Multi-Scale Feature Decode and Fuse Model with CRF Layer for Boundary Detection

Zihao Dong[1], Ruixun Zhang[2], Xiuli Shao[1], Huichao Li[1], and Zihan Yang[1]

[1]College of Computer and Control Engineering, NanKai University Tian Jin, China
[2]MIT Laboratory for Financial Engineering, Cambridge University MA, USA
1120170132@mail.nankai.edu.cn, zhangruixun@gmail.com, shaoxl@nankai.edu.cn, 2120160395@mail.nankai.edu.cn, pink_edward@126.com

Abstract. The key challenge for edge detection is that salient edge is difficult to detect due to the complex background. To improve the resolution and accuracy of salient edge effectively, we propose a novel method of edge detection called MSDF (Multi Scale Decode and Fusion) based on deep structured multi-scale features in this paper. The decoding layer of MSDF can fuse the adjacent features of the DNN multi-scale and increase the correlation between the features. In the fusion of different scale's information, the traditional method of up-sample based on deconvolution is not used and Subpixel [13] algorithm is adopted to improve the resolution of the convolution layer's output image. We also build a new Conditional Random Fields(CRF) model with CRF-RNN layer to reduce the number of irrelevant features and eliminate the weak correlation information while retaining the important structural attributes. Extensive experiments on BSDS500 [1] dataset and the larger NYUD [19] dataset show that the effectiveness of the proposed model and of the overall hierarchical framework.

Keywords: edge detection, HED [6], multi-scale feature, Subpixel, CRF

1 Introduction

Edge detection is to extract the salient edge from the input image with complex background accurately. It can be applied to image segmentation, visual saliency, object detection and image retrieval. Convolutional Neural Networks (CNNs) have become a recent trend to improve the state of varieties of methods and training models.

CNN is no longer dependent on the manual design of researchers for learning image features unlike traditional methods. Though training large quantities of image data, it tends to get general features such as intensity, depth, and texture. These features are usually used to get a classifier by supervised learning of edge detection, and the classifier will predict the edge and non-edge pixels. But edge detection is still a very challenging problem and remains unsolved due to the facts that: (i)Edge features are very sparse, we hardly distinguish the relevant features and irrelevant features of the edges. (ii) Some methods based on leaning

multi-scale features have a rough treatment of different scale features, which result in the lower correlation between scales, and a simple average of these edge maps of multi-scale will output low-quality edges.

This paper mainly has two parts. The first one is to propose a new edge detection method called MSDF (Multi Scale Decode and Fusion) based on learning multi-scale features. MSDF can extract the edge features of 5 scales in VGG-16 network [18] using Holistic Edge Detector(HED) [6], and combine the bottom-up decoding architecture with subpixel method [13], which is designed for learning clear edge features. Finally, the multi-scale edge extracted by HED is fused with the decoded edge to obtain the detection result of MSDF. The second one is to build a end-to-end training CRF model based on MSDF edge detector for weakening unrelated features and reducing the error rate.

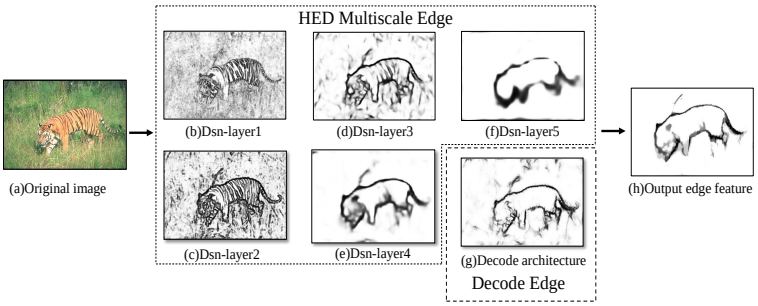

Fig. 1. Multi-scale extraction and fusion of MSDF algorithm.

In Fig. 1, we show a visualization of MSDF multi-scale edges. One can clearly see that the resolution of decoded edge features is higher than HED multiscale including low-level and high-level, but it contains some fine details that do not appear in other layers. MSDF model will output edge map that closed to Ground Truth through fusing feature maps from each stage.

