The Visual Computer (2021) 37:411–419
https://doi.org/10.1007/s00371-020-01811-x

ORIGINAL ARTICLE

A simplified ICA-based local similarity stereo matching

Suting Chen[1,2] · Jinglin Zhang[1] · Meng Jin[1]

Published online: 15 February 2020

Abstract
Since the existing stereo matching methods may fail in the regions of non-textures, boundaries and tiny details, a simplified independent component correlation algorithm (ICA)-based local similarity stereo matching algorithm is proposed. In order to improve the DispNetC, the proposed algorithm first offers the simplified independent component correlation algorithm (SICA) cost aggregation. Then, the algorithm introduces the matching cost volume pyramid, which simplifies the preprocessing process for the ICA. Also, the SICA loss function is defined. Next, the region-wise loss function combined with the pixel-wise loss function is defined as a local similarity loss function to improve the spatial structure of the disparity map. Finally, the SICA loss function is combined with the local similarity loss function, which is defined to estimate the disparity map and to compensate the edge information of the disparity map. Experimental results on KITTI dataset show that the average absolute error of the proposed algorithm is about 37% lower than that of the DispNetC, and its runtime consuming is about 0.6 s lower than that of GC-Net.

Keywords Stereo matching · Cost aggregation · Independent component correlation · Region-wise loss function

1 Introduction

Stereo matching as a critical part of stereo vision has been extensively used in the fields of autonomous driving, object detection and 3D reconstruction [1]. Stereo matching is intended to solve the corresponding relationship between left and right pixels in a stereo image pair to obtain the disparity map [2]. Classical stereo matching algorithms consist of four components: matching cost calculation, cost aggregation, disparity calculation and disparity refinement [3]. However, stereo matching is highly challenging because complex scenarios including occlusion, textureless and disparity discontinuity make it difficult to obtain dense and precise disparity map. As a result, it is of great significance to accurately obtain dense disparity from a stereo image pair.

Generally, stereo matching algorithms can be divided into two categories: conventional algorithms and deep learning-based algorithms. Specifically, conventional stereo matching algorithms can be subdivided into global stereo matching [4], local stereo matching [5–7], and semi-global stereo matching [8]. The effect of global stereo matching depends on accuracy of matching cost, and the calculation process is very slow as the disparity is solved by global energy function. On the contrary, local matching algorithm can match local properties within a certain range by comparing matching points, and consequently, it depends heavily on the rationality of the matching window and processes textureless areas poorly. Additionally, semi-global stereo matching algorithm is the ensemble of global matching and local matching that should consider all disparity changes and dynamically plan an optimal path to minimize the energy function; however, the convergence rate of such algorithms tends to be slower when implemented. In conventional stereo matching algorithms, the design of manually extracting image features and cost volume leads to inadequate expression of image

✉ Suting Chen
sutingchen@nuist.edu.cn

Jinglin Zhang
574428734@qq.com

Meng Jin
jinmeng0722@gmail.com

[1] Jiangsu Key Laboratory of Meteorological Observation and Information Processing, Nanjing University of Information Science and Technology, Nanjing 210044, China

[2] Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET), Nanjing University of Information Science and Technology, Nanjing 210044, China

information, which affects the execution of subsequent steps and the accuracy of the disparity map.

Stereo matching algorithm based on deep convolutional neural networks (DCNN) [9] has recently been developed to a large extent. It constructs the model of the stereo matching problem using the DCNN, and constructs the matching cost volume with stereo image pair as input. This practice can avoid the artificially designed matching cost function and manually extracted properties from being partial, which not only effectively improves the accuracy, but also reduces the computational complexity. Compared to conventional algorithms, the DCNN-based methods based on CNN to obtain disparity map have been significantly improved on both accuracy and speed [10].

DispNetC [11] uses the correlation layer to generate the initial matching cost volume and then uses the encoder-decoder structure to process the initial matching cost volume and finally uses the single pixel point loss function for disparity prediction. In this paper, the SILSSM algorithm is proposed. The cost aggregation step is introduced to the classic DispNetC, the matching cost volume pyramid and the SICA loss function are combined; and local similarity loss function is redefined based on the region-wise loss function combined with the conventional pixel-wise loss function; finally, the SICA loss function and local similarity loss function are combined to perform network training. This practice not only ensures the estimation speed of the disparity map, but also improves the estimation accuracy of the disparity map edge and the detail and reduces dependence on single pixel during the estimation process.

2 Related work

Stereo matching based on DCNN generally obtains aggregation cost to complete the disparity estimation by manually designing the matching cost and simply accumulating the initial matching cost of the pixels in the window. In the literature [12], the feature points on the image to be matched were treated as the matching support points to make a triangulation on the support points and run interpolation calculation against the disparity, but the effect of the obtained disparity map was general. In the literature [13], stereo matching algorithm based on an adaptive weight was proposed, and the impact of the pixels with remote special distance and large color difference was weakened according to the aggregation weight of the pixels in the support window identified from the color similarity and geometrical distance between pixel points. In the literature [14], the initial matching cost calculation and cost aggregation were further examined, the mutual information of left and right images was added to the matching cost, and the matching cost along the multi-directional path of the center pixel was accumulated using dynamic programming algorithm (DP), thereby significantly improving the accuracy of local stereo matching algorithm. A modified cross-based cost aggregation [2, 15] was then proposed to construct the adaptive irregular support window by extending to the horizontal and vertical directions of the center pixel according to color changes, which replaces conventional cost aggregation methods with high calculation load to a certain extent and achieves satisfying results. Because conventional stereo matching algorithm often involves several steps, its optimization will be limited. In addition, partial extraction of information may easily lead to missing of spatial structure, and the support window generation and the calculation of aggregation cost would be a problem [16].

