Multimedia Tools and Applications (2019) 78:29121–29135
https://doi.org/10.1007/s11042-018-6581-5

Prominent edge detection with deep metric expression and multi-scale features

Shulian Cai[1,2] · Jiabin Huang[1] · Jing Chen[2] · Yue Huang[1] · Xinghao Ding[1] · Delu Zeng[3]

Received: 2 May 2018 / Revised: 16 August 2018 / Accepted: 20 August 2018 /
Published online: 29 August 2018

Abstract

Edge detection is one of today's hottest computer vision issues with widely applications. It is beneficial for improving the capability of many vision systems, such as semantic segmentation, salient object detection and object recognition. Deep convolution neural networks (CNNs) recently have been employed to extract robust features, and have achieved a definite improvement. However, there is still a long run to study this hotspot with the main reason that CNNs-based approaches may cause the edges thicker. To address this problem, a novel semantic edge detection algorithm using multi-scale features is proposed. Our model is deep symmetrical metric learning network, which includes 3 key parts. Firstly, the deep detail layer, as a preprocessing layer and a guide module, is employed to remove some low-frequency information and still maintain the edge. Secondly, the deep encoder-decoder networks extract multi-scale features of original image, integrated for complementing information among each level feature. Finally, metric learning is introduced to generate a metric space used to predict edge result. It is easy to distinguish different categories, such as edge space and object space. Simulations and comparisons on benchmark datasets demonstrate the proposed algorithm is superior to the others through visual and quantitative evaluation, and specifically, the score of ODS reachs 0.788.

Keywords Edge detection · Guide filter · Convolution encoder-decoder network · Metric space · Multi-scale features

A preliminary version of this work appeared at ISPACS [4].

✉ Delu Zeng
dlzeng@scut.edu.cn

[1] School of Information Science and Technology, Xiamen University, Xiamen 361005, China

[2] Xiamen Key Laboratory of Mobile Multimedia Communications (Huaqiao University), Xiamen 361021, China

[3] School of Mathematics, South China University of Technology, Guangzhou 510641, China

1 Introduction

In the field of computer vision, edge detection is a hotspot. Its objective is to locate the boundaries from natural to distinguish the desired object from the background. Edge detection is usually regarded as low-level task, but it is widely used in high-level field nowadays. Meanwhile, edge detection has been demonstrated by many previous researchers to be so significant in many relative fields, i.e., object detection [10, 23, 45], image segmentation [5, 40, 51], object proposal [38, 49] and salient detection [42].

Generally, semantic edge detection can be separated into two approaches: primary (handcrafted based) approaches and deep-learned based approaches. For handcrafted methods [1, 7, 9, 27, 37], the researchers usually utilize several hand-made features (such as gradients, texture, brightness and colors) to attain edges of image, because image includes many important structures and background details. The researchers, like Sobel [37], Robert [33] and Prewitt [30], use filters to create gradient maps, and then construct an edge graph. The hand-made features are utilized as one of the most important cues in [1, 27]. Relying on hand-made features limits the inchoate edge detection method to capturing the semantic concepts of objects, although it has made great progress.

Furthermore, since deep convolution neural networks (CNNs) are able to construct a robust semantic features, it is widely employed in the field of pattern recognition and computer vision and has made great progress, i.e., image classification [19, 21, 31], semantic segmentation [25, 29], object recognition [12, 32, 46] , salient object detection [39, 48, 50] and feature extraction [43]. It's also employed to address the shortcoming of hand-crafted features in edge detection. For instance, HED [41], COB [26], Deepcontour [35] and DeepEdge [3] all construct the edge detection model based on the deep-learned. Though they have achieved several improvements on capability, there still exists plenty of space for development in these CNNs-based methods.

In previous works [41], multi-scale features have proved that it improves the accuracy of classification. Take this into account, we construct a deep edge detection model to extract robust multi-scale feature in our paper. Simultaneously, we also consider most previous works are sensitive to noise, and metric learning is widely applied in computer vision due to the ability to generate feature spaces. Therefore, in order to construct a robust metric space, an end-to-end deep metric expression is proposed. Different from DeepContour, we also expect to find a metric space composed of many robust features which can easily judge edge or object. In order to achieve the idea, we also construct a metric space by the deep metric expression algorithm. Three key contents and contributions of the paper are as follow:

Firstly, prior to the symmetrical network architecture, we preprocess the original image to filter some low frequency information and still maintain the image edges. Because the image edge belongs to detail information, too much redundant information in the original image causes the deep network non-converge in the process of training.

Secondly, we construct an end-to-end network which is more efficient to extract robust semantic feature. Simultaneously, all multi-scale features are integrated for complementing information among the features of each level. The multi-scale features can full use low-level features and high-level features and make the boundary of objects clearer.

