Stereo Matching Based on Features of Image Patch

Shi Zhou\textsuperscript{a,}\textsuperscript{*}, He Li\textsuperscript{b}, Miaomiao Zhu\textsuperscript{a}, Zhen Li\textsuperscript{a}, Mitsunori Mizumachi\textsuperscript{a}, Lifeng Zhang\textsuperscript{a}

\textsuperscript{a}Kyushu Institute of Technology, Kitakyushu, Fukuoka, 804-8550, Japan
\textsuperscript{b}Northeastern University, Shenyang, Liaoning, 110819, China

\textsuperscript{*}Corresponding Author: zhou.shi403@mail.kyutech.jp

Abstract

Stereo matching is a branch of 3D vision and has a wide range of applications in 3D reconstruction and autonomous driving. Recently, stereo matching methods leverage all the information of the stereo image to calculate disparity map. However, these methods still have difficulties in texture-less areas and occlusion areas, and post-processing improves accuracy. Therefore, there is a high computational cost in feature extraction and post-processing. In this paper, we propose a stereo matching method based on features of image patches to predict the disparity of non-occlusion areas instead of full image features. And aggregation methods are performed to modify all kinds of mismatching pixels based on the correct disparity in the non-occlusion areas. Furthermore, we evaluated the proposed method on the Middlebury dataset. The result shows that the proposed method performs well in all areas.

keywords: Stereo matching, Disparity prediction, Image patch, Convolutional neural networks, 3D vision.

1. Introduction

Stereo matching and disparity estimation from stereo images are fundamental problems in 3D computer vision. The goal of stereo matching is to compute the disparity map for two images of the same scene. Usually, these two images are taken from cameras at different horizontal positions but at the same vertical position. Disparity indicates the distance in the horizontal direction. Therefore, according to the parallel model of binocular vision, the distance from the object to the camera can be calculated from the disparity map.

As a popular research topic for decades, traditional stereo matching methods comprise four steps: calculation of matching cost, aggregation of matching cost, optimization, and disparity refinement. These methods usually calculate the similarity between pixels by designing an energy function. If the two pixels are very similar, the matching cost is low, and if the two pixels are very different, the matching cost is high. Therefore, calculating the matching cost with high accuracy is the main problem of the stereo matching method. However, the energy function is sensitive to different lighting conditions. For different applications, different energy functions should be designed. Therefore, traditional stereo matching methods have poor robustness for complex environments.

Recently, convolutional neural networks (CNN) have performed well in image feature extraction, and a lot of stereo matching methods based on CNN have been proposed. These methods use end-to-end disparity estimation networks to integrate all steps in stereo matching. However, these methods still have difficulties in ill-posed regions, such as occlusion areas and texture-less areas. Post-processing is necessary to obtain a high accuracy disparity map. Also, the end-to-end network usually has a complex network structure, so the computational cost is high.

In this work, we proposed a novel stereo matching method based on features extracted from image patches. These features extracted from the neural network describe the image patch so that the proposed method has strong robustness to different lighting conditions. According to these features, the similarity between image patches can be calculated. The matching cost is the opposite of similarity. For the scattered mismatching pixels that look like noise, the local processing method is employed to modify them. For the texture-less areas, where many mismatching pixels appeared, use the semi-global method to correct the mistakes. And for occlusion areas, where the object can be only observed in one of the two stereo images, use the left-right consistency check method to complete. The disparity of background areas replaces the disparity of occlusion areas. After extracting features from the neural network and performing post-processing for different mismatching pixels, the output disparity map of the proposed method has a good performance on the Middlebury dataset.
2. Proposed Method

In this section, we detail our stereo matching method. The pipeline of our method is shown in Fig. 1. Our methods follow the same pipeline as traditional stereo matching. Firstly, according to the correlation between stereo images, the matching cost can be calculated based on the features extracted from CNN. Next, the matching cost is aggregated so that mismatching pixels can be modified. Finally, we perform post-processing to obtain a disparity map with high accuracy.

When training the neural network, the input is an image patch. But when testing, a full image can be used as input to obtain features that describe each pixel.

