Fuzzy logic-based disparity selection using multiple data costs for stereo correspondence

Akhil Appu SHETTY1, Vadakekara Itty GEORGE1, C Gurudas NAYAK1,∗, Raviraj SHETTY2

1Department of Instrumentation and Control Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India
2Department of Mechanical and Manufacturing Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India

Received: 18.07.2018 • Accepted/Published Online: 29.11.2018 • Final Version: 22.01.2019

Abstract: Stereo matching algorithms are capable of generating depth maps from two images of the same scene taken simultaneously from two different viewpoints. Traditionally, a single cost function is used to calculate the disparity between corresponding pixels in the left and right images. In the present research, we have considered a combination of simple data costs. A new method to combine multiple data costs is presented and a fuzzy-based disparity selection method is proposed. Experiments with different combinations of parameters are conducted and compared through the Middlebury and Kitti Stereo Vision Benchmark.

Key words: Fuzzy logic, stereo matching, mutual information, normalized cross-correlation, Middlebury stereo dataset, Kitti stereo dataset

1. Introduction

Stereo matching is a fundamental problem for many computer vision tasks such as view synthesis, autonomous navigation, and 3-D reconstruction [1]. The main purpose of stereo matching is to extract 3-D information of a scene by evaluating the similarities between the given stereo pairs taken at different viewing positions of the same scene. There are two categories of stereo matching, namely dense stereo matching and sparse stereo matching. Dense stereo algorithms aim at determining correspondences of every pixel in images, whereas sparse stereo matching algorithms mainly consider only the features within the scene and not all the pixels. This topic was exhaustively reviewed in [2, 3]. Apart from the above-mentioned classification, stereo matching algorithms can also be categorized into local methods and global methods. Local approaches use windows or small patches of pixels in the reference image and utilize the information within this finite region around the pixel (window) to determine the disparity of the pixels by comparing similar patches in the target image. Global approaches, on the other hand, integrate explicit smoothness assumptions and calculate all the disparities concurrently by applying energy minimization methods. Generally, local stereo matching algorithms are usually faster than their global counterparts and exhibit a smaller memory requirement. Unfortunately, the results obtained through these methods have lower accuracy when compared to global state-of-the-art algorithms. Many new local algorithms based on adaptive weight [4, 5] exhibit results similar to those obtained using global stereo matching methods. Unfortunately, the computational complexity of this type of local algorithms is high, and it varies quadratically when compared to the window size used to aggregate the matching costs.

∗Correspondence: cg.nayak@manipal.edu

This work is licensed under a Creative Commons Attribution 4.0 International License.
The present work is influenced by [5–7]. Instead of taking only a single cost function to evaluate the level of similarity, we use a combination of normalized mutual information (NMI), normalized cross-correlation (NCC), and absolute differences of gradients on the image pairs. Moreover, a fuzzy-based method of disparity selection is also presented. Different combinations of the above-mentioned methods are considered along with different parameters such as window size and number of fuzzy rules. The selection of the best method is based on the error produced by the various combinations of the parameters mentioned above.

In the rest of the paper, Section 2 presents the literature that was reviewed for this research. This section describes the various relevant works. Section 3 talks about the method that has been proposed and the results obtained from them, followed by the conclusion in Section 4.

2. Literature review

It was observed in [7] that mutual information (MI) handles radiometric differences well but its performance depends heavily on the size of the window. MI is a parametric matching method that uses the intensity of the pixels for processing and matching as compared to nonparametric methods, which use the local ordering of the pixel intensities for processing/matching. The authors further claimed that if two signals are similar, then the joint entropy of the two signals will be minimum as the uncertainty between two signals is low. When joint entropy is minimized, the MI will increase. Hence, for similar signals, the MI is higher. In the case of images, two constant regions will also have low joint entropy. In such cases, to avoid spurious matches, it becomes necessary to maximize the individual entropies of the samples to be compared. Hence, in such cases, MI will be useful.