The first category of stereo matching algorithms based on DCNN is the matching cost learning. In the literature [17], the convolutional neural network for the matching cost volume constructed with MC-CNN was proposed, whose the matching cost was calculated using a deep Siamese network; this proposed network can be trained to measure the similarity of a pair of I_l image patch. In the literature [18], based on the MC-CNN, the matching cost volume was combined with the input color information using a federated filter to complete the cost aggregation until the final matching cost volume was obtained. Content-CNN [19], a network that can directly estimate disparity value, was proposed in the literature [19]. This method is similar to MC-CNN in terms of structure and idea, which can calculate the local matching cost by regarding disparity estimation as a multi-label classification task. However, these methods do not really use deep learning method to learn geometric features, but only use CNN to measure the similarity between image blocks and learn the matching cost. The involved steps in general are similar to conventional local algorithms. Admittedly, these modifications can improve the disparity estimation outcome, but the processing steps that follow still use conventional algorithms, which would affect the prediction accuracy of disparity map and it will be adverse to the overall network optimization.

The second category of stereo matching algorithms based on DCNN is the end-to-end training networks for the estimation of disparity, where the four steps of stereo matching are integrated into one network to reduce manual participation and increase the overall fitting of the model. DispNetC is a classic end-to-end disparity estimation network with the similar network structure as Flownet [20]. To generalize the use of optical flow estimation to disparity estimation, a correlation layer was used to generate matching cost volume that was then further processed by the network. In the literature [21], a cascading residual learning (CRL) network with more input information was proposed, which can extend the network structure of Dispnet and is formed by cascading DispFullnet and DispResNet, so that the detail part of the disparity map can be modified. In the literature [22], EdgeSreteo [22] is with a similar structure as DispNet, the edge extraction network HED was further added, and the edge loss function was combined to refine the edge information of the disparity map. SegStereo [23] was designed based on DispNet and then further added with a semantic segmentation network; as a result, the segmented semantic features were merged with the feature map to make a full use of the structure information of the image while improving the estimation accuracy on the disparity map. GC-Net [24] is a new high-performance disparity estimation network proposed in the literature, which extended the 3D feature spectra to 4D and added a new disparity dimension, as well as operated the matching cost volume on the disparity dimension using 3D convolution without extra post-processing operations. In the literature [25], a significant cost aggregation process was added based on the GC-Net, which can be divided into two branches: the estimation branch that is used to generate the cost aggregation result and the guiding branch that is used to extract information from the image using 2D convolution.

The end-to-end training disparity estimation network can improve the accuracy of estimation on the disparity map, but there are still some deficiencies. Firstly, the cascading structure or the 3D convolution-based structure for cost aggregation involves a high load of calculation at a slow speed; secondly, CNN may lose a part of the image edge information when acquiring features, resulting in inaccurate estimation on the disparity of the detail region or the boundary. When the foreground pixel and the background pixel approach or mix each other, the spatial discrimination ability will get lost. In the literature [26], an interaction-aware spatiotemporal pyramid attention mechanism was proposed, where the weights assigned by the correlation between channel features was used to extract salient features for face recognition. To use this attention mechanism for cost aggregation, the weights of pixels under different disparity scenarios may be considered without increasing the calculation load. In the literature [27], region-wise loss function was integrated based on single pixel loss function, so that the objects in the semantic segmentation have clear detail and edge information. The attempt to combine the region-wise loss function and the loss function for disparity estimation can compensate the near-boundary local information in estimation

3 SILSSM algorithm

The scheme of SILSSM algorithm is shown in Fig. 1. The specific steps are described below. Firstly, the algorithm inputs the stereo image pair, and then it extracts the feature of each pixel and constructs the matching cost volume by correlation operations, until the initial matching cost calculation is completed. Secondly, it sends the initial matching cost into the encoder-decoder structure. In the decoding part, the matching cost volumes of different layers are stacked for the SICA cost aggregation, and the defined SICA loss function is combined to obtain matching cost weights. Then, the aggregated matching cost volume is sent into the last deconvolution layer to estimate the full-size disparity map by combining local similarity loss function and the SICA loss function. Finally, the predicted disparity map is output.