2.1 Matching Cost Calculation

The traditional method calculates the matching cost based on a self-designed energy function, which cannot adapt to different lighting conditions. The overall architecture of the proposed neural network is shown in Fig. 2. This work calculates the matching cost based on features extracts from CNN, which is robust to different lighting conditions. The overall architecture of the proposed neural network is shown in Fig. 2. Firstly, input an image patch into a convolutional layer. Next, these features describe the image patch that will be extracted after two multi-scale blocks. About multi-scale block, 4 kinds of features can be extracted from the input. All the features are concatenated, and 3 convolutional layers without padding are performed to make a better feature extraction. The input image patch size is 11×11, and the size of output features is 1×64.

When training the network, it is considered a binary classification problem. For each image patch from the left image, extract a negative image patch and a positive image patch from the right image. Therefore, a negative stereo image patch sample and a positive stereo image patch sample are created for training. The well-described features are extracted by training the network to distinguish positive and negative samples.

The matching cost is computed directly from the features that output from the network.

\[ C = \sum (f_l * f_r) \]  \hspace{1cm} (1)

Where \( f_l \) denotes the feature of left image patch, and \( f_r \) denotes the feature of right image patch. The loss function is defined as

\[ L = \max(0, m - P_1 + P_0) \]  \hspace{1cm} (2)

Where \( P_0 \) denotes the matching cost of negative sample, and \( P_1 \) denoted the matching cost of positive sample. The margin \( m \) between positive sample and negative sample is a hyper-parameters, which is initialized to 0.2. When training the network, if matching cost of negative sample is \( m \) higher than the matching cost of positive sample, we think that the network can distinguish different samples. Therefore, the network pays more attention on the distinguishing of samples with difference between 0.2.
2.2 Matching Cost Aggregation

For each pixel, the network extract features only based on the information of the image patch, instead of full image. Therefore, some mismatching pixels exist, especially in texture-less areas. The matching cost calculated in Sec. 2.1 cannot be used directly to disparity prediction and further modifications are necessary. Therefore, different matching cost aggregation methods are used to modify different kinds of mismatching pixels.

Scattered mismatching pixels is very similar to the image’s noise, so it can be modified using the image smoothing method. Therefore, an adaptive cross window is designed to modify scattered mismatching pixels, as shown in Fig. 3(a). For pixel $p$, the control condition of the adaptive cross window is

$$\begin{align*}
|I(p) - I(q)| < \tau \\
\|p - q\| < \eta
\end{align*}$$

The distance between $p$ and $q$ should be less than $\tau$. The difference between $p$ and $q$ in input image should be less than $\eta$. Each pixel will get an adaptive cross window $W$, as shown in the right part of Fig. 3(a). The matching cost of each pixel will be aggregated based on the matching cost of each pixel in its adaptive cross windows

$$C'(p, d) = \frac{1}{n} \sum_{i \in W} C_i(p, d)$$

There are many pixels mismatched in the texture-less areas because the pixels are very similar to each other. These mismatched pixel cannot be modified base on the nearby pixels in its adaptive cross window. Therefore, A larger region-of-interesting around the target pixel should be under consideration. Thus the semi-global method is used for modification. For adjacent pixels, if the image intensity between them are very close, their disparity will be close too. Therefore, we refine the matching cost by enforcing smoothness constraints in texture-less areas. The principle of matching cost aggregation is shown in Fig. 3(b). In direction $r$, the semi-global aggregation method of matching cost is

$$L_r(p, d) = C(p, d) + \min(L_r(p - r, d),$$

$$L_r(p - r, d - 1) + P_1, \min[L_r(p - r, d + 1) + P_1,$$

$$\min[L_r(p - r, i) + P_2]\} - \min[L_r(p - r, k)]$$

Where $C$ denotes the matching cost of pixel $p$ before semi-global aggregation. When the difference between adjacent pixels is 1, a penalty $P_1$ will be added. When the difference between adjacent pixels is larger than 1, a penalty $P_2$ will be added. Moreover, the parameters $P_1$ and $P_2$ are set according to the gradient of the image.

Finally, we refine the matching cost by combining the four directions of up, down, left, and right

$$C'(p, d) = \frac{1}{n} \sum_r L_r(p, d)$$

2.3 Disparity Prediction and Post-processing

After the matching cost is aggregated, most of the mismatching pixels are modified. The distance in the horizontal direction with the smallest matching cost is considered as the predicted disparity.