The remainder of this paper is structured as follows. In Section 2, the related work is introduced in edge detection. Section 3 describes the overall framework of the proposed method. Section 4 introduces the CRF model structure based on CRF-RNN layer. Section 5 presents the experiment and application in BSDS500 and NYUD datasets. Section 6 concludes this paper.

2 Related Work

This paper focuses on edge detection in images with complex background. In edge detection algorithms proposed in the literature, we divide them into four groups, including the local edge feature detectors, edge classification methods using machine learning, and CNN-based training methods.

Typically, some traditional methods focus on extracting local cues of brightness, colors, gradients and textures, or other manually designed features. As for the local edge feature detectors, Arbelaez et al. [1] combine multiscale local brightness, color and texture cues into a globalization framework using spectral clustering to detect salient edge. Lim et al. [2] defines the method of sketch token to represent the structure estimation of local edges, then uses random forest to complete the classification of different image patch in the sketch token. Among machine learning-based methods, Dollar et al. [3] present structured edge detector SE to construct a structured decision tree for edge detection, where PCA is used to achieve data dimensionality reduction and the random forest is used to capture the structured information.

In recent years, CNN-based method began to be applied and achieve impressive performance in the edge detection. Shen et al. [5] use k-means clustering method to classify image patches and multi-class shape patches is extracted through a 6-layer CNN network, then structured random forest is used to further classify the edge to obtain more discriminative features. Xie and Tu [6] propose a deep learning model combining full convolutional network (FCN) [8] and deeply supervised nets [7] to detect edges. Based on the deep learning model of [6], some methods focus on improving the network structure to generate better features for performing the pixel-wise prediction, such as RCF [9] and CED [12], other methods add some useful components to achieve better accuracy, such as Deep Boundary [10] and COB [11], these components include multi-scale, extern training data with PASCAL Context dataset [21] and Normalized Cuts [1]. In addition to the above-mentioned researches, some works have begun to consider a new type of network that can go deeper without the problem of vanishing/exploding gradients and use feature parameters effectively. Taking Dense Net as an example, [17] has successfully applied it to the area of object segmentation and edge detection.

To explore the object level contours, [14–16] focus on how to add other mechanisms in VGG-16 network to weaken the influence of unrelated feature information to obtain edge map. Liao et al. [15] proposed an edge correlation graph (CCG) to predict the validity of candidate edges and convert the CCG segmentation information into feature mapping function to obtain an effective CRF model. Yang et al. [16] develop an object-centric edge detection method using efficient fully convolutional encoder-decoder network, this method uses a refinement method based on dense CRF to concern with the imperfect edge annotations from polygons. Xu [14] proposed the AG-CRFS model, which mainly adopts two methods of CRF and attention mechanism to produce more rich and complementary representations.

3 MSDF Model

3.1 Network Architecture

Overview The overall network architecture of the proposed MSDF model is illustrated in the left part of Fig. 2. The MSDF model leverages a bottom-up

decoded pathway to complete the fusion of multi-layer features. It consists of two sub-networks: (i)a HED subnet that extracts top-down multi-scale features, (ii)a decoder subnet to decode bottom-up feature information. Each branch produces different edge maps at different layers of this network, such as HED dsn-layer (1-5) and decoded convolutional layer. In decoded subnet, each module fuses a bottom-up feature map from its bottom layer with a top-down feature map from the convolutional layer in VGG-16 FCN. The two subnets are fused finally, we average them to generate the final edge map.

