ITERATIVE RESIDUAL NETWORK FOR STRUCTURED EDGE DETECTION

Yupei Wang[1,2], *Xin Zhao*[1,2], *and Kaiqi Huang*[1,2,3]

[1]CRIPAC, NLPR, Institute of Automation, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences
[3]CAS Center for Excellence in Brain Science and Intelligence Technology

ABSTRACT

Edge detection aims to find visually distinctive edges or boundaries in input images. Edge detection has made significant progress with the help of deep Convolutional Networks (ConvNet). Most ConvNet-based edge detectors predict each pixel independently and ignore the inherent correlations between pixels. However, structured cues in input images are critical to learn a good edge detector. To this end, we propose a novel Iterative Residual Holistically-nested Edge Detection (IRHED) network. IRHED incorporates multi-scale features from the hierarchy of the network, and learns to iteratively refine the output boundary map in a deeply supervised manner. In this way, global structural cues, such as object shape, are learned implicitly, thus edges can be effectively distinguished. Extensive experiments demonstrate that IRHED achieves state-of-the-art results on the widely used BSDS500 dataset. We also show the benefit of structured edge map for higher-level task, such as object proposal generation.

Index Terms— Iterative residual, structured edge detection

1. INTRODUCTION

Edge detection seeks to identify all image edges and has made significant progress using deep models. [1] proposed a holistically-nested edge detection network, the first end-to-end method that approaches human performance on edge detection. [2] further extended the idea of HED by combining multilevel features in ConvNet.

Most ConvNet-based edge detection models predict each of the pixels independently, focusing on the corresponding receptive field in the input image. These methods achieve good performance owing to the strong ability to learn robust representation. However, due to the limitation of this independent prediction scheme, current ConvNet-based edge detectors fail to capture structured cues encoded in input images.

In this paper, we propose a novel Iterative Residual Holistically-nested Edge Detection (IRHED) network. Figure 1 presents an overview of our method. Specifically, our IRHED model is divided into two stages: the prediction and the refinement stage. In the first prediction stage, we employ the state-of-the-art Holistically-nested Edge Detection (HED) [1] network to predict initial edge map. To incorporate the inherent structural correlation in input images, the refinement stage is utilized to enforce the structural constraints. In the refinement stage, we concatenate the output boundary map from the prediction stage with the input image, and fit them into another network with similar architecture as the first stage. In this case, the refinement network is tasked to correct the error between the prediction of the first stage and the ground truth in a deeply supervised manner. Learning from the residual error has shown to be successful for a number of visual recognition problems [3]. And we found it critical for the performance of our model. By learning for correcting edges, our refinement network implicitly captures global structural constraints, such as object shape.

To test our model, we evaluate IRHED network on the widely used Berkeley Segmentation Dataset and Benchmark (BSDS500) dataset [4, 5]. Extensive experiments demonstrate the effectiveness of the iterative residual learning scheme. IRHED also achieves new state-of-the-art result with an ODS of 0.804, slightly outperforming human-level performance (ODS 0.803). We also demonstrate the benefit of structured edge map for higher-level task, such as object proposal generation.

2. METHOD

This section presents our IRHED model. We start by introducing our baseline HED [1], which is the basis of our method. We then present our full iterative model.

2.1. HED network

HED [1] employs a single-stream deep ConvNets with multiple side outputs, and computes multi-scale intermediate predictions at all side-outputs, then fuses these intermediate edge maps as the final output. We use Enhanced-HED, an enhanced version of HED, which improves original HED in two folds [6]. Specifically, Enhanced-HED first adds two additional convolutional layers at each side output to enable robust representation capacity. Moreover, Enhanced-HED uses larger kernel size in higher layers as [6, 7]. For the five blocks

