Stereo Matching Algorithm Based on Fast Guided Image Filtering for 3-Dimensional Video Service

Gwang-Soo Hong*, Byung-Gyu Kim**

Abstract

Stereo matching algorithm is an essential part in computer vision and photography. Accuracy and computational complexity are challenges of stereo matching algorithm. Much research has been devoted to stereo matching based on cost volume filtering of matching costs. Local stereo matching based guided image filtering (GIF) has a computational complexity of $O(N)$, but is still not enough to provide real-time 3-dimensional (3-D) video services. The proposed algorithm concentrates reduction of computational complexity using the concept of fast guided image filter, which increase the speed up to $O(N/s^2)$ with a sub-sampling ratio $s$. Experimental results indicated that the proposed algorithm achieves effective local stereo matching as well as a fast execution time for 3-D video service.

Keywords : Stereo Matching Algorithm, Guided Image Filtering, Fast Guided Image Filtering, 3-D Video Service

1. Introduction

Dense stereo matching algorithms are an essential area in computer vision and photography, and it plays a significant role in many 3-D applications. It is a popular technique for building a Depth of a scene observed from two slightly different viewpoints. Depth information can be obtained through disparity based on finding
correspondent pixels between a left image and a right image. Disparity \( d \) is the difference in location of the image of the object between the reference and target images as left and right image, respectively. This process is called a stereo matching, which is one of the most active research topics in computer vision. However, the captured stereo pair are disturbed by noise, e.g. sensor noise, light variations, perspective distortion and uniform regions. Many approaches for development of stereo matching algorithms have been used to improve accuracy and fast execution.

A stereo matching algorithm can be classified into the two main groups of global and local approaches [1]. Global approaches, which achieve minimization of an energy function, aim to determine disparity values for all image pixels at once and usually return more accurate disparity values at the cost of a high degree of computational complexity.

Local approaches use intensity values located in a close neighborhood of pixels with use of a support window, leading to the smoothness assumption that all pixels within a support window have the same disparity. This leads to poor performance when the window contains disparity discontinuities.

F. Tombari et al. [2] explicitly deployed a smoothness constraint within objects according to the segments. N. Y. Kwak proposed an object-based stereo matching method using segmented region[3]. Hosni et al. [4] proposed to compute the weights by the geodesic distance, which apply the foreground connectivity with low weights. Yang [5] proposed a non-local approach, in which the cost values are aggregated adaptively on a minimum spanning tree.

In recent years, cost aggregation is conducted by filtering on the cost–volume. K. J. Yoon et al. [6] used the bilateral filter weight inside a window. This approach provides weights for object boundaries that preserve edge characteristics. But the brute force implementations are of high computational complexity when the kernel window is large. Many accelerated bilateral weight aggregation methods have been proposed [7] using a sliding window technique. In this case, the aggregation process is independent of the support window size. A Honsi et al. [8] used this approach to increase the speed of a bilateral weight aggregation algorithm. C. Rhemann et al. [9] subsequently proposed local stereo matching as an efficient edge–aware local cost aggregation method using guided image filtering (GIF) [10] with time complexity \( O(N) \).

Herein, a local stereo matching algorithm based on fast GIF [11] for improvement of the aggregation step with reduction of the time complexity from \( O(N) \) to \( O(N/s^2) \) is proposed. The contribution of this research is using a reduction of image dimensions for computation of linear coefficients at every disparity.

2. Proposed Algorithm

An overview of the proposed fast cost–volume filtering based approach is shown in (Figure 1). First, the cost volume is constructed based on pixel-wise matching cost computation functions.
used. Then, each slice of the cost volume is independently filtered using the proposed method based on fast guided image filtering. Finally, the disparity of any pixel is simply chosen in a winner-take-all manner.

2.1 Cost Aggregation with Guided Filtering

Cost volume filtering approaches usually use a pixel-to-pixel comparison, which is sensitive to noise. Hence, each slice of cost volume $C$ has to be filtered. A guided image filter, designed to preserve edges, has linear time that is independent of the filter size and, thus, only depends on the number of image pixels.

GIF uses the guidance color image $G$ as the left image. Let $C$ be the filtered cost volume. The basic idea of GIF is a local linear model between $G$ and $C$ for each square window $a_k$ with a size of $k$ where the $k$-dependent filtered cost volume $C'_k$ is defined to be a linear model as:

$$ C'_k(p) = a_k \cdot G(p) + b_k, \forall p \in a_k, \tag{1} $$

where $a_k$ and $b_k$ are a $3 \times 1$ coefficient vector and scalar, respectively. For determination of the linear coefficients $a_k$ and $b_k$, the cost function is defined to minimize the error between $C'$ and $G$ as:

$$ E(a_k, b_k) = \sum_{p \in a_k} ((a_k \cdot G(p) + b_k - C(p))^2 + \epsilon \cdot a_k^2). \tag{2} $$

where $\epsilon$ is a regularization parameter to prevent $a_k$ from being too large, and the criterion of flat and textured areas is controlled. The linear coefficients $a_k$ and $b_k$ are obtained when the cost function is solved as an Euler equation, as:

$$ b_k = C'_k - a_k^T \begin{pmatrix} \mu^R_k \\ \mu^G_k \\ \mu^B_k \end{pmatrix}, \tag{4} $$

where $\mu^R$, $\mu^G$, and $\mu^B$ are mean images of each color channel of the guidance color image $G$. $\Sigma_k$ is the covariance matrix of the guidance color image $G$ in the window $a_k$, and $I_k$ is a $3 \times 3$ identity matrix.