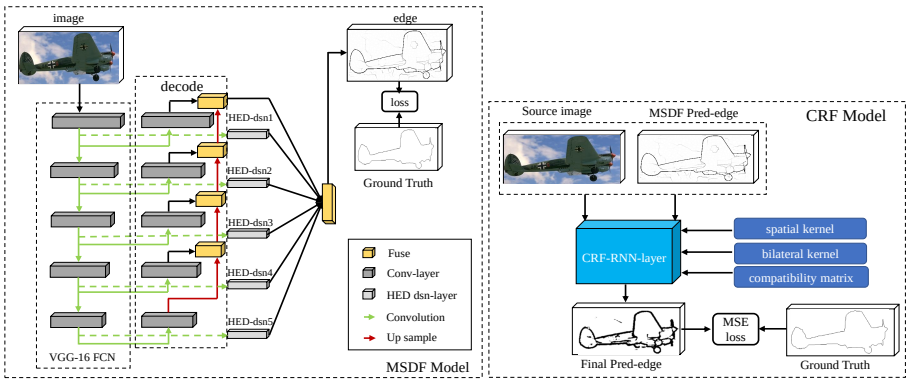

Fig. 2. Architecture of the proposed multi-scale deep learning MSDF model (left) and CRF model with CRF-RNN layer (right).

Top-down scale feature extraction The top-down architecture is based on that of VGG-16, we cut all the fully connect layers and pool5 layer. The multi-scale features can be learned, including low-level and object-level. In Fig. 2, some parts of this model are omitted: a conv-layer with kernel size (1×1) and channel depth 1 follows each conv-layers(convolutional layers) of VGG-16 network, then a deconvolutional layer is used to up-sample this scale feature map, finally edge map of input image will be obtained from HED dsn-layer.

Bottom-up decoder subnet The modules of decoder are associated with each conv-layer in VGG-16 Net. Each conv-layer of this subnet up-samples the map by a factor $(2\times)$ and fuses with the left layers of VGG-16 FCN. For bottom to up pathway, the number convolutional layer is set to 1024 for the bottom layer, this value is reduced by 2 times from bottom layer to top layer. For example, the first Conv-layer on the top of the bottom layer will have 256 feature channels and the second, third bottom-up module will have 128 and 64 feature channels respectively. In this sub-net, up-sampling and fusion is important for better resolution and correlation of edges.

Up-sample of decoder sub-net In order to improve the image resolution after up-sampling, the subpixel method is used to perform standard convolution processing on the low-scale features, and then rearrange feature values by the

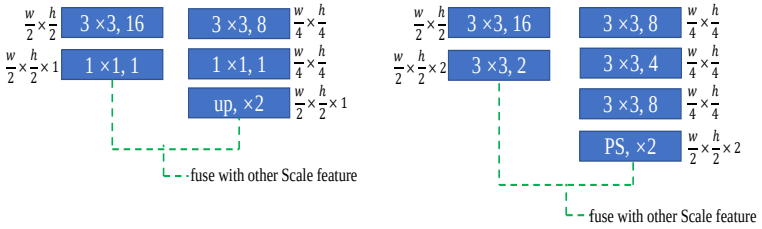

Fig. 3. The assumption of two different fuse methods of HED and decoder sub-net.

phase shift (PS) [13] operation. Assume that the scale of a convolution feature map is $h \times w \times c$, where h, w and c represent the height, width,and channels of the feature map, respectively. According the output feature map being r times larger than it, the size of feature map using the traditional deconvolution method is $(r \times h) \times (r \times w) \times c$. The subpixel method firstly generates a feature map of $r^2 \times c$ feature channels with the same resolution through a standard convolutional operation, and then uses the PS operation to combine the output feature maps into a feature map with c feature channels, but the resolution will increase by r times.

Fusion method of decoder subnet HED uses a simple fusion strategy that directly completes the stitching of two feature matrices, but the premise of this strategy is that the number of channels of the input feature map must be the same. In the decoder subnet of MSDF model, the number of feature channels of different scales is not the same, splicing two feature matrices directly will lead to lost some low-dimensional features. Therefore, this paper reduces the dimensions of high-scale features by increasing the number of convolutional layers and has the same number of channels as low-scale features. In this way, the matrix splicing method of HED can be used to fuse the feature map between different scales. Please see Fig. 3 for the fusion method details. We assume that the features of $(3 \times 3, 16)$ convolutional layer and $(3 \times 3, 8)$ convolutional layer need to be fused, 3 is kernel size, (8 and 16) are number of channels, $(\frac{w}{2} \times \frac{h}{2})$ is the size of feature map. The left is the strategy of HED, it only concatenate two different feature map by additional $(1 \times 1, 1)$ convolutional layer and upsample layer. The right is our fusion method, we add some refinement modules such as convolutional layers and PS operation above to realize the fuse between two different feature maps. From this case, our method can retain more information of low-dimensional features.