Finally, a metric space to distinguish edge from object is generated by metric learning. In this space, the between-class distance will be larger than the within-class distance so that we can decrease the metric losses caused via being dropped the low-level information in symmetrical networks. Therefore, we can easily divide each pixel into edge category and object category.

2 Related works

Detecting the edge of the image is one of the most fundamental issues in computer vision, but it is also considered as one of the most difficult study. So far, edge detection is generally classified into two kinds: hand-made feature based approaches and deep feature based approaches. In the next discussion, we will briefly review some representative methods.

The traditional features contain color, texture and gradients, etc. For example, the common one of the gradient methods is Robert [33] Priwitt [30] and Sobel [37]. The Robert [33] operator is very simple and sensitive to noise. It may be easy to appear isolated points. Therefore, J.Priwitt and Sobel is proposed to deal with this problem. Canny is an optimization operator that combines image smoothing, edge enhancement and edge detection. In canny, the Gaussian function is added as a processing operation, and the principle of non maximal suppression is employed to strengthen the margin. Finally, the thresholds are employed to extract the edge. In addition, in [27], multiple local information like color, brightness and texture are combined into a globalization framework. Dollar [8] proposed a semantic edge detection approach which refer to the boosted edge learning which tries to generate edge category and non-edge category via probabilistic boosting trees. The above approaches are on the basis of hand-made features, which easily have the limitation that hand-made features are sensitive to noise.

Recently, Both of HED [41] and COB [26] approaches employ multi-scale features, which have a great performance: fine-tune the classifier. HED algorithm deal well with two main problems, composed of versatile image train and multi-scale features learn. COB is diverse from HED that mixes contour orientation estimation and the pixels classification, and introduces the representation of sparse boundary. In DeepEdge [3] algorithm, the author employs the features that is related to main object as the cues, but not low-level cues, to semantic edge detection. For DeepContour [35], the work of edge detection is regarded as a classification issue, which utilizes the CNNs to create a stable model to obtain two kinds of categories. The achievement based on CNNs have appeared generally and it still has a large place to improve the performance of these CNN-based approaches. Therefore, a prominent edge detection with deep metric expression and multi-scale features is proposed.

The proposed method in this paper obviously differs from the above-mentioned approaches in the following two aspects. Firstly, the proposed approach integrates multi-scale features with multiple levels. Abundant features can be obtained because the high-scale features can denote semantic features and low-level features which contain detail information. Secondly, the idea of metric learning is employed to predict the edge and non-edge regions, that will make the model more robust. In the paper, the metric space, used to metric multi-level features, can enhance the robustness of proposed method. Finally, we employ stochastic gradient descent (SGD) method to update parameters of our model.

3 Proposed network

Inspired by architectures of ResNet [17] and U-Net [34], as shown in Fig. 1, a deep symmetrical metric learning (DSML) network is proposed to deal with the issue of edge detection. We generate an end-to-end network (DSML) to learn robust features, and use matric learning to reduce the training loss that are caused via dropping low-level features in the process of symmetrical network. The DSML architecture is composed of 3 key parts: (1) Guide filter pre-process the origin image. (2) Deep symmetrical network is used to generate robust features. (3) Metric learning space is utilized to construct a outline graph.

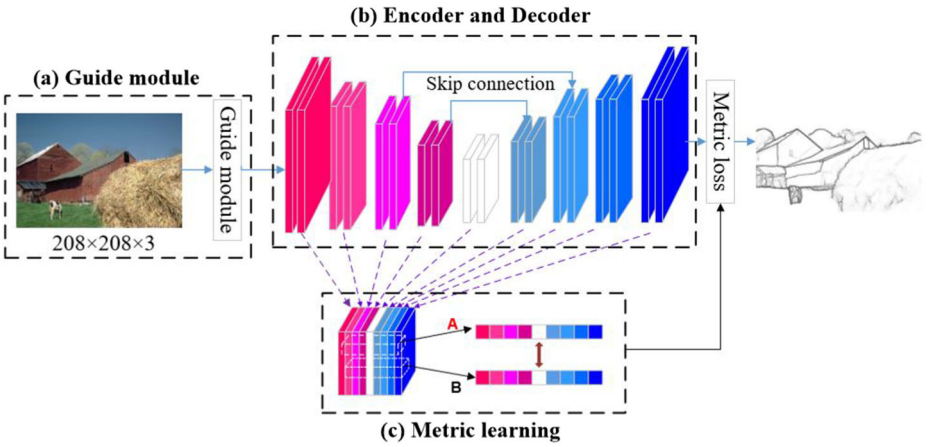


Fig. 1 The framework of our DSML network is employed to detect the edge of image, where 'A' and 'B' respectively denote the feature spaces of edge and object

3.1 Motivation

With the development of the machine learning and pattern recognition, metric learning is increasingly fashionable in many fields, such as image classification [13, 15, 47], information retrieval [22], human identification [20] and face recognition [18, 28]. Furthermore, the goal of metric learning is specially for classifying edge category and non-edge category. Specifically, metric learning is able to construct multi-scale features that have ability to judge each pixel as edge or object. In addition, some previous works have proved that the multi-scale feature can be beneficial to improve the result of edge. Since the low-level features contain more details and the high-level features denote the sematic property of images. Combining the low-level, middle-level and high-level features can make them complement each other. Therefore, due to the advantage of multi-scale features combination and the robustness of metric space, we propose a symmetrical network based on deep metric expression.