Another study presented in [8] calculated the initial disparity map using NCC. Disparities with weak confidence are removed from the map based on the results of performing edge detection on the disparity map.

For every edge pixel, the neighboring row and columns are denoted as weak confidence elements based on the equation

$$\forall \text{edgepixel}(x,y), \bigcup_{i=x-R}^{x+R} \text{Disparity}_{i,y}\text{and} \bigcup_{j=y-R}^{y+R} \text{Disparity}_{x,j} \text{are weak}, \tag{1}$$

where $R$ is the radius.

Based on the elements of weak confidence obtained in the previous section, elements in the left and right image are rejected. The authors then perform a histogram specification on the modified images. The disparity map was obtained by using the sum of absolute differences (SAD) data cost:

$$\text{SAD} c(x,y) = \min(\text{SAD} (x,y), \text{SAD} h(x,y)), \tag{2}$$

where $\text{SAD} h (x,y) = \text{SAD}$ between histogram specified images.

Another work [9] shows that the authors create a descriptor vector for every pixel, which was further used for matching. The proposed descriptor consists of 17 elements. The first three elements are CIELAB values, which indicate the color difference in the low frequency band.

The remaining elements describe the components belonging to higher frequencies, which are extracted from gradient and texture information. Through experimentation, it was shown that the proposed method provided a good performance in terms of the percentage of bad pixel matching, and it outperformed state-of-the-art algorithms by reducing about 2% mismatching error, which achieves about 16.5% performance improvement.
Additionally, outdoor image pairs were used to show the effectiveness of the proposed algorithm for real-world applications, and better results were achieved in the proposed system.

Some authors proposed an augmented version of the census transform [10]. The general equation of census transform is given as:

$$CT(P_1, P_2) = \begin{cases} 1, & P_1 > P_2 \\ 0, & P_1 \leq P_2, \end{cases}$$  \hspace{1cm} (3)

where $P_2$ is the central pixel and $P_1$ is the pixel under consideration. The original equation of the census transform was modified by the authors. The matching cost is the Hamming distance between the two strings. The best disparity is selected through a winner-takes-all mechanism. Instead of using the regular census transform, the authors here used the mean value and the midpoint value of the elements of the window to classify the pixels into 4 categories:

$$\varepsilon(p', \bar{p}, p_{mid}) = \begin{cases} 0, & p' \leq \bar{p} \\ 1, & p' \leq p_{mid} \\ 2, & p' > p_{mid} \\ 3, & p' > \bar{p}, \end{cases}$$  \hspace{1cm} (4)

where $p'$ is the pixel under consideration, $\bar{p}$ is the mean of the pixels in the window considered, and $p_{mid}$ is the middle value of the pixels. The authors claim that their method shows better matching accuracy than the traditional census transform.

Another work on census transform [11] proposed a color census transform, which the authors claimed to have provided better quality images for stereo matching when compared to gray-scale modified census transform. The Gaussian color model was used, which could overcome the problems posed by shadows and highlights.

$$\begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix} = \begin{bmatrix} 0.06 & 0.63 & 0.27 \\ 0.30 & 0.04 & -0.35 \\ 0.34 & -0.60 & 0.17 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$  \hspace{1cm} (5)

The color difference between two pixels can be calculated using the Euclidean distance denoted by $\Delta E_G$. The average of the color differences between the pixels within the window and the central pixel is described as $\Delta E_m$.

The authors here have incorporated the Gaussian color model into the regular census transform as follows:

$$CCT(u, v) = CT(\Delta E_m, (I(u + i, v + j))),$$  \hspace{1cm} (6)

where $i$, $j$ are elements within the window being considered. Again, the sum of the Hamming distance is used as a matching cost. Compared to gray-scale modified census transform, the authors claimed that their method produces better results.

Further work on MI [12, 13] stressed the fact that MI is a very good candidate for stereo matching even in multimodal matching. This data cost can also easily be combined with other parameters such as gradients to produce an even more enhanced result.