3.2 Learning and Deployment

The image edge detection problem can be expressed as a general mathematical form. This paper uses X_n to represent the image in data layer, which is generally transformed into a multidimensional matrix, Y_n denotes the set of edge feature pixels of the Ground Truth $Y_n = \{y_j^{(n)}, j = 1, ..., |X_n|\}$ Where $|X_n|$ represents the number of pixels in the image n. $P(y_j = 1|X_n)$ is defined to represent the

possibility of predicted pixel labeled by annotator. Here, 0 means the pixel of edge predicted by the model is not the Ground Truth label, and 1 means the pixel of edge predicted by the model is labeled by all annotators. Because BSDS500 has five different annotations for each image, which leads to controversial issues, a threshold μ is defined here, we set it to 0.7 depending on training data in loss function. When $P(y_j = 1|X_n)$ is greater than 0 and less than μ, the pixel is controversial and should not be considered. The image training data will be represented as $S = \{(X_n, Y_n), n = 1, 2, ..., N\}$, where N is the number of training images.

We compute the loss functions of the top-down architecture and the decoder subnet at every pixel with respect to the Ground Truth pixel label as

When $P(y_j = 1|X_n) \geq \mu$,

$$l_{side}(W, X_n) = -\beta \sum_{j \in y^+} \log P(y_j = 1|X_n, W) - (1 - \beta) \sum_{j \in y^-} \log P(y_j = 0|X_n, W) \tag{1}$$

When $0 < P(y_j = 1|X_n) < \mu$, $l_{side}(W, X_n) = 0$.

In which

$$\beta = \frac{|Y_-|}{|Y_-| + |Y_+|}; 1 - \beta = \frac{|Y_+|}{|Y_-| + |Y_+|} \tag{2}$$

Y_+ is an edge label and Y_- is a non-edge label.

The total loss function of the multi-scale edge detection model also needs to include the fusion loss. This paper assumes that the top-down architecture has the same loss function as the decoder subnet, the value of W will be learned and updated during the training process. Therefore, our improved loss function can be formulated as:

$$l(W, X_n) = \sum_{i=1}^{N} (\sum_{j=1}^{M} l_{side}^j(W, X_n) + l_{fuse}(W, X_n)) \tag{3}$$

where N is the number of training images, and M is the total number of HED dsn-layers and the layers of decoder subnet (here is set to 6). $l_{fuse}(W, X_n)$ represents the loss function of the fusion layer.

4 CRF Model based on MSDF

4.1 CRF Model Architecture

This paper applies a new CRF model to connect with the network structure of MSDF model. It can be trained end-to-end utilizing the usual back-propagation algorithm and mainly used to post-process the edge maps predicted by MSDF model. The input of CRF model is composed of the original image and the final edge maps predicted by MSDF method. It estimates the CRF parameters on each edge and is adopted to fulfill the optimization of estimation. In the right part of Fig. 2, our CRF model is made up of CRF-RNN Layer with the weights

of spatial kernel and bilateral kernel [22] which depend on the number of classes. Finally, the weights of this network are optimized to solve a regression task by Mean Squared Error (MSE) function, where the objective to reduce the error between predicted edge map and ground truth. Due to the CRF model can make full use of the features of adjacent scales and predict the validity of local edges, our method will eliminate most blurry and noisy boundaries significantly.

CRF-RNN layer [22] combines the strengths of CNN and CRF-based probabilistic graphical modeling to imply end to end training in semantic segmentation. In our model, CRF-RNN method is adopted to solve the optimization problem of edge detection. We use the function $f(U, Q_{in}, X_n, \theta)$ to denote the transformation done by one mean-field iteration: the image X_n, the unary potential function U, the marginal distribution Q_{in} and the CRF parameters θ. The unary potential function U is the multi-scale fuse layer's output from VGG-16 network. Each iteration takes Q_{in} value from the previous iteration to achieve multiple mean-field iterations. The CRF parameters θ includes class numbers, iteration numbers(i), and other hyperparameters. So, the behavior of CRF-RNN layer is given by the following equations:

$$H_1(t) = \begin{cases} sigmoid(U), & i = 0. \\ H_2(t-1), & 0 < t < i. \end{cases} \tag{4}$$

where $H_1(t)$ and $H_2(t-1)$ are hidden states. We use $sigmoid(U)$ instead of $softmax(U)$ to solve binary classification problem.