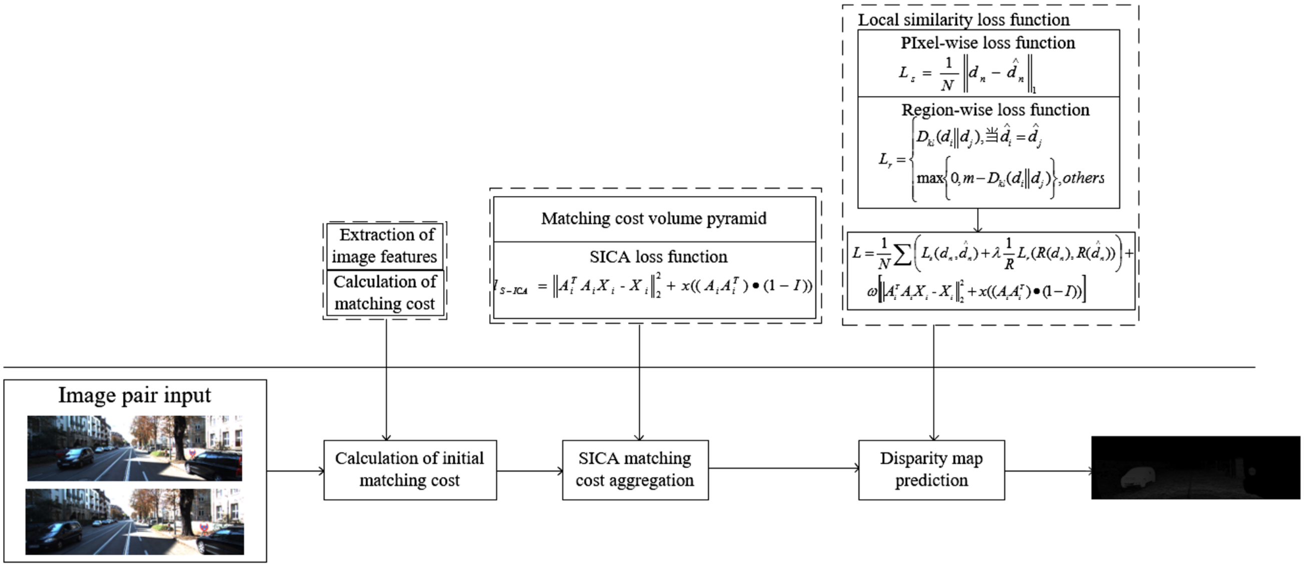
Local similarity loss function
PIxel-wise loss function
$L_s = \frac{1}{N}\left\|d_n - \hat{d}_n\right\|_1$
Region-wise loss function
$L_r = \begin{cases} D_{kl}(d_i \| d_j), 当 \hat{d}_i = \hat{d}_j \\ \max\left\{0, m - D_{kl}(d_i \| d_j)\right\}, others \end{cases}$
$L = \frac{1}{N}\sum\left(L_s(d_n, \hat{d}_n) + \lambda\frac{1}{R}L_r(R(d_n), R(\hat{d}_n))\right) +$
$\omega\left[\left\|A_i^T A_i X_i - X_i\right\|_2^2 + x((A_i A_i^T)\bullet(1-I))\right]$
Extraction of image features
Calculation of matching cost
Matching cost volume pyramid
SICA loss function
$l_{S-ICA} = \left\|A_i^T A_i X_i - X_i\right\|_2^2 + x((A_i A_i^T)\bullet(1-I))$
Image pair input
Calculation of initial matching cost
SICA matching cost aggregation
Disparity map prediction

Fig. 1 SILSSM algorithm scheme

3.1 Calculation of the initial matching cost

(1) Extraction of image features

To compare the similarity of two pixels, it is required to obtain a strong expression of each pixel; the image features I_r and y on the input pair F_l and F_r are extracted using CNN to prepare for the construction of matching cost.

(2) Calculation of the matching cost

Input F_c and F_c to the correlation layer to obtain the relationship between the relative location of $M_i^{'}$ and $M_i^{'}$ in the feature space, obtain the initial matching cost y and complete the conversion from the expression of features to the measurement of the pixel similarity.

3.2 SICA matching cost aggregation

The cost aggregation based on SICA is completed in the decoder part by the spatial pyramid formed from the stack of matching cost volumes, combined with the SICA loss function. It is devised to complete the measurement of the importance of this pixel with its neighboring pixels within the entire range of disparity search and complete the update of weights by using the correlation between channel vectors.

The matching cost volumes are stacked to form a spatial pyramid and to obtain the weights of channel by channel vector. The specific step is shown in Fig. 2:

Step 1 Stacking deconvolution results of different layers forms a spatial pyramid; upsampling the stacked deconvolution layers and the sampled size is the same as the final deconvolution size $f_j^{'}$, $f_j^{'} = \begin{cases} f_i, & j = i \\ \Re(f_i), & j = 1, 2, \ldots, i-1 \end{cases}$

Step 2 Keeping the channel number of $f_j^{'}$ constant flattens $f_j^{'}$ into $\mathrm{X_j} \in R^{\mathrm{W_i}H_i \times \mathrm{d_j}}$, in which X_j is composed of vector $h_{\mathrm{k}}^{\mathrm{i}} \in R^{w \times h \times c}$ in each position.

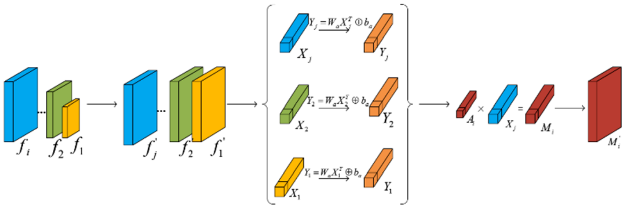

Fig. 2 Steps for construction of matching cost pyramid

Step 3 Obtain a weight matrix Y_j from the flattened X_j and Y_j which is the sum of the weights of all pixels of the channel vector h_k^i, $Y_j = W_a X_j^T \oplus b_a$.