The authors of [14] presented another work that combines multiple data costs. The authors combined absolute differences (ADs), gradient matching, and census transform for matching cost computation. The disparity selection was made on a winner-takes-all (WTA) scheme. Further refinement of the disparity map was carried out through left-right consistency check and hole filling as postprocessing. The authors also mentioned...
that since they use multiple cost functions, their method overcomes the problems that are usually faced by methods that use a single data cost for calculation of disparity.

A method of disparity map refinement was proposed in [15]. Holes in disparity maps can be detected by comparing their disparities with those of their neighbors. These holes are then updated by finding the most appropriate disparity in the neighborhood. Inconsistencies around the edges were also considered. The original images were segmented using mean shift segmentation and the edges were compared with the disparity map generated. Any disparity edges that did not coincide with the edges of the image were considered as inconsistent edges.

A matching method based on separate aggregations of the area was proposed in [16]. The cost function used here was the truncated SAD on color images. The innovation of this work was the aggregation of the cost. The authors used two aggregation regions, Cs and Cw. The image was initially segmented and each region was considered as an area for aggregation of the cost and denoted by Cs, whereas Cw denotes the aggregation over a regular window. The final aggregation cost is derived as a linear combination of the two areas.

An extension of the SAD, the truncated absolute difference (TAD) has been used by the authors [17]. The TAD has been used on the image and also on the gradient of the image. Similar to the previously referred work, the authors again used a linear combination of the two data costs, along with the weight parameters obtained from a guided filtering weight kernel.

The combination of the costs was represented as

\[ C(p, d) = (1 - a) \ast TAD + a \ast (TAD \ of \ gradients). \]  

The final aggregated cost was

\[ C'(p, d) = \sum_{q \in I} w_{p,q}(I)C(q, d). \]  

The weight \( w_{p,q} \) was calculated based on a guided filtering weight kernel.

3. Proposed methodology

The images for this work were obtained from the Middlebury (http://vision.middlebury.edu/stereo/data/) as well as Kitti Stereo Vision Benchmark [18], which are the standard platforms for researchers working in this area. We are using the latest (2014 and 2015, respectively) versions of the mentioned datasets. The reason we have selected these datasets is because they are being used repeatedly in almost every research paper related to stereo matching. Moreover, the Middlebury dataset includes images that represent a variety of situations that can prove to be a hindrance for stereo matching algorithms. These include images with different lighting, different exposure, regions with large depth discontinuities, and even texture-less regions. One of the image pairs along with the ground truth is shown in Figure 1a, 1b, and 1c, respectively.

Our initial approach included using just a single matching cost. It is claimed in the literature reviewed in the previous section that mutual information is a very useful metric, which can be used for stereo matching.

A variant of this metric, known as the normalized mutual information (NMI), is used here and is denoted by Eq. (9) [19]:

\[ NMI(a, b) = \sqrt{\frac{MI(a, b)}{\text{entropy}(a)} \ast \frac{MI(a, b)}{\text{entropy}(b)}}. \]
One of the image pairs from the Middlebury dataset (left and center) along with the ground truth (right).

Figure 1.

where

\[
MI(a, b) = \text{entropy}(a) + \text{entropy}(b) - \text{joint entropy}(a, b).
\]  

(10)

Using just this metric to estimate the similarity between the stereo pair of images resulted in a disparity map with many errors as shown in Figure 2. It is evident that this method alone cannot be used for this purpose.

Figure 2. Result of using NMI as a stereo matching metric.

Hence, in an attempt to reduce the errors of the resulting disparity maps, NMI was used in conjunction with a few preprocessing techniques including histogram equalization (HE) [20], top hat filtering (THF), log chromaticity normalization (LCN) [21], and census transform [22]. Since the Middlebury dataset that we are using in our work (2014 version) even consists of images with radiometric variations, we have attempted to use the above-mentioned techniques to compensate for the issue.

The results of the various combinations of these preprocessing methods are presented in Figure 3a, 3b, and 3c, respectively.