$$H_2(t) = f(U, H_1(t), \theta), \quad 0 < t < i. \tag{5}$$

Output is get by Eq. (5) operation when t achieve to mean-field iterations T (set to 5):

$$Y(t) = H_2(t), \quad t = T \tag{6}$$

In order to be compatible with predicted result of MSDF, we build the CRF network to solve a regression problem and employ the MSE function to reduce the training error. Using the predicted results of Eq. 6, we can obtain the loss value computed by MSE, which is shown as follows:

$$l_{CRF} = \sum_{i=1}^{N} ||\hat{Y_i} - Y(t)_i||^2, \quad t = T, N = batchsize \tag{7}$$

with Y_i the ith the ground truth, $Y(t)_i$ the ith predicted edge map of Eq. (6), N the batch size.

In this way, two independent end-to-end models (MSDF and CRF model) are trained for edge detection.

Training with external data The number of training image in BSDS500 dataset is only 300, so the author in [6] used 16 rotations and flipping of 200 images used in train set and 100 images used in val set. But this method will result in generating some different scale images. We have used a new method called random-cropping to achieve dataset augmentation. Our modification to

this boundary have been to random generate X and Y, which will help to crop 256×256 image in this coordinate. We also flip the image at 16 different angles, leading to an augmented training set with 30000 training images.

5 Experiments

Dataset We perform model training and performance evaluation on the BSDS500 [1] and NYUD [19] datasets. BSDS500 is a dataset widely used in image segmentation and image edge detection. It is composed of 200 training images, 100 verification images, and 200 test images. Each image is manually identified and there is a corresponding Ground Truth label. We rotated and scale 200 training pictures and 100 verification pictures to expand the size of the training set , and use 200 test pictures as performance evaluation. The NYUD dataset consists of 1449 images containing three parts: RGB, Depth, and acceleration data. In order to facilitate the model training, we divide the NYUD dataset into two parts of RGB and HHA. We found that the training of MSDF model on HHA part is unstable making it difficult to draw any conclusion, only RGB part is selected. There are 381 images in the RGB training set, 414 in the validation set, and 654 images in the test set.

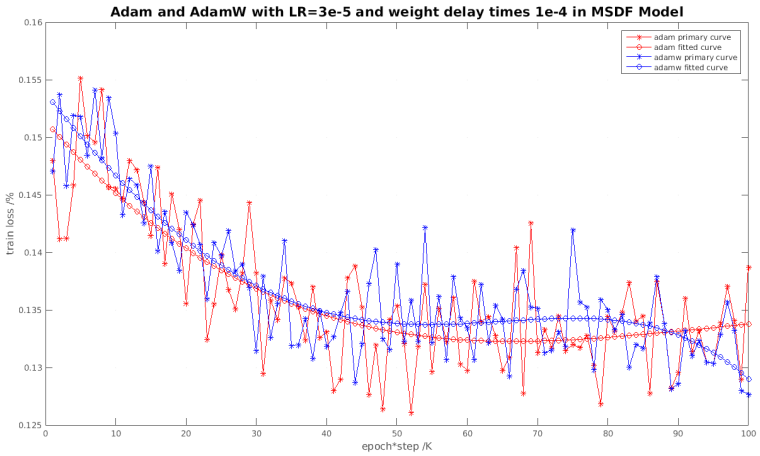

Fig. 4. Loss curves obtained by MSDF+CRF model trained by Adam and AdamW on BSDS500 dataset.