Step 4 Normalize softmax to obtain the normalized weight matrix $A_i = \left[a_1^{iT}, a_2^{iT}, \ldots, a_{W_i H_i}^{iT}\right]$. a^i is calculated by: $a^i = \mathrm{softmax}(\Gamma(y^1, \ldots, y^i))$, where Γ is the fusion function using element-wise sum.

Step 5 Multiple A_i by X_j to obtain the aggregated vector M_i, $M_i = A_i X_j$. Transform the vector after cost aggregation $M_i \in R^{W_i H_i \times d_i}$ into the cost volume $M_i^{'} \in R^{W_i \times H_i \times d_i}$.

The acquisition of the weight of channel as mentioned above can be regarded as a simplified ICA process: X_j can be regarded as the signal that requires dimension reduction in the ICA process; $\oplus b_a$, the weight obtained from $Y_j = W_a X_j^T \oplus b_a$, can be treated as the centralizing step in the ICA process when calculating the weight; the weight matrix A_i corresponds to the transformation matrix W in ICA, assigning a high weight to a core part in the matching cost volume $f_j^{'}$ is similar to the process of extracting key components from ICA.

However, the weight A_i at this point is just obtained by weighting the channel vector $h_{\mathrm{k}}^{\mathrm{i}}$ itself, without considering the influence of peripheral pixels. In the literature [26], the loss function inspired by principal component analysis (PCA) was used. For the extraction performance of features, when the number of independent principal components is more, the stability of principal components corresponding to the feature value gets worse; however, the inference speed of PCA is not efficient. For independent component correlation algorithm (ICA), when the number of independent principal components is more, its recognition speed tends to be more efficient [28]. Given the complex structure of the disparity map, more objects are involved and more principal components need to be extracted, and thus this paper replaces PCA loss function with ICA reconstruction loss function. ICA reconstruction lost function is $\left\|W^T W x - x\right\|_2^2$, and the orthonormality constraint is $WW^T = I$, where W means the mapping matrix, mapping original data x to the feature.

Given the correlation between local pixels, the SICA loss function is combined here, the relationship between channels is used, that is, the similarity of different pixels under the same disparity case, to complete the cost aggregation. The matching cost volume processed with ICA can extract the main pixels from it, and the validity and reliability of weights can be further enhanced by assigning greater weights to these pixels, combined with the interactive information between channel vectors $h_{\mathrm{k}}^{\mathrm{i}}$. SICA loss function can be defined as:

$$l_{\mathrm{SICA}} = \left\|A_i^T A_i X_i - X_i\right\|_2^2 + x\left(\left(A_i A_i^T\right) \cdot (1 - I)\right) \tag{1}$$

where I means the unit matrix and x means the sum of square function. The weight matrix A_i corresponds to the mapping matrix W in the ICA reconstruction loss function and X_i corresponds to the original data x.The SICA loss function is combined with the spatial pyramid structure to form a complete cost aggregation process.

3.3 Disparity map prediction

The aggregated matching cost volume is continuously sent to the decoder part and then combined with local similarity loss function, and the detail and the edge of the disparity map are modified. The diagram of local similarity loss function is shown in Fig. 3.

In stereo matching, pixel-wise loss function is usually used and the difference between the estimated disparity map and the ground truth disparity map is calculated and used as training loss. The loss function for a single pixel in DispNetC is $L_s = \frac{1}{N}\left\|d_n - \overset{\wedge}{d_n}\right\|_1$, where N is the number of pixels and d_n, $\overset{\wedge}{d_n}$ are the estimated disparity and the ground truth disparity of the n_{th} pixel. This loss function only considers the factor of single pixel, instead of the image structure and the semantic information between neighboring pixels between labels. The disparity map generated by the decoder part can obtain a relatively accurate effect from textureless region, but the edge of an object is often obscure and the details cannot be shown. Therefore, region-wise loss function is used to make up for the deiciencies of pixel-wise loss function. The use of region-wise loss funtion can extend the dependence on independent pixels to the dependence on neighboring pixels information.

Region-wise loss function [27] can be written as:

$$L_r = \begin{cases} D_{kl}\left(d_i \middle\| d_j\right), \text{ when } \overset{\wedge}{d_i} = \overset{\wedge}{d_j} \\ \max\left\{0, m - D_{kl}(d_i \| d_j)\right\}, \text{ others} \end{cases} \tag{2}$$

where $D_{kl}()$ means the Kullback–Leibler divergence and offers a way to quantify the difference between the two distributions P and Q. Region-wise loss function measures the similarity between two neighboring pixels with KL divergence. There are two types of label relationships between pixel i and neighboring pixel j: whether their labels are the same or different. When the ground truth of i and j are same, then the loss of the estimated disparity of i and j should be made as small as possible. If the ground truth of i and j are different, then the loss of the estimated disparity of i and j should be made as big as possible.