The census transform is given by [22]:

\[
V(x) = \begin{cases} 
1, & I(x) < I(y) \\
0, & \text{otherwise.}
\end{cases}
\]  

(11)
A modified version of the census transform can be stated as \[ V(x) = \begin{cases} 1, & I'(x) < I(y) \\ 0, & \text{otherwise} \end{cases} \quad (12) \]

where \( I'(x) \) is the mean of the neighborhood of the pixels under consideration.

The above two census transform-based methods produce only binary outputs. Since the quality of the results produced by NMI is better with more bins, it is advisable to reformulate the modified census transform to generate nonbinary results. This can be represented as Eq. \((13)\). The disparity maps produced as a result of these data cost metrics are presented in Figures 4a and 4b, respectively.

\[
V'(x) = \begin{cases} I'(x), & I'(x) < I(y) \\ I(y), & \text{otherwise} \end{cases}
\quad (13)
\]

It can clearly be observed that except for the ones using census transform, the results are not favorable even after the use of preprocessing methods. Having said that, even in the case where the census transform is used, the quality of the results is not up to the mark.

It has been observed from the literature review section that using multiple data costs provides a good solution to the problem at hand. Hence, we adopted a similar approach in our work and used the cost functions advocated in the literature, namely MNCC and NMI, which are denoted by:

\[
MNCC(X, Y) = \frac{2COV(X, Y)}{VAR(X) + VAR(Y)},
\quad (14)
\]
\[ NMI(X, Y) = \frac{H(X) + H(Y) - H(X, Y)}{\sqrt{H(X) * H(Y)}}, \]  

(15)

where

\[ H(A) = -\sum_{i=0}^{n} p(i) \log p(i), \]  

(16)

\[ H(A, B) = -\sum_{i=0}^{n} p(i, j) \log p(i, j). \]  

(17)

Instead of considering the above-mentioned metrics individually as the matching cost, their product has been taken as the final matching cost. The resulting disparity map is shown in Figure 4c. Upon visual inspection, it is observed that the combination of NMI and MNCC yields better results than the ones observed earlier.

Another problem observed was the selection of the right disparity value for a pixel. Most of the methods in the literature concentrate mainly on the selection and development of the cost function. The selection of the disparity value is almost always done through a WTA-based method [22]. There is always a possibility that a cost function might produce multiple maximums/minimums while calculating the cost. This leads to much ambiguity. When a WTA-based method is used in the presence of such ambiguities, the results tend to be erroneous. Keeping this problem in mind, an “intelligence”-based disparity selection method is proposed.

3.1. Disparity selection method

It has been observed that all the disparity selection was mainly based on the WTA method [22]. As mentioned in the previous section, using a WTA approach is not always desirable since the ambiguities might lead to an erroneous result. Hence, a novel method of using fuzzy logic to select the right disparity has been proposed here.

![Figure 5. Selection of disparities through a fuzzy based approach.](image)

The proposed approach is represented as shown in Figure 5. The present work was conducted using a Mamdani fuzzy inference system.
Table 1. Table of fuzzy rules.