Implementation Details Our MSDF and CRF model build on the publicly available code of HED [6], using TensorFlow as backend with a single NVIDIA GTX1070 GPU. In this MSDF model, we initialize the conv-layers of encoder subnet with Gaussian random distribution with fixed mean (0.0) and variance (0.01) and the conv-layers of top-down subnet with constant weight (0.2) and variance (0). The hyper parameters, including the initial learning rate, weight

decay and momentum, are set to $3e-5$, $1e-6$ and 0.7, respectively. We use AdamW optimizer (see details in next section) as the method of gradient optimization with the parameter $\varepsilon = 1e-3$. The learning rate is set as $3e-5$ at first 30000 iterations and further decreased to $1e-6$ for another 10000 iterations with an epoch size of 100. In BSDS500 training, we randomly cropped an original 321×481 image of size 20×30 and rotated it 90° for data augmentation during training. In NYUD training, we randomly cropped an original 425×560 image of size 26×35, rotated it 90°, 180°, 270°, and flipped it vertically and horizontally for data augmentation during training.

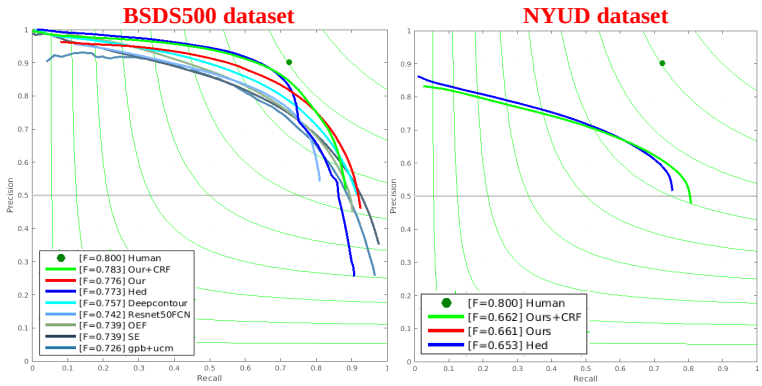

Fig. 5. The evaluation results on BSDS500 dataset and NYUD dataset. The left one is the P-R curve in BSDS500 and the right one is the P-R curve in NYUD-RGB.

ADAMW Optimization In order to recover the original formulation of weight decay regularization by decoupling the weight decay, we use a new adaptive gradient method called AdamW [23] as the method of gradient optimization instead of Adam. This method improves Adam's performance in three practices: decoupling weight decay, formal analysis of weight decay vs $L2$ regulation and cosine annealing and warm restarting.

According to the experiment of [23], the hyper parameters including the initial learning rate and weight decay, are set to $3e-5$ and $1e-4$, respectively. The learning rate is set as $3e-5$ at first 30000 iterations and further decreased to $1e-6$ for another 10000 iterations with an epoch size of 100.

Fig. 4 shows the results for same settings of weight decay of Adam and AdamW. Importantly, the use of weight decay $1e-4$ in Adam did not yield as good results as in AdamW, the train loss of AdamW fitted curve is lower than the Adam Curve in 100 epoch and has the faster convergence.

Evaluation Metrics The evaluation metrics used here are P/R (precision/recall) curves, ODS (optimal dataset scale), OIS (optimal image scale), AP (average precision), and F-measure parameters for predicting edge matching degree. ODS divides the fixed parameters of all images. OIS selects the optimal

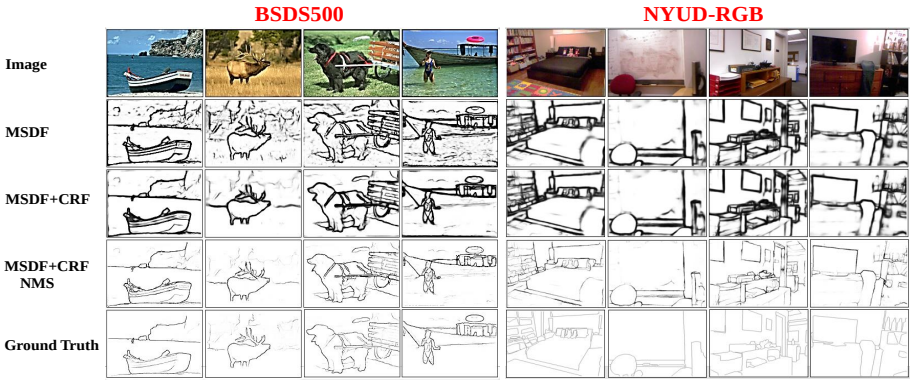

Fig. 6. Some examples of MSDF and MSDF+CRF. From left to right: BSDS500 and NYUD. From top to bottom: origin image, MSDF edge map, MSDF+CRF edge map, MSDF+CRF edge map with NMS, and Ground Truth.

