error ratio (PER), at various SNRs. PER is computed as follow.

$$PER = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (1 - \delta(d_{i,j} - \hat{d}_{i,j}))$$

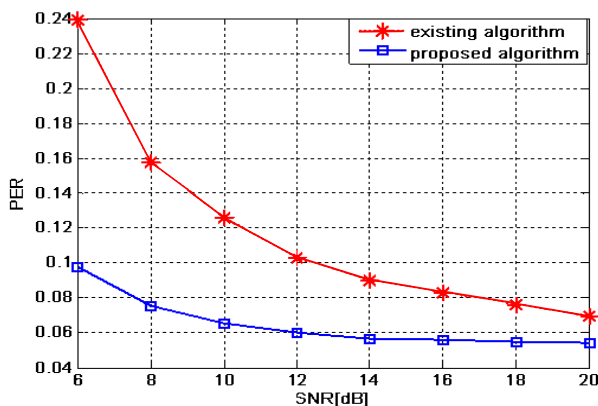


Figure 4. Error performance versus SNR

We can see graph show that PER is half of existing algorithm especially in low SNR.

In result of test image, it still shows the noises in image, but compared with the existing algorithm, performance is quite improved. In baby test image, background of existing algorithm result image has streaking noise which do not occurred in proposed algorithm. Also, boundary point of object has same result at boundary of result image. RDS result and test image result show that proposed algorithm is much reliable than existing algorithm.

V. CONCLUSIONS

A Stereo vision is used in many areas such as distance measurement area to robot-vision. But it is hard to apply in real-time because stereo matching method need much computation. By trellis-based algorithm, it is possible to get disparity result in real-time. A hardware implementation using a large array of simple PES is currently under development for real-time stereo matching. Using FPGA chip, trellis based DP algorithm can calculates 320x240 size image in 30 fps and 128 disparity level. In spite of this advantage, existing algorithm had lower matching accuracy

compared with MRF-based matching and has "streaking" noise problem. Our algorithm is proposed to reduce such noise and improve accuracy by using the upper line disparity information. PER ratio of our algorithm using RDS image shows much more good performance rather than existing algorithm.

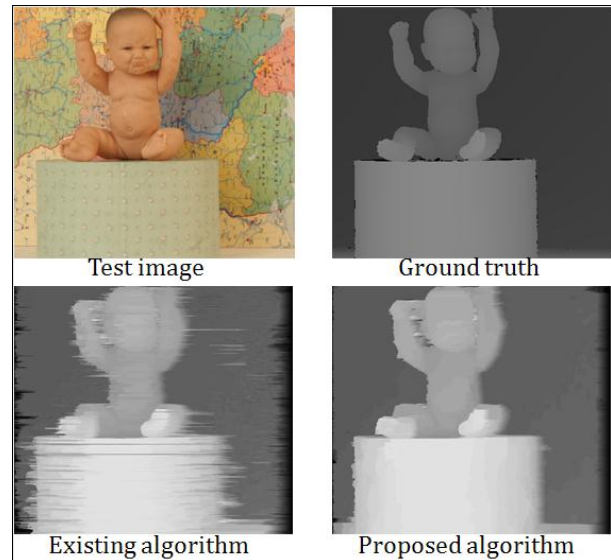


Figure 5. Result of 'Baby' test image

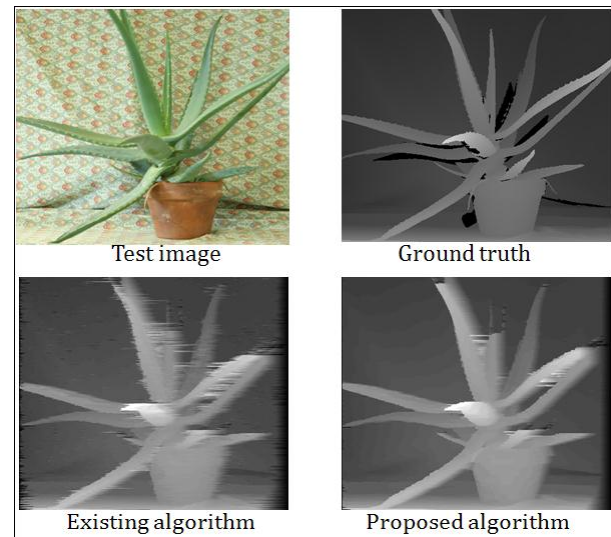


Figure 6. Result of 'Aloe' test image

ACKNOWLEDGMENT

This work was supported by the following funds: the Brain Korea 21 project and the Ministry of Knowledge Economy under Human Resources Development Program for Convergence Robot Specialists in 2010.

REFERENCES

- [1] 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.
- [2] 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.
- [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] 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.
- [5] <http://vision.middlebury.edu/stereo/>
- [6] 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.
- [7] 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.
- [8] 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.
- [9] O. Veksler. Stereo matching by compact windows via minimum ratio cycle. In *Proc. Intl Conf. Computer Vision*, 2001.
- [10] 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.