3.2 Guide module

Guide filter is the first module of our proposed architecture. From the part (a) of Fig. 1, prior to symmetrical CNNs, we employ guide module to filter the RGB-color image to get a detail graph.

No similar to the way of training previous networks, they use large ImageNet [6] to initialize the parameters of their networks. In the stage of our training, only the BSDS500 dataset and its augmentation dataset are used as our training dataset. Our network is tough to converge on account of our not rich training dataset. Therefore, it's necessary to preprocess our input image for efficiently improving the convergence speed of our network.

Take the above analysis into account, in order to deal with non-convergence problem, the guide filter [16] is introduced. We are able to obtain a lot of high-frequency information, and most information of low-frequency features will be ignored via the filter. Magically, guide filter still remain the existence of edges. The operation has the ability to accelerate the pace of our DSML convergence, which ensure to greatly extract multi-scale features.

3.3 Convolution symmetrical networks

It is universally known that, the low-level feature consists of a lot of available information, and the receptive field becomes larger and larger with the convolution layers increased. Finally, too much information of the low-level features will be suppressed, as high-level features form. If the model only uses the high-level features of the last layer, the edge of natural object will be specially thick in the final outline graph because lots of available details of low-level features will be lost. It urges us to introduce the convolution symmetrical network to attain multi-scale features.

As shown in Fig. 1, based on the U-Net architecture [34] and the ResNet [17] algorithm, the proposed encoder and decoder module is a symmetrical architecture. Our encoder-decoder CNN contains basic module of ResNet that is an international fashionable and popular network at present. Thanks to residual mapping structure of ResNet, it is good for network gradient to feed-forward, which avoids gradient disappear and learns metric space well. In encoder module or decoder module, we repeat a basic operation, respectively. Encoder part includes lot of down-samplings, and at every down-sampling operation, we use convolution with stride 2 to add one time higher than the number of feature channels. Decoder part contains many up-samplings, the deconvolution in full convolution network [25] is used in the operation of up-sampling. Specifically, there are 8 ResNets block both in our Encoder and Decoder symmetrical network. As shown in Fig. 1 c, in order to change the abstract information into quantifiable information, the feature vectors are employed to metric the distance of the corresponding pixels.

As we know, too many low-level features are dropped in symmetrical network, which will make gradient vanish so that the network is tough to converge. To avoid too many details from being dropped, skip connection is employed to deal with the problem. Finally, in the decoder network of Fig. 1, we directly fortify the scale with the final two down-sampling layers. The contribution of the process is that the geometric features learned from the encoder block will be full used in the decoder network so that some improvement will be abtained in the final result. Merely two skip connection is employed in high layer, because there are needn't too much detail information in edge detection.

Through convolution symmetrical CNNs, we hope to create multi-scale feature graph. As the part (b) of Fig. 1 shown, different-scale image is converted to lots of same-scale features via up-samplings. From the part (c) of Fig. 1, metric loss is utilized to metric the distance between different pixels. The details of metric loss are described in next Section 3.4.

3.4 Metric loss function

On the basis of deep learning, many previous networks in the field of edge detection utilized cross-entropy function which measures the losses of the output. The loss function expression of the n times iteration is as follow:

$$Loss^{(n)}(\theta) = -\sum_{i=1}^{m}\sum_{j=0}^{1} Q\{l_i = j\} \log P((l_i = j)|\theta) \tag{1}$$

where the θ is the coefficient of the whole network, the $j \in \{0, 1\}$ denotes binary edge map of the corresponding GT (Ground Truth) image, the $Q(l_i = j) \in [0, 1]$ and $P((l_i = j)|\theta) \in [0, 1]$. Our softmax loss formula has a same expression as formula (1).

The proposed algorithm uses two losses: metric loss function and softmax loss function. In process of training, the metric loss is leveraged to practice the proposed algorithm. And

in the process of testing, the softmax loss is leveraged to get outline map. The first one is the metric loss, which measures the loss from misjudging edge to non-edge or non-edge to edge. In training stage, the input of our network is an image with the shape of $H \times W$, and the output of our network is a feature space(FS) with the shape of $H \times W \times C$. Since every pixel in original image is matched to a C-dimensional vector in the feature space, the metric loss formula of our model is as follow:

$$L_{DSML}^{(n)}(f, f^-, f^+) = -\|f - f^-\|_2^2 + \|f - f^+\|_2^2 + p \tag{2}$$

where f is the matched anchor vector of each pixel in RGB color image, and the f^+ and f^- denote respective the positive vector and negative vector of anchor vector. The final one p is regularization term, and p is equal to 4 in our model. Intuitively, the distance of two vectors from different categories ought to be larger, on the contrary, the distance of two vectors from the same category ought to be smaller.