<table>
<thead>
<tr>
<th>Rule number</th>
<th>Input 1 (data cost 1)</th>
<th>Input 2 (data cost 2)</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Poor</td>
<td>Poor</td>
<td>Poor</td>
</tr>
<tr>
<td>2</td>
<td>Poor</td>
<td>Very bad</td>
<td>Poor</td>
</tr>
<tr>
<td>3</td>
<td>Poor</td>
<td>Bad</td>
<td>Very bad</td>
</tr>
<tr>
<td>4</td>
<td>Poor</td>
<td>Average</td>
<td>Bad</td>
</tr>
<tr>
<td>5</td>
<td>Poor</td>
<td>Good</td>
<td>Average</td>
</tr>
<tr>
<td>6</td>
<td>Poor</td>
<td>Very good</td>
<td>Average</td>
</tr>
<tr>
<td>7</td>
<td>Very bad</td>
<td>Poor</td>
<td>Poor</td>
</tr>
<tr>
<td>8</td>
<td>Very bad</td>
<td>Very bad</td>
<td>Very bad</td>
</tr>
<tr>
<td>9</td>
<td>Very bad</td>
<td>Bad</td>
<td>Very bad</td>
</tr>
<tr>
<td>10</td>
<td>Very bad</td>
<td>Average</td>
<td>Bad</td>
</tr>
<tr>
<td>11</td>
<td>Very bad</td>
<td>Good</td>
<td>Below average</td>
</tr>
<tr>
<td>12</td>
<td>Very bad</td>
<td>Very good</td>
<td>Below average</td>
</tr>
<tr>
<td>13</td>
<td>Bad</td>
<td>Poor</td>
<td>Very bad</td>
</tr>
<tr>
<td>14</td>
<td>Bad</td>
<td>Very bad</td>
<td>Very bad</td>
</tr>
<tr>
<td>15</td>
<td>Bad</td>
<td>Bad</td>
<td>Bad</td>
</tr>
<tr>
<td>16</td>
<td>Bad</td>
<td>Average</td>
<td>Below average</td>
</tr>
<tr>
<td>17</td>
<td>Bad</td>
<td>Good</td>
<td>Average</td>
</tr>
<tr>
<td>18</td>
<td>Bad</td>
<td>Very good</td>
<td>Average</td>
</tr>
<tr>
<td>19</td>
<td>Average</td>
<td>Poor</td>
<td>Bad</td>
</tr>
<tr>
<td>20</td>
<td>Average</td>
<td>Very bad</td>
<td>Bad</td>
</tr>
<tr>
<td>21</td>
<td>Average</td>
<td>Bad</td>
<td>Below average</td>
</tr>
<tr>
<td>22</td>
<td>Average</td>
<td>Average</td>
<td>Average</td>
</tr>
<tr>
<td>23</td>
<td>Average</td>
<td>Good</td>
<td>Above average</td>
</tr>
<tr>
<td>24</td>
<td>Average</td>
<td>Very good</td>
<td>Good</td>
</tr>
<tr>
<td>25</td>
<td>Good</td>
<td>Poor</td>
<td>Below average</td>
</tr>
<tr>
<td>26</td>
<td>Good</td>
<td>Very bad</td>
<td>Below average</td>
</tr>
<tr>
<td>27</td>
<td>Good</td>
<td>Bad</td>
<td>Average</td>
</tr>
<tr>
<td>28</td>
<td>Good</td>
<td>Average</td>
<td>Above average</td>
</tr>
<tr>
<td>29</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>30</td>
<td>Good</td>
<td>Very good</td>
<td>Very good</td>
</tr>
<tr>
<td>31</td>
<td>Very good</td>
<td>Poor</td>
<td>Average</td>
</tr>
<tr>
<td>32</td>
<td>Very good</td>
<td>Very bad</td>
<td>Average</td>
</tr>
<tr>
<td>33</td>
<td>Very good</td>
<td>Bad</td>
<td>Above average</td>
</tr>
<tr>
<td>34</td>
<td>Very good</td>
<td>Average</td>
<td>Good</td>
</tr>
<tr>
<td>35</td>
<td>Very good</td>
<td>Good</td>
<td>Very good</td>
</tr>
<tr>
<td>36</td>
<td>Very good</td>
<td>Very good</td>
<td>Best</td>
</tr>
</tbody>
</table>

The main attribute of this fuzzy inference system is the rule base. The output is decided based on these rules formulated in an 'if-then' setup [23]. The first input to the fuzzy inference system (FIS) is matching cost 1 (combination of NMI and MNCC) and the second input is matching cost 2 (SAD of gradients). The second
matching cost was decided to be the SAD of gradients as this matching cost provided visually acceptable results when compared to other methods. Hence, the cost functions considered for this work were the NMI, MNCC, and SAD of gradients of the images. The rule base for our system is shown in Table 1.