segmentation parameters for each image, and F-measure is calculated by the precision and recall:

$$F - measure = \frac{2 \times precision \times recall}{precision + recall} \tag{8}$$

Table 1. Comparative results against state-of-the-art edge detection method performance.

DataSet	Methods	ODS	OIS	AP
BSD500	gpb+spb+ucm [1]	0.726	0.760	0.727
	SE [3]	0.739	0.759	0.792
	OEF [4]	0.739	0.761	0.720
	Resnet50FCN	0.742	0.755	0.585
	DeepContour [5]	0.757	0.776	0.790
	HED [6]	0.770	0.789	0.645
	COB [11]	0.782	0.801	**0.824**
	MSDF	0.780	0.793	0.798
	MSDF+CRF	**0.783**	**0.802**	0.801
NYUD	HED [6]	0.653	0.665	0.559
	MSDF	**0.661**	**0.674**	**0.565**
	MSDF+CRF	**0.662**	**0.675**	**0.567**

Comparisons against other methods Our model is compared to other better methods in Table 1. The expansion method of dataset (details as the section of implementation details) is different from other algorithms such as HED,

so the evaluation results based on our model structure and the reports of other methods in original papers exist obvious variances. The following observations can be made: (i) Our method (MSDF+CRF) outperforms all compared models. For BSDS500 data, ODS F-measure of MSDF is 1% higher than HED, which indicates the decode part of our method plays a main role for improving the accuracy of boundary detection. But AP of COB is the best performance, the reason may be that the decoding part of MSDF model adds some noise to the detection process, which is the direction we need to further study. For NYUD data, when compared with HED, ODS F-measure of MSDF is higher 0.8% than it, the performance of MSDF+CRF is almost same as the single MSDF model. (ii) In Fig. 5, the precision-recall curves of our methods are also higher than HED's. (iii) The poor result of Resnet50 [20] suggests that the performance of DNN is not better as deep as possible, that's why we choose VGG-16 net as the based network. (iv) ODS F-measure of Our model in NYUD is lower than BSDS500's, because images in NYUD dataset are larger than images in BSDS500 dataset.

Qualitative Results Example edge detection results of the proposed model are shown in Fig. 6. The results suggest that the model is more robust over different kinds of images. It can be seen that MSDF model can improve global resolution of the image edges, and the CRF model has a better effect on eliminating irrelevant features, our edge detection method is fairly close to the annotated edge of Ground Truth, especially the edges detected in the second column, Our method almost eliminates the irrelevant effects of complex backgrounds.

6 Conclusion

This paper proposes a new CNN structure MSDF based on learning multi-scale features. The architecture mainly includes the top-down scale feature extraction and the new proposed decoder subnet, the decoder subnet shows a new method for feature fusion in different scales and how to balance the fusion and up-sampling to carry out edge detection. Finally, the paper uses a new CRF model based on MSDF to link relevant information together, this makes it promising to be used for image segmentation, object detection and another high-level task. The implementation code of MSDF model will be available at https://github.com/zihaodong/Multi-Scale-Decode-Feature-.

References

1. P, Arbelez., M, Maire., C, Fowlkes., J, Malik.: Contour detection and hierarchical image segmentation. In: TPAMI 2011, vol. 33, pp. 898 - 916. IEEE Press, New York(2011).
2. J, J, Lim., C, L, Zitnick., P, Dollr.: Sketch tokens: A learned mid-level representation for contour and object detection. In: CVPR 2013, pp. 3158-3165. IEEE Press, New York(2013).
3. P, Dollr., C, L, Zitnick.: Fast edge detection using structured forests. In: TPAMI 2015, vol. 37, pp. 1558 - 1570. IEEE Press, New York(2015).