Another one is softmax loss, which is just employed in testing stage, but not employed in training stage. It makes sure that the final result is a continuous edge graph and keeps the edge not too thick.

4 Simulations

Since Caffe is widely used in the stage of training network, our network is also implemented on it. Gradient is used to deal with image composition [44], but we use its application (stochastic gradient descent) as our loss function. We utilize SGD algorithm with the mini-batch size of 5, the momentum value of 9/10 and the weight decay value of 10^{-8}. We train nearly 1.1×10^5 iterations for our algorithm. With the motivation of making DSML more equitable and universal through the comparison with other deep algorithms, we make experiments at BSDS500 dataset which is widely employed in deep learning. All simulations are trained on a PC with GTX TITAN X Pascal, 64GB RAM and Intel(R)Xeon(R) CPU E5-2670.

4.1 Training and testing

As everyone knows, it's worthless that the network is easy to appear the result of over-fitting if the dataset is too small in training stages. Many traditional works often employ the ImageNet [6] to train VGG16 model to initialize their network parameters so that their network can avoid over-fitting. Without initialization, our training dataset just consists of 200 which is very few in the process of training, so we need expand our dataset in a good strategy. Inspired by the data expansion of HED, our original data is employed to achieve data augmentation via multi-angles of flipping, rotation. Only original dataset and its augmentation dataset are used to train our proposed network. Simulation shows that, we also have reached great result even if we don't utilize any pre-training approach to initialize our networks.

4.1.1 Hard negative mining

Through our observation, a serious issue occurred when we train our network architecture. The problem is that the network cannot converge to a balance point because the number of non-edge samples is more than edge samples' in our training stages. Take this into account, i.e. SSD algorithm [24], we only keep the number of edge samples and non-edge samples at a balanced ratio so that our DSML can converge healthily. We have also experienced this

difficulty in our other paper, we delete some negative samples (non-edge samples) in our training stages. The specific operations of strategy are that the counts of edge pixels are set as Y and the counts of non-edge pixels are set as X, then sorting all the loss via their confidence values in a non-increasing order. In our model, $X > Y$, our model just need take the top Y of negative samples according to the descending order to ensure the rate of edge samples and non-edge samples is 1, then the coordinate of each edge pixel is renewed.

4.2 Evaluation

For the evaluation criteria, we use 4 standard evaluation indexes: fuzzy measure $\left(\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}\right)$ with Optimal-dataset-scale(ODS), Optimal-image-scale(OIS) and Average-precision(AP) and PR curves. OIS denotes fuzzy measure value when each image is in an optimal state. ODS denotes the F-Score with a confirmed outline threshold. AP value is the shorthand of average precision. We can get PR curve with binary masks formed through recall sliding by the step of 0.01, which is most widely utilized. All of these four indexes are better when the value of them are larger.

Our approach is compared with 6 traditional approaches and 5 deep learning approaches. The six traditional approaches are Dollar [8], MCG [2], gPb [1], OEF [14], SCG [2] and SE [9]. The five deep learning approaches are COB [26], HED [41], Deepcontour [35], N^4-Fields [11] and DeepEdge [3].

We evaluate our model by 100 validations at BSDS500 dataset. From the Fig. 2, we have direct visual comparison with other five deep learning networks to finish qualitative assessment. We can see that, our edge maps are more accurate than others, which shows our DSML doesn't have too much redundant details. Our DSML is comparative with these five kinds deep networks. Besides, we also select some images with complex edge to explain