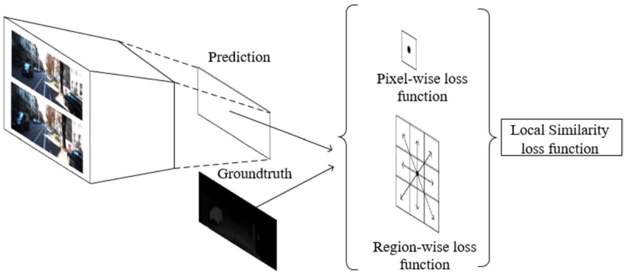

Fig. 3 Diagram of local similarity loss function

Local similarity loss function can be defined as:

$$L_l = \frac{1}{N}\sum\left(L_s\left(d_n, \widehat{d}_n\right) + \lambda\frac{1}{R}L_r\left(R(d_n), R(\widehat{d}_n)\right)\right) \tag{3}$$

where N is the total number of pixels. In the region loss function L_r, $R(d_n)$ means the estimated disparity value in the region, $R(\widehat{d}_n)$ means the ground truth in the region, n means the center pixel in the region, in which $R()$ means a 3×3 neighborhood.

In a word, the SICA loss function is combined with local similarity loss function, and then total loss function can be defined as:

$$\begin{aligned} L &= L_l + \omega L_{\text{SICA}} \\ &= \frac{1}{N}\sum\left(L_s\left(d_n, \widehat{d}_n\right) + \lambda\frac{1}{R}L_r\left(R(d_n), R(\widehat{d}_n)\right)\right) \\ &+\omega\left[\left\|A_i^T A_i X_i - X_i\right\|_2^2 + x\left((A_i A_i^T)\cdot(1-I)\right)\right] \end{aligned} \tag{4}$$

wherein ω is the weight parameter, which is used to control the ratio of importance of the SICA loss function to the local similarity loss function. In this paper, the learning process of the proposed SILSSM algorithm is explained in Algorithm 1. The proposed method is trained with ground truth depth data by supervised learning. As the entire loss function is differentiable, the network is trained using backpropagation algorithm in order to minimize the loss function. The disparity map can be estimated by training the loss function.

```
Algorithm 1: SILSSM algorithm

Input   Stereo image pair $I_l$ and $I_r$.
Output  Network parameters and disparity map $y$.
1:      Calculate depth features $F_l$ and $F_r$ based on network structure.
2:      Calculate the initial matching cost $F_c$.
3:      Send $F_c$ to encoder–decoder structure and calculate the
          weighted matching cost volume $M_i'$ in the decoder part.
4:      The matching cost volume $M_i'$ is sent to the last layer of the
          decoder part and output the disparity map $y$.
5:      Calculate the derivative of Eq. (3) and update the network
          parameters using backpropagation algorithm.
6:      Repeat steps 1 to 4 until the parameter values have not
          changed.
```

4 Experimental results

4.1 Dataset and network training

This proposed algorithm is written in Python, and its training and testing are implemented on Tensorflow. All experiments are conducted on NVIDIA Tesla V100 GPU. The effect of the proposed method is validated on Scene Flow and KITTI binocular image dataset. The ground truth values provided by the KITTI dataset are sparse, that is, only the disparity values of certain points on the original image are provided. Consequently, the training process is first on Scene Flow dataset. Scene Flow training set is a synthesized dataset containing 35,454 pairs of training images, 4370 pairs of test images, the dataset is so large that over-fitting will not occur during the training process. The real-world KITTI dataset includes two subsets with sparse ground truth disparities. KITTI 2012 contains 194 training and 195 test image pairs, while KITTI 2015 consists of 200 training and 200 test image pairs. After a certain number of training–validation repeated experiments, the iteration and learning rate will be set as 100 k and 0.01, respectively. To perform the matching cost calculation, the max displacement disparity will be set to 100.

4.2 Analysis of algorithm performance

The itemized validation of this network is analyzed on Scene Flow. The proposed network structure is similar to the DispNetC. In the method of DispNetC, the disparity map is directly estimated using the pixel-wise loss function in the decoder part. In this paper, the SICA matching cost aggregation and local similarity loss function are added on the DispNetC encoder–decoder structure to make the edge of the disparity map clearer and the estimated accuracy of disparity higher. Table 1 analyzes the advantages of the SICA loss function and local similarity loss function on Scene Flow and compares the PCA cost aggregation [26] and the SICA cost aggregation.

Metrics include error > 1px, error > 3px, mean absolute error (MAE) and running time. Wherein, error > t px means the percentage of erroneous pixels; when the endpoint error (EPE) of the estimated disparity of a pixel is greater than t pixels, this pixel would be considered to be erroneously estimated. EPE means the average Euclidean distance between the estimated disparity and the ground truth. MAE means the average value of the estimated disparity and the true error.

Table 1 Algorithm itemized validation

Model	>1px	>3px	MAE	Time
DispNetC	11.3	7.2	4.0	0.06
+PCA cost aggregation	10.7	7.1	3.9	0.19
+SICA cost aggregation	9.6	6.5	3.4	0.19
+Local similarity loss Function	10.0	6.6	3.1	0.13
Proposed	8.5	5.2	2.5	0.29

As is seen from Table 1, if the cost aggregation is only considered, the MAE in the case of using the SICA cost aggregation reduces by 12% when compared with the PCA cost aggregation, and the use of the SICA cost aggregation has little influence on the running time. The running time of the proposed network framework is 0.23 s slower than that of the DispNetC, but the MAE of the former reduces by about 37%, which provides a theoretical basis for future stereo matching in the shooting scenario.