For every pixel, the data costs for all the considered disparities are calculated using the cost functions mentioned above and are fed into the FIS. The results for all the disparities are saved and the disparity for which the FIS produces the maximum value is considered to be the right disparity.

We have implemented the FIS through the MATLAB 2018 Graphical User Interface (GUI). As mentioned previously, two inputs were fed into the FIS. Figure 6a presents a bird’s eye view of the two-input fuzzy inference system that we have used. Data cost 1 and Data cost 2 are nothing but Input 1 and Input 2 mentioned in Table 2.

![Bird's eye view of FIS system used](image)

**Figure 6.** Bird’s eye view (left) and Data cost 1 membership function (right).

Figure 6b and Figure 7a describe the membership functions of the two inputs that we have used. Based on experimentation results, we have used six membership functions (poor, very bad, bad, average, good, very good) for both the inputs as this gave the best results. We have used trapezoidal membership functions for both the inputs as we wanted to consider a wider range of values to fall under a certain membership function.

![Data cost 2 membership function](image)

**Figure 7.** Data cost 2 membership function (left) output membership function (right).

Figure 7b describes the membership functions of the output. For the output, we have used nine membership functions (poor, very bad, bad, below average, average, above average, good, very good, best). We have used Gaussian membership functions as we wanted our output to be precise and not be constant for a range of values.
Figure 8. Fuzzy inference system with two inputs and one output.

Figure 8 describes the numerical representation of the rules that are mentioned in Table 1. It tells us what output we would obtain for a certain input combination.

The variables that were experimented on in this work are the window size, the number of fuzzy rules used, and the different combinations of the cost functions given as input to the FIS.

Initially, matching cost 1 was calculated as a combination of MNCC and NMI and matching cost 2 was the SAD of gradients of the two input images.

The above-mentioned combination of cost functions was tested with window sizes of 11×11 and 13×13 with 16 and 36 rules. For the case with 16 rules, 4 membership functions for each of the two inputs were used. It was observed that increasing the window size from 11 to 13 and changing the number of rules from 16 to 36 improved the error from 25.8 to 23.5.

The new combination of MNCC and SAD of gradients as the first input and NMI as the second input produces slightly better results. Hence, to observe the effect of window size, we fix the fuzzy rules to 36 and the input combinations as mentioned above and keep increasing the window size up to 19×19. It was noticed that increasing the window size beyond 13 increases the error gradually by 0.5–0.6 for every iteration.

Next, fixing the window size as 13×13 and 36 fuzzy rules, the combination of inputs were changed to NMI and SAD of gradients as matching cost 1 and matching cost 2 was changed to MNCC. This resulted in a still lower error of 22.5 as the average error of all pixels and an average error of nonoccluded pixels as 13.8.

Since this provided the least error of all the combinations the same combination with 13×13 window was used to see if increasing the rule base would further reduce the error. To check this, the rule base was increased from 36 to 64. It was found that increasing the rule base led to increase in error. The tabular representation of the above-mentioned observations are presented in Table 2.

Upon fixing the combination of methods, window size, and fuzzy rules, based on the above observations, a
Table 2. Error comparison of various combinations of methods and parameters for Middlebury Stereo Dataset (2014).

<table>
<thead>
<tr>
<th>Method</th>
<th>Input 1 to fuzzy based disparity selector</th>
<th>Window size</th>
<th>Fuzzy rules</th>
<th>Average error (all pixels)</th>
<th>Average error (nonoccluded)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>11 × 11</td>
<td>16</td>
<td>25.8</td>
<td>17.4</td>
</tr>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>13 × 13</td>
<td>16</td>
<td>25.5</td>
<td>16.9</td>
</tr>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>13 × 13</td>
<td>36</td>
<td>23.5</td>
<td>14.9</td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>13 × 13</td>
<td>36</td>
<td>22.7</td>
<td>13.7</td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>15 × 15</td>
<td>36</td>
<td>23.3</td>
<td>14.2</td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>17 × 17</td>
<td>36</td>
<td>23.8</td>
<td>14.7</td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>19 × 19</td>
<td>36</td>
<td>24.3</td>
<td>15.3</td>
</tr>
<tr>
<td>NMI+SAD of Image Gradient</td>
<td>MNCC</td>
<td>13 × 13</td>
<td>36</td>
<td>22.5</td>
<td>13.8</td>
</tr>
<tr>
<td>NMI+SAD of Image Gradient</td>
<td>MNCC</td>
<td>13 × 13</td>
<td>64</td>
<td>23.7</td>
<td>15.0</td>
</tr>
</tbody>
</table>