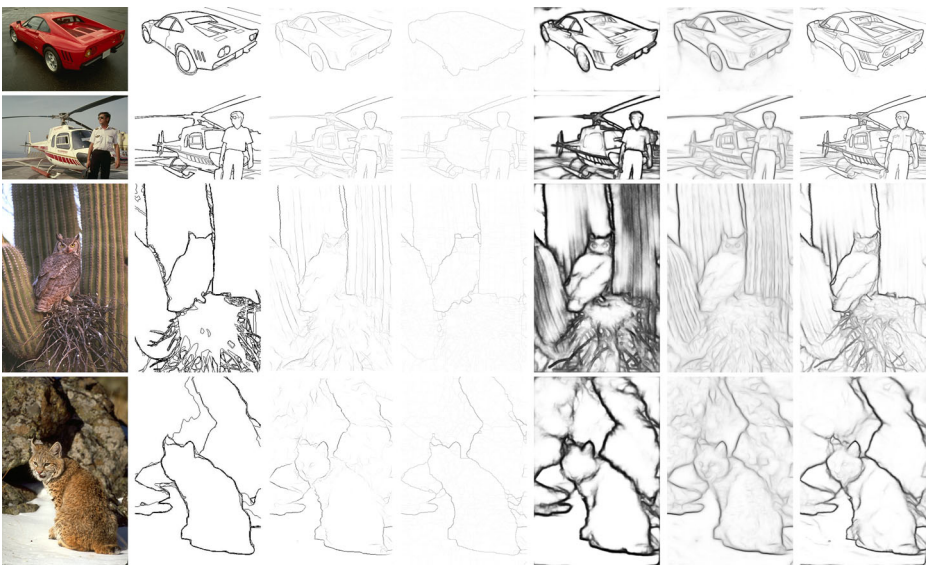


Fig. 2 Comparisons our result with other five popular approaches in qualitatively visual scenes

Table 1 Comparison our method with other 11 previous methods quantitatively, in three evaluation indexes (ODS, OIS and AP)

Method	OIS	ODS	AP
Human	0.80	0.80	–
Dollar [8]	0.719	0.698	0.564
MCG [2]	0.772	0.748	0.789
gPb [1]	0.745	0.719	0.750
OEF [14]	0.760	0.739	0.719
SCG [2]	0.763	0.740	0.774
SE [9]	0.767	0.741	0.790
COB [26]	0.802	0.782	0.825
HED [41]	0.806	0.786	0.840
DeepContour [35]	0.776	0.757	0.795
DeepEdge [3]	0.772	0.753	0.806
$N^4-Fields$ [11]	0.741	0.730	0.753
Ours	**0.804**	**0.788**	**0.845**

The bold fonts are to emphasize the favorable performance of our algorithm compared with others

the capability of our propose method. It's clear that our proposed method has achieve competitive results, as shown in Fig. 4. Though we reach great performance, we still can reach more satisfactory outcome if we also initialize our network parameter by ImageNet.

Furthermore, We also have quantitative comparison with the 11 previous methods, with the result in Table 1 and Fig. 3. Compared with COB method, our DSML is absolutely dominance based on all evaluation indexes. Compared with the most popular method HED

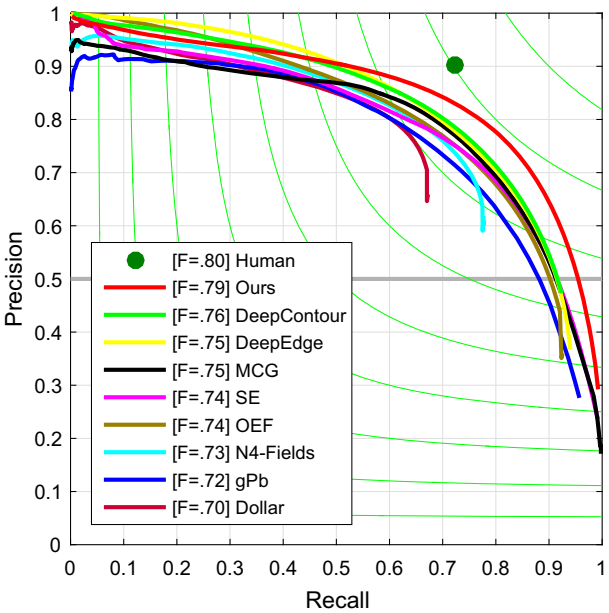
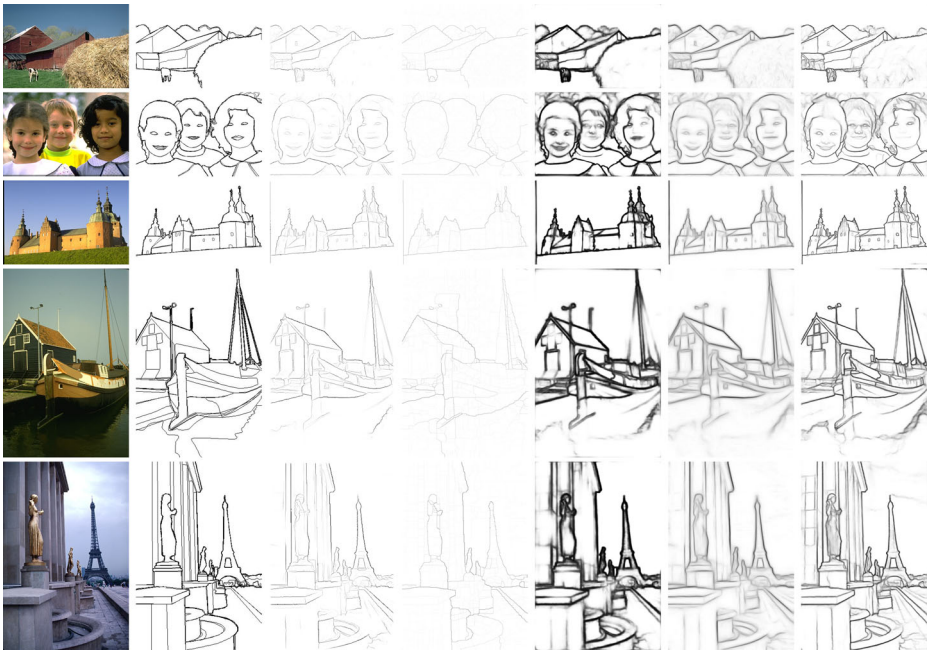