4. S, Hallman., C, C, Fowlkes.: Oriented edge forests for boundary detection. In: CVPR 2015, pp. 1732-1740. IEEE Press, New York(2015).
5. W, Shen., X, Wang., Y, Wang., X, Bai., Z, Zhang.: Deep Contour: A deep convolutional feature learned by positive sharing loss for contour detection. In: CVPR 2015, pp. 3982-3991. IEEE Press, New York(2015).
6. S, Xie., Z, Tu.: Holistically-nested edge detection. In: ICCV 2015, pp. 1395-1403. IEEE Press, New York(2015).
7. C,-Y, Lee., S, Xie., P, Gallagher., Z, Zhang., Z, Tu.: Deeply supervised nets. In: PMLR, pp. 562-570. 2015.
8. J, Long., E, Shelhamer., T, Darrell.: Fully convolutional networks for semantic segmentation. In: CVPR 2015, pp. 3431-3440. IEEE Press, New York(2015).
9. Y, Liu., M,-M, Cheng., X, Hu., K, Wang., X, Bai.: Richer convolutional features for edge detection. In: CVPR 2017, pp. 3000-3009. IEEE Press, New York(2017).
10. I, Kokkinos.: Pushing the boundaries of boundary detection using deep learning. arXiv:1511.07386, 2015.
11. K, K, Maninis., J, Pont-Tuset., P, Arbelaez., L, VanGool.: Convolutional oriented boundaries: From image segmentation to high-level tasks. In: ECCV 2016, pp. 580-596. Springer, Cham(2016).
12. Y, Wang., X, Zhao., K, Huang.: Deep Crisp Boundaries. In: CVPR 2017, pp. 1724-1732. IEEE Press, New York(2017).
13. W, Shi., J, Caballero., F, Huszr., J, Totz., A, P, Aitken., R, Bishop., D, Rueckert., Z, Wang.: Real-time single image and video super-resolution using an efficient subpixel convolutional neural network. In: IEEE CVPR 2016, pp. 1874-1883. IEEE Press, New York(2016).
14. D, Xu., W, Ouyang., X, Alameda-Pineda., E, Ricci., X, Wang., N, Sebe.: Learning deep structured multi-scale features using attention-gated crfs for contour prediction. In: NIPS 2017. IEEE Press, New York(2016).
15. Y, Liao., S, Fu., X, Lu., C, Zhang.: Deep learning based contour detection with CCG and CRF. In: ICME 2017, pp. 859-864. IEEE Press, New York(2017).
16. J, Yang., B, Price., S, Cohen., H, Lee., M,-H, Yang.: Object contour detection with a fully convolutional encoder-decoder network. In: CVPR 2016, pp. 193-202. IEEE Press, New York(2016).
17. Q, Hou., J, Liu., M,-M, Cheng., A, Borji., PHS, Torr.: Three Birds One Stone: A Unified Framework for Salient Object Segmentation, Edge Detection and Skeleton Extraction. arXiv preprint arXiv:1803.09860,2018.
18. K, Simonyan., A, Zisserman.: Very deep convolutional networks for large-scale image recognition. In: ICLR 2015. IEEE Press, New York(2016).
19. N, Silberman., D, Hoiem., P, Kohli., R, Fergus.: Indoor segmentation and support inference from rgbd images. In: European Conference on Computer Vision 2012, pp. 746-760. Springer, Cham(2012).
20. K, He., X, Zhang., S, Ren., J, Sun.: Deep residual learning for image recognition. arXiv:1512.03385, 2015.
21. R, Mottaghi., X, Chen., X, Liu., N,-G, Cho., S,-W, Lee., S, Fidler., R, Urtasun., A, Yuille.: The role of context for object detection and semantic segmentation in the wild. In CVPR 2014, pp. 891-898. IEEE Press, New York(2014).
22. S, Zheng., S, Jayasumana., B, Romera-Paredes., V, Vineet., Z, Su., D, Du., C, Huang., P, Torr.: Conditional random fields as recurrent neural networks. In ICCV 2015, pp. 1529-1537. IEEE Press, New York(2015).
23. I, Loshchilov., F, Hutter.: Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101, 2017.