A few preprocessing methods were also applied to the images before following the methodology mentioned. These methods include contrast limited adaptive histogram equalization (CLAHE) [24], mean filter, and median filter [25].

It was observed that CLAHE produced an average error of 26.2 and an error of 16.8 in nonoccluded areas. The mean filter gave a slightly lower error as compared to the median filter for a window of 3 × 3 (median produced an average error of 22.3 and an error of 13.2 in nonoccluded regions). Hence, higher window sizes only for mean filter were considered and the results are tabulated in Table 3.

Table 3. Error comparison of applying mean filter.

<table>
<thead>
<tr>
<th>Window size</th>
<th>Mean filter</th>
<th>Average error</th>
<th>Nonoccluded error</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 × 3</td>
<td>21.9</td>
<td>12.8</td>
<td></td>
</tr>
<tr>
<td>5 × 5</td>
<td>22.6</td>
<td>13.5</td>
<td></td>
</tr>
<tr>
<td>7 × 7</td>
<td>23.5</td>
<td>14.4</td>
<td></td>
</tr>
</tbody>
</table>

3.2. Postprocessing

The above-mentioned results can be further improved through a postprocessing method [25]. We have considered the results of using only the 3 × 3 window size as it produced the least error.

In the disparity maps obtained, the main idea is to replace the positions with minimum values (zero) as they indicate pixels that are infinitely far away. In the Middlebury stereo dataset considered, all the pixels have
a fixed distance; hence, if a pixel shows a zero value, it has to be because of an error in the calculation of the disparity of that pixel. The final error is presented in Table 4 and the methodology is presented below.

Consider matrix \( A \), which has the same size as the disparity map produced by the methodology mentioned above. This matrix would only consider the positions in the initial disparity map that have the least values and replace them with the maximum value of the disparity map. The matrix is further dampened using a weight “\( \delta \)” and added with the initial disparity map to obtain the final disparity map.

\[
\text{final disparity} = \text{disparity} + \delta A
\]

\[
A(x) = \begin{cases} 
\max(\text{disparity}), & \text{disparity}(x) = \min(\text{disparity}(x)) \\
0, & \text{otherwise}
\end{cases}
\]

Here, \( \text{disparity} \) is the initial disparity map obtained and \( A(x) \) is the new matrix formed.

<table>
<thead>
<tr>
<th>( \delta )</th>
<th>Average error</th>
<th>Nonoccluded error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>21.4</td>
<td>12.7</td>
</tr>
<tr>
<td>0.2</td>
<td>21.1</td>
<td>12.6</td>
</tr>
<tr>
<td>\textbf{0.3}</td>
<td>\textbf{20.9}</td>
<td>\textbf{12.5}</td>
</tr>
<tr>
<td>0.4</td>
<td>21</td>
<td>12.5</td>
</tr>
<tr>
<td>0.5</td>
<td>21.1</td>
<td>12.5</td>
</tr>
</tbody>
</table>

We have also applied our stereo matching algorithm on the Kitti Stereo dataset (2015) to have a better understanding of the behavior and performance of said algorithm. The advantage of the Kitti dataset is that it has extensive data on stereo images of real-world environments. Table 5 describes the performance of our algorithm in different parameter settings. Two of the reference images belonging to the Kitti Stereo dataset (2015) along with their obtained disparity maps are shown in Figure 9a, 9b, 10a, and 10b, respectively.