Fig. 3 Comparison of PR curves with other approaches at BSDS500 dataset



(a) Images (b) GT (c) DeepContour (d) COB (e) HED (f) N^4-field (g) Ours

Fig. 4 Comparisons our results with other four models in more complex scenes

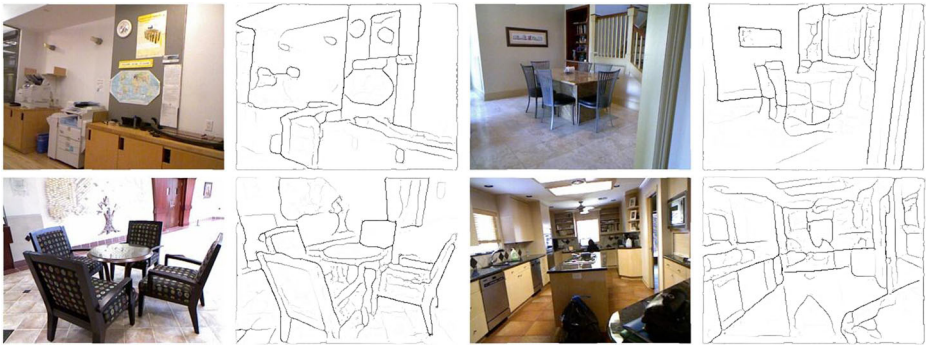
nowadays, though our result is close to HED on OIS F-scores and ODS, our result is a bit higher than on AP. On the one hand, the OIS of the proposed model is 3.2% and 2.8% greater than DeepEdge and DeepContour respectively. On the other hand, the ODS of DSML are 3.5% and 3.1% greater than theirs respectively. The OIS and ODS score of DSML, to their credit, are far greater than other primary algorithms. With the comparison on AP value, it's not doubt that our DSML is nearly higher than all traditional methods. As shown in Fig. 3, it's clear that our PR curve (the red one) is greater than others. Therefore, we can say that our DSML has a dominance hierarchy, as shown by the visual comparisons in Fig. 4.

The advantage of guide filter To intuitively illustrate the advantage of preprocessing, we utilize the different strategies to evaluate the performance of our network (DSML). We show the statistic comparison in Table 2. As shown in Table 2, it is clear that the model outcome of using guide filter as the preprocessing layer is better than the other. Meanwhile, it's proved that Guided Filtering can improve the capability of the model.

Running time Though our model is deeper, it only spends 94 ms for our model to generate each outline with GPU.

Table 2 The advantage of guide filter

Method	Guide filter	ODS	OIS	AP
Ours	No	0.776	0.782	0.801
	Yes	0.788	0.804	0.845



(a) Images (b) Ours (c) Images (d) Ours

Fig. 5 The performances of our model at NYUD dataset

NYUD dataset Furthermore, we select NYUD Dataset as the other test datasets. NYUD [36] dataset includes 1449 samples with pairs of matched depth and RGB images. Most of them contain large edge information, it belongs to challenging dataset for the field of edge detection. It has widely used in edge detection. And we choose 200 images for testing. It's noted that our model does not train on this dataset. During testing, the trained models is then directly tested on distorted images. Fig. 5 shows that our model can obtain well performance.

5 Conclusion

In this work, a deep symmetrical metric learning algorithm is proposed for edge detection. In DSML algorithm, semantic edge detection with deep metric expression is mainly explored where favorable results are obtained. Our network is constructed by deep metric expression consisting of three key blocks: guide filter, convolution symmetrical networks, metric learning space. Since the detail layers are able to quickly reduce computational cost via metric learning, we are able to construct a robust metric space which are capable of telling the difference between the desired objects and the background. Furthermore, experimental results on public datasets show that our model is comparative with many previous models.

Acknowledgments This work was supported in part from the grants of National Science Foundation of China (6151005, 61571382, 61103121, 81671766) and the funding from China Scholarship Council CSC NO. 201806155037, and open funding from Xiamen Key Laboratory of Mobile Multimedia Communications (Huaqiao University), and Guangdong Natural Science Foundation (2015A030313007, 2015A030313589).