The filtered cost volume $C'$ using the linear coefficients is defined as:

$$ C'(p) = \frac{1}{|a_k|} \sum_{p \in a_k} (a_k \cdot G(p) + b_k) = \frac{1}{\sigma_k} G(p) + \bar{b}_k, \tag{5} $$

where $\sigma_k$ and $\bar{b}_k$ are the average of the linear coefficients $a_k$ and $b_k$ in the window $k$.

Algorithm 1 shows the overall procedure for cost volume filtering using GIF.

**Algorithm 1 Computation of the filtered cost volume.**

```plaintext
Procedure CostVolumeFiltering (G, C, k) as a guidance image G, cost volume C, window size k, and regularization parameter $\epsilon$
for $d \in [d_{min}, d_{max}]$ do
    compute $\mu_k, \Sigma_p = \frac{1}{|a_k|} \sum_{p \in a_k} G(p)C(p)$ at each channel,
    $C'_k$, $C'_k$ and $\Sigma_k$
    compute $a_k$ and $b_k$ compute $\bar{a}_k$ and $\bar{b}_k$
    compute $C(p)$
end for
end procedure
```

2.2 Cost Aggregation with Fast Guided Filtering

Disparity values obtained using GIF can provide good visual quality as well as a computational complexity of $O(N)$. The proposed cost volume filtering algorithm reduces computational complexity based on fast GIF from $O(N)$ to $O(N/s^2)$ for a sub-sampling ratio $s$. The algorithm uses linear coefficients $a_k$ and $b_k$ that are computed based on a sub-sampled slice of the cost
volume \( C \) and a sub-sampled guidance color image \( G \). Then, the linear coefficients are rewritten as:

\[
\frac{1}{|B|} \sum_{p \in B} G(p) c(p) - \mu^B_{y/s} C_{y/s},
\]

\[
\frac{1}{|B|} \sum_{p \in B} G^{(b)}(p) c(p) - \mu^{B}_{y/s} C_{y/s},
\]

\[
\frac{1}{|B|} \sum_{p \in B} G^{(b)}(p) c(p) - \mu^{B}_{y/s} C_{y/s},
\]

where \( k/s \) denotes the window size for sub-sampling. To obtain a filtered cost volume \( C' \), sub-sampled linear coefficients are averaged. Up-sampled linear coefficients are calculated as:

\[
\begin{align*}
\tilde{a}_p &= F_{up-sampling}(a_{y/s}), \quad (8) \\
\tilde{b}_p &= F_{up-sampling}(b_{y/s}), \quad (9)
\end{align*}
\]

where \( F_{up-sampling} \) is a function for obtaining up-sampling data.

The filtered cost volume \( C' \) is computed using equation (5), and \( \tilde{a}_p \) and \( \tilde{b}_p \) are calculated based on up-sampled average values of linear coefficients obtained using equations (8) and (9).

Examples of cost volume slices filtered using both fast GIF and conventional GIF at disparity \( d \) are shown in (Figure 2).

(Figure 2) A filtered cost volume slice (\( d = 10 \)): (a) image patch, (b) cost volume slice, (c) cost volume slice based on fast GIF (\( s = 2 \))

Cost volume slices constructed based on matching cost functions contain considerable amounts of noise. For filtering, the proposed cost volume filtering approach preserves edges, noise is removed, and halo artifacts are avoided.

**Algorithm 2** Fast computation of the filtered cost volume.

Procedure FastCostVolumeFiltering

\( (G, C, k, s) \) as a guidance image \( G \), cost volume \( C \), window size \( k \), regularization parameter \( \epsilon \), and sub-sampling ration \( s \)

compute \( G_0 = F_{sub-sampling}(G, s) \)

for \( d \in [d_{min}, d_{max}] \) do

compute \( C_0 = F_{sub-sampling}(C, s) \)

compute \( \mu_{y/s}, \sigma_{y/s}, C_0(p), C_0(p) \) at each channel, \( C_{y/s} \) and \( \sigma_{y/s} \)

compute \( \tilde{a}_p = F_{up-sampling}(a_{y/s}, s) \) and \( \tilde{b}_p = F_{up-sampling}(b_{y/s}, s) \)

compute \( C_0(p) \)

end for

end procedure

Algorithm 2 shows the overall procedure for computation of the filtered cost volume \( C \). For efficient implementation of the proposed algorithm, all summations in Algorithm 2 are computed using the box filter technique proposed by Crow [12].