The occlusion area means a set of pixels can only be seen in one image but not in the other one. This phenomenon happened because nearby objects shade distant objects. Therefore, the occlusion area is the background area. If the object cannot be found in one image, it must appear in the other image. The disparity of pixels in the occlusion areas can be modified by performing a left-right consistency check.

We divide all the pixels into four classes: matching, occlusion, mismatching, and no matching, as shown in Fig. 4. For each class, modify the disparity with a different method:
Table 1. Comparison of error rate and RMS error during stereo matching

<table>
<thead>
<tr>
<th>Processing method</th>
<th>all area</th>
<th></th>
<th></th>
<th></th>
<th>all area</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&gt;0.5</td>
<td>&gt;1.0</td>
<td>&gt;2.0</td>
<td>&gt;4.0</td>
<td></td>
<td>&gt;0.5</td>
<td>&gt;1.0</td>
<td>&gt;2.0</td>
<td>&gt;4.0</td>
</tr>
<tr>
<td>CNN</td>
<td>24.07</td>
<td>17.80</td>
<td>15.69</td>
<td>13.64</td>
<td>14.42</td>
<td>15.55</td>
<td>9.05</td>
<td>7.79</td>
<td>7.16</td>
</tr>
<tr>
<td>2nd time aggregation</td>
<td>18.23</td>
<td>12.88</td>
<td>10.58</td>
<td>8.28</td>
<td>9.89</td>
<td>9.63</td>
<td>4.40</td>
<td>3.28</td>
<td>2.66</td>
</tr>
<tr>
<td>3rd time aggregation</td>
<td>17.90</td>
<td>12.28</td>
<td>9.99</td>
<td>7.70</td>
<td>9.78</td>
<td>9.43</td>
<td>3.94</td>
<td>2.82</td>
<td>2.27</td>
</tr>
<tr>
<td>post-processing</td>
<td>16.09</td>
<td>10.36</td>
<td>7.49</td>
<td>4.73</td>
<td>4.40</td>
<td>8.89</td>
<td>3.70</td>
<td>2.58</td>
<td>1.94</td>
</tr>
</tbody>
</table>

3. Experiments

3.1 Datasets

We evaluate our method on the Middlebury dataset. Because it is difficult to obtain the ground truth of the stereo image, the dataset is small. To obtain a large amount of data for training, the image is scaled and rotated before the image patch is extracted. We predict the disparity of each pixel and calculate the error rate with different error tolerance.

3.2 Implementation details

We train the proposed network by using gradient descent with momentum method. When the neural network parameters continue to update in the same direction, the speed update is faster than the stochastic gradient descent method, which can speed up the network training speed. When the parameters update direction changes suddenly, the method can move forward for a while due to the large momentum. Therefore, some local best points and saddle points will be dropped out, and the possibility of finding the global best point will increase. In addition, we train our network with an exponential decay learning rate so that the global best point can be found. The network is build based on TensorFlow and trained on a GPU with 6GB memory.

Three times of matching cost aggregation are operated, the first and third time are based on adaptive windows, and the second time is based on semi-global method. The operation of aggregation based on adaptive windows before and after aggregation based on semi-global method can avoid the influence of noise on it and remove the noise it brings. The calculation cost of aggregation based on adaptive windows is small, so it only takes a little time to perform it twice.

3.3 Result

According to the standard evaluation metrics of Middlebury dataset, we evaluate the proposed method, as shown in
Table. [1] With different tolerance errors, the error rate and RMS error of the disparity map keep decreasing with stereo matching. When the tolerance error is 4 pixels, the error rate of the disparity map is 4.73. By comparing the error rate and RMS error after CNN in non-occlusion areas with all areas, the results show that stereo matching based on image patch features can achieve high accuracy disparity prediction in the non-occlusion areas. The disparity map output from the proposed method is shown in Fig. [5]. According to the error mask of the edge area of the bear, there are only a few mismatching pixels, and the proposed method can achieve high accuracy disparity prediction.

4. Conclusions

This paper proposed a stereo matching method based on features of image patch. The result on the Middlebury dataset suggests that stereo matching based on features extracted from the image patch can perform well and have a high accuracy in the non-occlusion area. Employing the correct disparity in the non-occlusion area, the occlusion area’s disparity can be modified with adaptive cross window and semi-global method. However, our method is not yet suitable for real-time applications, such as autonomous driving. Future work can focus on improving the post-processing method to achieve real-time application.

References