References

1. Arbelaez P, Maire M, Fowlkes C, Malik J (2011) Contour detection and hierarchical image segmentation. IEEE Trans Pattern Anal Mach Intell 33(5):898–916

2. Arbeláez P, Pont-Tuset J, Barron JT, Marques F, Malik J (2014) Multiscale combinatorial grouping. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 328–335
3. Bertasius G, Shi J, Deepedge LT (2015) A multi-scale bifurcated deep network for top-down contour detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4380–4389
4. Cai S, Huang J, Ding X, Zeng D (2017) Semantic edge detection based on deep metric learning. In: International symposium on intelligent signal proceeding and communication systems, pp 707–712
5. Cheng M-M, Liu Y, Hou Q, Bian J, Torr P, Hu S-M, Tu Z (2016) Hfs: hierarchical feature selection for efficient image segmentation. In: European conference on computer vision. Springer, pp 867–882
6. Deng J, Dong W, Socher R, Li L-J, Li K, Li F-F (2009) Imagenet: a large-scale hierarchical image database. In: CVPR 2009. IEEE conference on computer vision and pattern recognition, 2009. IEEE, pp 248–255
7. Dollar P, Tu Z, Belongie S (2006) Supervised learning of edges and object boundaries. In: 2006 IEEE computer society conference on computer vision and pattern recognition, vol 2. IEEE, pp 1964–1971
8. Dollár P, Zitnick CL (2013) Structured forests for fast edge detection. In: Proceedings of the IEEE international conference on computer vision, pp 1841–1848
9. Dollár P, Zitnick CL (2015) Fast edge detection using structured forests. IEEE Trans Pattern Anal Mach Intell 37(8):1558–1570
10. Ferrari V, Fevrier L, Jurie F, Schmid C (2008) Groups of adjacent contour segments for object detection. IEEE Trans Pattern Anal Mach Intell 30(1):36–51
11. Ganin Y, Lempitsky V (2014) N 4-fields: neural network nearest neighbor fields for image transforms. In: Asian conference on computer vision. Springer, pp 536–551
12. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
13. Goldberger J, Hinton GE, Roweis ST, Salakhutdinov RR (2005) Neighbourhood components analysis. In: Advances in neural information processing systems, pp 513–520
14. Hallman S, Fowlkes CC (2015) Oriented edge forests for boundary detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1732–1740
15. Hastie T, Tibshirani R (1996) Discriminant adaptive nearest neighbor classification. IEEE Trans Pattern Anal Mach Intell 18(6):607–616
16. He K, Sun J, Tang X (2013) Guided image filtering. IEEE Trans Pattern Anal Mach Intell 35(6):1397–1409
17. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
18. Hu J, Lu J, Tan Y-P (2014) Discriminative deep metric learning for face verification in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1875–1882
19. Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Li F-F (2014) Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1725–1732
20. Koestinger M, Hirzer M, Wohlhart P, Roth PM, Bischof H (2012) Large scale metric learning from equivalence constraints. In: 2012 IEEE Conference on computer vision and pattern recognition (CVPR). IEEE, pp 2288–2295
21. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
22. Li Z, Tang J (2015) Weakly supervised deep metric learning for community-contributed image retrieval. IEEE Trans Multimedia 17(11):1989–1999
23. Lim JJ, Zitnick LC, Dollár P (2013) Sketch tokens: a learned mid-level representation for contour and object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3158–3165
24. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC (2016) Ssd: single shot multibox detector. In: European conference on computer vision. Springer, pp 21–37
25. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440
26. Maninis K-K, Pont-Tuset J, Arbeláez P, Gool LV (2016) Convolutional oriented boundaries. In: European conference on computer vision. Springer, pp 580–596
27. Martin DR, Fowlkes CC, Malik J (2004) Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans Pattern Anal Mach Intell 26(5):530–549
28. Nguyen HV, Bai L (2010) Cosine similarity metric learning for face verification. In: Asian conference on computer vision. Springer, pp 709–720