Finally, after filtering the cost volume, a disparity map is obtained based on determination of the disparity \( d \) of each pixel \( p \) between a reference image and a target image using winner-take-all optimization [1] as:

\[
d = \arg\min_{k \in [d_{min}, d_{max}]} C(k)
\]

where \( d_{min} \) and \( d_{max} \) are minimum and maximum disparity values, respectively.

The raw disparity map usually contains many outliers, especially near depth continuities and in occlusion regions. To
Stereo Matching Algorithm Based on Fast Guided Image Filtering for 3-Dimensional Video Service 527

remove outliers, we use a weighted median filter [13], which may lead to artifact and removal of thin structures.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Tsukuba</th>
<th>Venus</th>
<th>Cones</th>
<th>Teddy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>non occ</td>
<td>all</td>
<td>disc</td>
<td>non occ</td>
</tr>
<tr>
<td>CostAggr [9]</td>
<td>0.24</td>
<td>2.77</td>
<td>8.4</td>
<td>1.58</td>
</tr>
<tr>
<td>Proposed Alg.</td>
<td>2.15</td>
<td>2.87</td>
<td>7.9</td>
<td>2.12</td>
</tr>
</tbody>
</table>

<Table 1> Quantitative comparison of CostAggr [9] based on GIF and the proposed algorithm based on fast GIF with different sub-sampling ratios (s = 2)

3. Experimental Results

The proposed algorithm was verified using the Middlebury benchmark [14], which provides a collection of stereo pairs for development of stereo matching algorithms. The four standard data sets used were Tsukuba, Venus, Cones, and Teddy. The Middlebury benchmark defines three measures for evaluation of performance, including non-occluded (nonocc), all, and depth discontinuity (disc) regions. The parameters \{h, T, T_e, c, a\} = \{15, 7, 2, 0, 0.01, 0.1\} were used and kept constant for all datasets.

Quantitative performance of the proposed algorithm with different sub-sampling ratios and CostAggr [9] values is shown in <Table 1> with percentages of “bad pixels” in measurements (nonocc, all, and disc), which is defined as:

\[
B_{nonocc} = \frac{1}{N_{nonocc}} \sum_{s \in nonocc} |d(s) - d_0(s)| > \delta_t, \quad (11)
\]

\[
B_{all} = \frac{1}{N_{all}} \sum_{s \in all} |d(s) - d_0(s)| > \delta_t, \quad (12)
\]

\[
B_{disc} = \frac{1}{N_{disc}} \sum_{s \in disc} |d(s) - d_0(s)| > \delta_t, \quad (13)
\]

where \(B_{nonocc}, B_{all}, B_{disc}\) are percentages of bad pixels, which is determined ground-truth image with disparity tolerance \(\delta_t\).

<Table 1> shows the difference of average result about 0.07% compared with CostAggr [9] which was implemented ourself. We obtained similar quantitative results since the sub-sampling and up-sampling effects corrected the loss of disparity values by removing noise. The algorithm with a sub-sampling ratio of \(s=2\) achieved quantitative performance similar to CostAggr [9].

Only the execution time of the aggregation performance for the left image was measured using a PC with an Intel i7 Core at 3.4 GHz. The proposed method was approximately 2× than CostAggr [9] with sub-sampling ratios of \(s=2\) in <Table 2>. Estimated disparity maps are shown for visual quality comparison in (Figure 3). Use of fast GIF with a sub-sampling ratio of \(s=2\) achieved visual performance similar to CostAggr [9] because the sub-sampled matching cost for computation of linear coefficients removed noise.

<table>
<thead>
<tr>
<th>Image</th>
<th>Resolution</th>
<th>Disparity range</th>
<th>Time (Sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>CostAggr</td>
</tr>
<tr>
<td>Tsukuba</td>
<td>384×288</td>
<td>15</td>
<td>0.55</td>
</tr>
<tr>
<td>Venus</td>
<td>434×383</td>
<td>20</td>
<td>1.06</td>
</tr>
<tr>
<td>Cones</td>
<td>450×375</td>
<td>60</td>
<td>3.34</td>
</tr>
<tr>
<td>Teddy</td>
<td>450×375</td>
<td>60</td>
<td>3.33</td>
</tr>
</tbody>
</table>

<Table 2> Computational efficiency of the proposed algorithm with CostAggr [9] in the aggregation step.
4. Conclusions

In this paper, a fast local stereo matching algorithm using fast GIF has been proposed that focuses on reduction of computation complexity for 3D video service. Experimental results showed the performance similar to CostAggr[9] with faster execution time (over 2 times) using the Middlebury benchmark images. The proposed algorithm achieved the reduced complexity of up to $O(N/s^2)$ with a sub-sampling ratio $s$, theoretically.

ACKNOWLEDGEMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (NRF-2016R1D1A1B04934750).
References