Figure 4 plots the variation of the validation error during the training on the Scene Flow and compares the proposed loss function and the conventional pixel-wise loss function. As is seen from Fig. 4, the training speed of the conventional pixel-wise loss function is faster than that of the proposed loss function. However, the error rate of the proposed function is comparatively lower.

In Table 2, this proposed method is compared with GC-Net, EdgeStereo and SegStereo on KITTI2012 stereo dataset. In this table, "All" means that all pixels were considered in error estimation, whereas "Noc" means that only the pixels in non-occluded regions were taken into account. In KITTI2012, error > 2px, error > 3px, error > 5px and running time are calculated in the case of occlusion and no occlusion. In Table 3, this proposed framework is compared with GC-Net, EdgeStereo and DispNetC on KITTI2015 stereo dataset. In this table, the three columns "D1-bg," "D1-fg" and "D1-all" mean that the pixels in the background, foreground, and all areas. In KITTI2015, the performance assessment is performed in the case of occlusion and no occlusion, and the three-pixel-error (3PE) was calculated in the case "D1-bg," "D1-fg" and "D1-all," respectively. 3PE

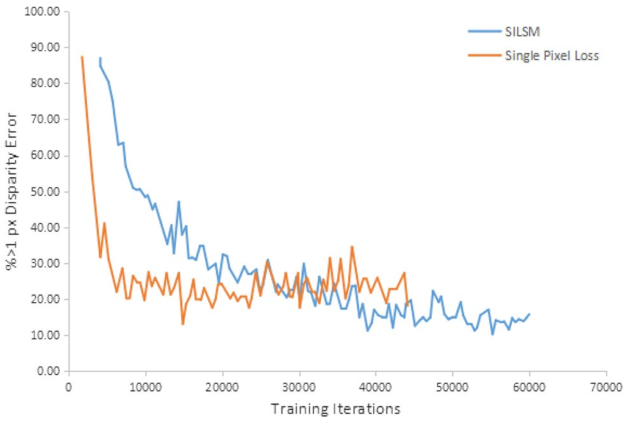

Fig. 4 Validation error

Table 2 Results on KITTI 2012 stereo benchmark

Model	>2px		>3px		>5px		Time
	Non-occ	All	Non-occ	All	Non-occ	All	
GC-Net	2.70	3.46	1.77	2.30	1.15	1.47	0.90
EdgeStereo	2.64	3.33	1.72	2.15	1.05	1.30	0.31
SegStereo	3.27	3.80	2.47	2.87	1.09	1.29	0.68
Proposed	**2.66**	**3.26**	**1.67**	**2.27**	**1.05**	**1.29**	**0.29**

In experiments, bold performs better than other networks

Table 3 Results on KITTI 2015 stereo benchmark

Model	All pixels			Non-occluded pixels			Time
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	
GC-Net	**2.21**	6.16	2.87	2.02	5.58	2.61	0.91
EdgeStereo	2.27	4.18	2.59	2.12	**3.84**	2.40	0.33
DispNetC	4.32	4.41	4.34	4.11	3.72	4.05	0.06
Proposed	2.31	**4.05**	**2.46**	**2.02**	3.89	**2.31**	**0.30**

In experiments, bold performs better than other networks

means the percentage of pixels with endpoint error more than 3.

As is seen from Table 2, the matching accuracy of the proposed method is the highest. Given all pixels and when the error threshold is 2px, the matching accuracy of the proposed algorithm is about 0.2 and 0.07% higher than that of GC-Net and EdgeStereo, respectively. Of which, the matching accuracy of SegStereo is the lowest and is about 0.54% lower than that of the proposed method. In addition, the running speed of SILSSM algorithm is about 3 times of GC-Net, GC-Net can process the matching cost volume with 3D convolution to extract the geometry and context information, thereby significantly increasing the calculation complexity. In contrast, SILSSM algorithm uses SICA cost aggregation combined with local similarity loss function to modify the spatial structure of the disparity map without increasing excessive calculations.

As is seen from Table 3, compared with the DispNetC, the proposed algorithm adds an obvious cost aggregation step. Despite the reduced running speed, its accuracy is improved substantially. Compared with GC-Net and EdgeStereo, the accuracy of the proposed method increases by 0.41 and 0.13%, respectively, under the conditions of all pixels D1-all. EdgeStereo is further added with a network for the extraction of edge information. Compared with the proposed method, its estimation accuracy for the background region of all pixels is about 0.06% higher. However, its running time is 30 ms slower due to the addition of an extra network.

4.3 Subjective quality evaluation

The images of KITTI2012 and 2015 stereo datasets are shot in the actual outdoor driving. The quality of the disparity map is a direct reflection of the performance of the algorithm. Therefore, the KITTI dataset is used to perform subjective quality evaluation. Figure 5 shows the qualitative comparison results among the proposed algorithm framework and PSMNet, GC-Net, SegStereo and DispNetC on KITTI2015.