29. Noh H, Hong S, Han B (2015) Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision, pp 1520–1528
30. Prewitt JMS (1970) Object enhancement and extraction. Picture Processing and Psychopictorics 10(1):15–19
31. Rastegari M, Ordonez V, Redmon J, Farhadi A (2016) Xnor-net: imagenet classification using binary convolutional neural networks. In: European conference on computer vision. Springer, pp 525–542
32. Ren S, He K, Girshick R, Jian S (2015) Faster r-cnn: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99
33. Roberts LG (1963) Machine perception of three-dimensional solids. PhD thesis Massachusetts Institute of Technology
34. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241
35. Shen W, Wang X, Wang Y, Bai X, Zhang Z (2015) Deepcontour: a deep convolutional feature learned by positive-sharing loss for contour detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3982–3991
36. Silberman N, Hoiem D, Kohli P, Fergus R (2012) Indoor segmentation and support inference from rgbd images. In: European conference on computer vision, pp 746–760
37. Sobel I (1970) Camera models and machine perception. Technical report, Stanford Univ Calif Dept of Computer Science
38. Uijlings JRR, De Sande KEAV, Gevers T, Smeulders AWM (2013) Selective search for object recognition. Int J Comput Vis 104(2):154–171
39. Wang L, Lu H, Ruan X, Yang M-H (2015) Deep networks for saliency detection via local estimation and global search. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 3183–3192
40. Wei Y, Liang X, Chen Y, Shen X, Cheng M-M, Feng J, Zhao Y, Yan S (2017) Stc: a simple to complex framework for weakly-supervised semantic segmentation. IEEE Trans Pattern Anal Mach Intell 39(11):2314–2320
41. Xie S, Tu Z (2015) Holistically-nested edge detection. In: Proceedings of the IEEE international conference on computer vision, pp 1395–1403
42. Yang B, Zhang X, Li C, Yang H, Gao Z (2017) Edge guided salient object detection. Neurocomputing 221:60–71
43. Yang Z, Xiang Y, Xie K, Lai Y (2017) Adaptive method for nonsmooth nonnegative matrix factorization. IEEE Transactions on Neural Networks and Learning Systems 28(4):948–960
44. Zhang W, Cham W-K (2012) Gradient-directed multiexposure composition. IEEE Trans Image Process 21(4):2318–2323
45. Zhang W, Ma B, Liu K, Huang R (2017) Video-based pedestrian re-identification by adaptive spatio-temporal appearance model. IEEE Trans Image Process 26(4):2042–2054
46. Zhang W, Yu X, He X (2017) Learning bidirectional temporal cues for video-based person re-identification. IEEE Trans Circuits Syst Video Technol. https://doi.org/10.1109/TCSVT.2017.2718188
47. Zhang Z, Kwok JT, Yeung D-Y (2003) Parametric distance metric learning with label information. In: IJCAI, p 1450
48. Zhao R, Ouyang W, Li H, Wang X (2015) Saliency detection by multi-context deep learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1265–1274
49. Zitnick CL, Dollár P (2014) Edge boxes: locating object proposals from edges. In: European conference on computer vision. Springer, pp 391–405
50. Zou W, Komodakis N (2015) HARF: hierarchy-associated rich features for salient object detection. In: Proceedings of the IEEE international conference on computer vision. IEEE, pp 406–414
51. Zou W, Liu Z, Kpalma K, Ronsin J, Zhao Yong, Komodakis N (2015) Unsupervised joint salient region detection and object segmentation. IEEE Trans Image Process 24(11):3858–3873

Shulian Cai Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. Shulian Cai received the bachelor's degree in communication engineering from the Fuzhou University of Economics in 2015. She is currently pursuing the master?s degree in communications and information systems with Xiamen University, Xiamen, China. Her research interests include image processing, deep learning, and machine learning.

Jiabin Huang Fujian Key Laboratory of Sensing and Computing for Smart City, School of Information Science and Engineering, Xiamen University, Xiamen, China. Jiabin Huang received the bachelor's degree in electronic information engineering from the Hubei University of Economics in 2015. He is currently pursuing the master's degree in electronic and communication engineering with Xiamen University, Xiamen, China. His research interests include image processing, deep learning, and machine learning.

Jing Chen received the B.S. and M.S. degrees from Huaqiao University, Xiamen, China, and the Ph.D. degree from Xiamen University, Xiamen, China, all in computer science. She is now an Associate Professor at the School of Information Science and Engineering, Huaqiao University, Xiamen, China. Her current research interests include image processing and video coding.

Yue Huang received the B.S. from Department of Electrical Engineering, Xiamen University, and Ph.D. degrees from Department of Biomedical Engineering, Tsinghua University, Beijing, China, in 2005 and 2010, respectively. Since 2010, she has been an associate professor with the Xiamen University, Xiamen, China. Her main research interests include sparse signal representation, and machine learning.

Xinghao Ding received the B.S. and Ph.D. degrees from the Department of Precision Instruments, Hefei University of Technology, Hefei, in 1998 and 2003, respectively. He was a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA, from 2009 to 2011. Since 2011, he has been a Professor with the School of Information Science and Engineering, Xiamen University, Xiamen, China. His main research interests include machine learning, sparse signal representation, and image processing.

Delu Zeng received the B.S. and M.S. degrees in applied mathematics and the Ph.D. degree in electronic and information engineering from South China University of Technology, Guangzhou, China, in 2003, 2005, and 2009, respectively. He is now an associate professor in the school of Mathematics in South China University of Technology, China. His research interests include partial differential equations, machine learning, and their applications in image and video processing, i.e., image segmentation.