**Figure 9.** Reference image (left) and the obtained disparity map (right).

### 4. Conclusion

A method consisting of a combination of the NMI, MNCC, and SAD of gradients is tested, which was not previously observed in the literature. A method of fuzzy-based disparity selection is proposed. A comparison of the behavior of the algorithm is made by applying the mentioned algorithm on both the Middlebury and Kitti stereo datasets. The former dataset contains many artificially created environments such as variations in
Table 5. Error comparison of various combinations of methods and parameters for Kitti Stereo Dataset (2015).

<table>
<thead>
<tr>
<th>Method</th>
<th>Input 1 to fuzzy based disparity selector</th>
<th>Input 2 to fuzzy based disparity selector</th>
<th>Window size</th>
<th>Fuzzy rules</th>
<th>Error present</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>11 x 11</td>
<td>16</td>
<td>10.1232</td>
<td></td>
</tr>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>13 x 13</td>
<td>16</td>
<td>10.1685</td>
<td></td>
</tr>
<tr>
<td>NMI+MNCC</td>
<td>SAD of Image Gradient</td>
<td>13 x 13</td>
<td>36</td>
<td>10.5765</td>
<td></td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>13 x 13</td>
<td>36</td>
<td>8.9224</td>
<td></td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>15 x 15</td>
<td>36</td>
<td>9.3481</td>
<td></td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>17 x 17</td>
<td>36</td>
<td>9.7473</td>
<td></td>
</tr>
<tr>
<td>MNCC+SAD of Image Gradient</td>
<td>NMI</td>
<td>19 x 19</td>
<td>36</td>
<td>10.1271</td>
<td></td>
</tr>
<tr>
<td>NMI+SAD of Image Gradient</td>
<td>MNCC</td>
<td>13 x 13</td>
<td>36</td>
<td>7.7235</td>
<td></td>
</tr>
<tr>
<td>NMI+SAD of Image Gradient</td>
<td>MNCC</td>
<td>13 x 13</td>
<td>64</td>
<td>9.0239</td>
<td></td>
</tr>
</tbody>
</table>

lighting and uneven depth discontinuities while the latter has more real-world natural outdoor environments. The results on the Kitti dataset provided lower error when compared to the Middlebury stereo dataset. Based on the experiments conducted, the combination of the NMI and SAD of gradients as one input and MNCC as another input along with 36 fuzzy rules and a window size of 13 gave the best results. Even though the above-mentioned methods provide a considerably accurate result in both artificially created and natural outdoor environments, it has been observed that the regions with depth discontinuities tend to be thickened in the obtained disparity map. Reducing this ‘thickening’ effect may further improve the accuracy of the results.

Acknowledgments

This research work was funded by the Dr. T.M.A. Pai PhD scholarship program by the Manipal Academy of Higher Education, Manipal. The authors would like to thank the department of Instrumentation & Control.
Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, for allowing
the research to be conducted in their laboratory and providing the required facilities.

References

Anal 2011; 33: 807-22.


2010.


650-656.


[8] Baydoun M, Al-Alaoui MA. Enhancing stereo matching with varying illumination through histogram information
and normalized cross correlation. In: IEEE Systems, Signals and Image Processing; 7 July 2013; Bucharest,

Lett 2016; 78: 41-47.


Imaging 2016; 25: 033013.

[13] Pluim JP, Maintz JA, Viergever MA. Image registration by maximization of combined mutual information and

[14] Hamzah RA, Ibrahim H, Hassan AH. Stereo matching algorithm based on per pixel difference adjustment, iterative


[16] Tombari F, Mattoccia S, Di Stefano L, Addimanda E. Near real-time stereo based on effective cost aggregation. In:
IEEE International Conference on Pattern Recognition; 8 December 2008; Tampa, FL, USA. New York, NY, USA:
IEEE. pp.1-4.

Optics 2014; 53: 6885-6892.

1231-1237.