It can be observed from Fig. 5 that the proposed method can generate smoother disparity prediction results. The edges of traffic lights, traffic signs and other objects remain intact, and the details are clearer. Compared with DispNetC, the proposed method is able to handle challenging scenarios and ensure the integrity of the disparity map structure. The prediction of objects such as traffic lights and traffic signs are better than that of the other three methods, and the details are more discriminative.

5 Conclusion

In this paper, the network structure and objective function of DispNetC are modified to propose SILSSM algorithm. The edge and details of the disparity map are refined by using the SICA cost aggregation and the local similarity loss function. The experimental results show that the proposed method can estimate the edge of the disparity map on KITTI dataset clearly. In addition, this proposed algorithm runs at a fast rate and its running speed improves by 0.6 and 0.3 s, respectively, compared to GC-Net and EdgeStereo. The subsequent research will be intended to further improve the estimation accuracy of the disparity map by the proposed algorithm while reducing its complexity.

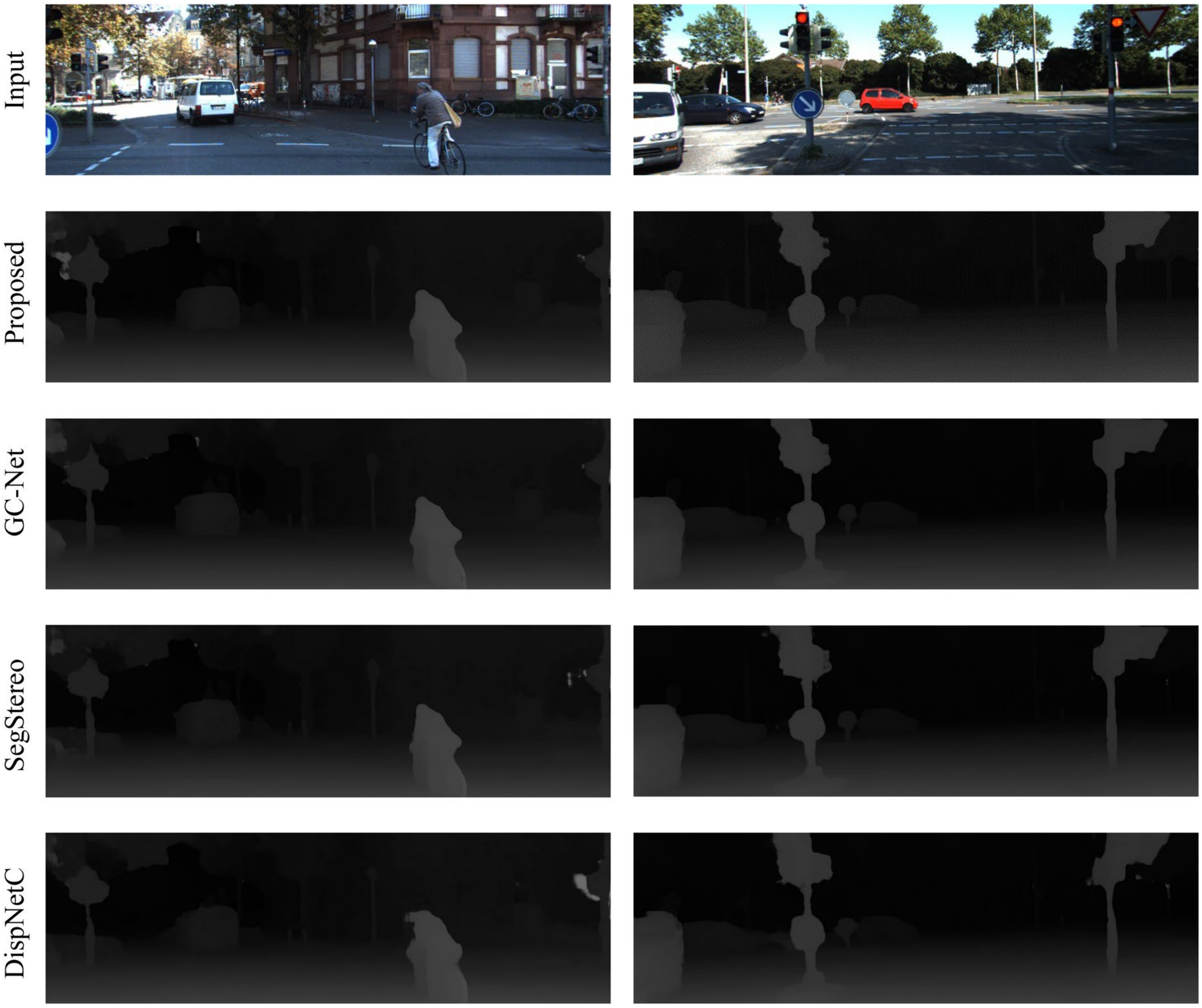

Fig. 5 Comparison of running results on KITTI2015

Acknowledgements This work is supported by National Natural Science Foundation of China (61906097).

References

1. Baldacci, A., Bernabei, D., Corsini, M., et al.: 3D reconstruction for featureless scenes with curvature h-ints. Vis. Comput. **32**(12), 1605–1620 (2016)
2. Cheng, F., Zhang, H., Sun, M., et al.: Cross-trees, edge and superpixel priors-based cost aggregation for Stereomatching. Pattern Recogn. **48**(7), 2269–2278 (2015)
3. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence Algorithms. Int. J. Comput. Vis. **47**(1–3), 7–42 (2002)
4. Tao, H., Sawhney, H.S., Kumar, R.: A global matching framework for stereo computation. In: International Conference on Computer Vision (ICCV2001), Vancouver, Canada (2001)
5. Zhang, K., Jiangbo, L., Lafruit, G.: Cross-based local stereo matching using orthogonal integral images. IEEE Trans. Circuits Syst. Video Technol. **19**, 1073–1080 (2009)
6. Hosni, A., Rhemann, C., Bleyer, M.: Fast cost-volume filtering for visual correspondence and beyond. IEEE Trans. Pattern Anal. Mach. Intell. **35**(2), 504–511 (2013)
7. Zhu, S., Yan, L.: Local stereo matching algorithm with efficient matching cost and adaptive guided i-mage filter. Vis. Comput. **33**(9), 1087–1102 (2016)
8. Hirschm, H.: Stereo processing by semiglobal matching and mutual information. IEEE Trans. Pattern Anal. Mach. Intell. **30**(2), 328–341 (2007)

9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision & Pattern Recognition (CVPR2016), Las Vegas, USA (2016)
10. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: Computer Vision & Pattern Recognition (CVPR2015), Boston, USA (2015)
11. Mayer, N., Ilg, E., Häusser, P., et al.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: Computer Vision & Pattern Recognition (CVPR2016), Las Vegas, USA (2016)
12. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: Asian Conference on Computer Vision (ACCV2010), Queenstown, New Zealand (2010)
13. Yoon, K.J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 650–656 (2006)
14. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: Computer Vision & Pattern Recognition (CVPR2005), CA, USA (2005)
15. Chang, X., Zhou, Z., Wang, L. et al.: Real-time accurate stereo matching using modified two-pass aggregation and winner-take-all guided dynamic programming. In: International Conference on 3D Imaging, Model-ing, Processing, Visualization and Transmission (3DIMPVT2011), Hangzhou, China (2011)
16. Yang, Q.: A non-local cost aggregation method for stereo matching. In: Computer Vision & Pattern Recognition (CVPR2012), Procidence, Rhode Island (2012)
17. Žbontar, J., Lecun, Y.S.: Matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res. **17**(2), 1–32 (2016)
18. Zhou, C., Zhang, H., Shen, X., et al.: Unsupervised learning of stereo matching. In: Computer Vision & Pattern Recognition (CVPR2017), Hawaii, USA (2017)
19. Luo, W., Schwing, A.G., Urtasun, R.: Efficient deep learning for stereo matching. In: Computer Vision and Pattern Recognition (CVPR2016), Las Vegas, USA (2016)
20. Fischer, P., Dosovitskiy, A., Ilg, E., et al.: Flow-net: learning optical flow with convolutional networks. In: International Conference on Computer Vision (ICCV2015), Santiago, Chile (2015)
21. Pang, J., Sun, W., Ren, J.S., et al.: Cascade residual learning: a two-stage convolutional neural network for stereo matching. In: Computer Vision and Pattern Recognition (CVPR2017), Hawaii, USA (2017)
22. Song, X., Zhao, X., Hu, H., et al.: EdgeStereo: a context integrated residual pyramid network for stereo matching. In: Asian Conference on Computer Vision (ACCV2018), Perth Australia (2018)
23. Yang, G., Zhao, H., Shi, J., et al.: SegStereo: exploiting semantic information for disparity estimation. In: Computer Vision and Pattern Recognition (CVPR2018), Salt Lake City, USA (2018)
24. Kendall, A., Martirosyan, H., Dasgupta, S., et al.: End-to-end learning of geometry and context for deep stereo regression. In: Computer Vision and Pattern Recognition (CVPR2017), Hawaii, USA (2017)
25. Yu, L., Wang, Y., Wu, Y., et al.: Deep stereo matching with explicit cost aggregation sub-architecture. In: The National Conference on Artificial Intelligence (AAAI2018), Louisiana, USA (2018)
26. Du, Y., Yuan, C., Li, B., et al.: Interaction-aware spatio-temporal pyramid attention networks for action classification. In: European Conference on Computer Vision (ECCV2018), Munich, Germany (2018)
27. Ke, T.W., Hwang, J.J., Liu, Z., et al.: Adaptive affinity field for semantic segmentation. In: European Conference on Computer Vision (ECCV2018), Munich, Germany (2018)
28. Rui, T., Shen, C., Ding, J.: Comparison and analysis on ICA & PCA's ability in feature extraction. Pattern Recognit. Artif. Intell. **18**(1), 124–128 (2005)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Suting Chen received her Ph.D degree from Institute of Optics and Electronics, the Chinese Academy of Sciences in 2007. She is currently an assistant of professor at Nanjing University of Information Science & Technology. Her research interests include digital image processing and computer vision.

Jinglin Zhang is studying as a postgraduate in the Nanjing University of Information Science & Technology. Her current research interests include digital image processing and computer vision.

Meng Jin is studying as a postgraduate in the Nanjing University of Information Science & Technology. His current research interests include digital image processing and computer vision.