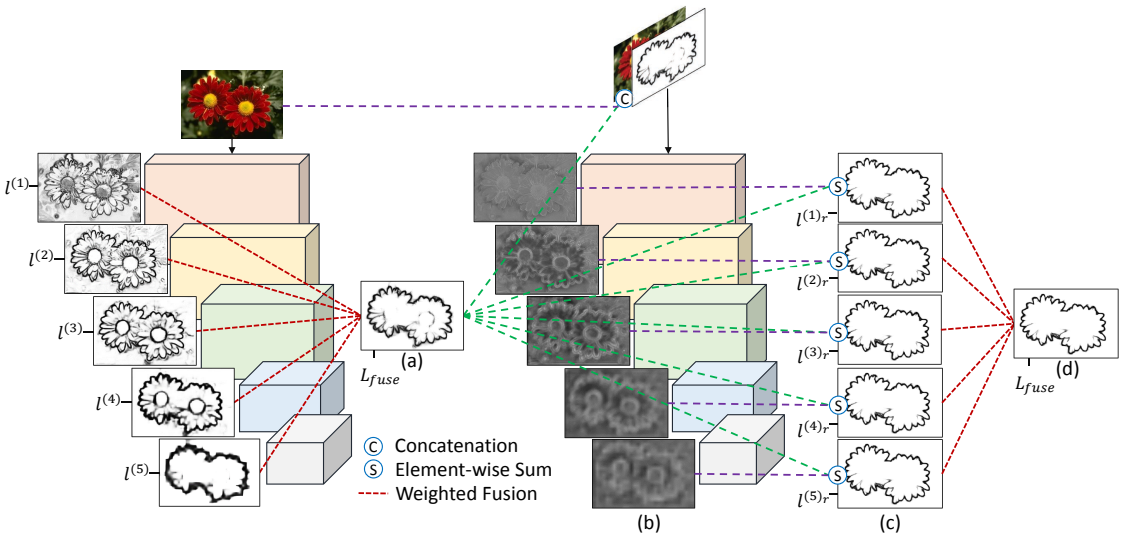

Fig. 1. IRHED network architecture. IRHED consists of two stages: the prediction and the refinement stage. The prediction stage is a feed-forward network (HED) for generating initial predictions. The predicted boundary map is further concatenated with the original input image and sent to the refinement stage. The refinement network seek to fit the residual error between the prediction of the first stage and the ground truth in a deeply supervised way. And the final prediction in each side-output ((c)) is the element-wise sum of the initial prediction ((a)) and corresponding residual error map ((b)). Finally we use the weighted fusion of the five side-output predictions as the final output ((d)).

of VGG16 net [8], the kernel size of the newly added convolutional filters are set as $3 \times 3, 3 \times 3, 5 \times 5, 5 \times 5, 7 \times 7$.

2.2. Iterative Residual HED Network

Instead of the independent prediction scheme of HED, inspired from methods of iterative segment [9, 10, 11], and residual learning [3], we propose a novel IRHED network to capture structured cues in input images. Figure 1 shows the details of our IRHED network. Our model learns the intrinsic structural knowledge of edge in a deeply supervised iterative residual learning scheme.

Specifically, our method is divided into two stages: the prediction and the refinement stage. With the same architecture as Enhanced-HED, the first prediction stage produces initial boundary map $\hat{Y}$ (the activation map before $\sigma(.)$ is $\hat{A}$). Then we concatenate the initial activation map $\hat{A}$ and the input image X as the input of the second refinement stage: $X_r = X \oplus \hat{A}$. The refinement stage duplicates the structure of the first prediction stage, and produces a final boundary map $\hat{Y}_r$ by fusing side-output predictions $\hat{A}_r^{(1)}, ..., \hat{A}_r^{(M)}$ from M scales. However, the refinement network learns to correct previous mistake of the first prediction stage at its M side-outputs in a deeply supervised manner.

We denote the union of convolutional weights in base refinement network as W_r, and the weights in M side-outputs as $w_r = (w^{(1)_r}, ..., w^{(M)_r})$. The previous prediction $\hat{A}$ is added to the multiple side-predictions $\hat{A}_r^{(m)}$ of the refinement network. More concretely, the loss function for refinement network is

$$L_{HED}(W, w, h; \hat{A}) = L_{side}(W_r, w_r; \hat{A}) + L_{fuse}(W_r, w_r, h_r) \tag{1}$$

where L_{side} and L_{fuse} are the losses for side outputs and fused output, respectively. This loss function considers the initial prediction $\hat{A}$. This is done by

$$\begin{aligned} L_{fuse}(W_r, w_r, h_r) &= CE(Y, \hat{Y}_r) \\ L_{side}(W_r, w_r; \hat{A}) &= \sum_{m=1}^{M} l^{(m)_r}(W_r, w_r^{(m)}; \hat{A}) \end{aligned} \tag{2}$$

where $l^{(m)_r}(W_r, w_r^{(m)}; \hat{A})$ is our residual weighted cross entropy loss, given by

$$\begin{aligned} l^{(m)_r}(W_r, w_r^{(m)}; \hat{A}) = &-\beta \sum_{j \in Y_+} \log\left(P(y_j = 1 \; X_r, \hat{A}; W_r, w_r^{(m)})\right) \\ &- (1-\beta) \sum_{j \in Y_-} \log\left(P(y_j = 0 \; X_r, \hat{A}; W_r, w_r^{(m)})\right) \end{aligned}$$

where $P(y_j = 1 \; X_r, \hat{A}; W_r, w_r^{(m)};) = \sigma(\hat{a}_j + \hat{a}_{r_j}^{(m)})$, and $\hat{a}_j, \hat{a}_{r_j}^{(m)}$ are the activation values at pixel j of activation map $\hat{A}$ and $\hat{A}_r^{(m)}$, respectively. Moreover, $\beta = Y_- \; / \; Y$ and $1 - \beta = Y_+ \; / \; Y$. Y_+ and Y_- are the edge and non-edge ground truth label sets. This is to address the challenging issue of the imbalanced edge and non-edge pixels.

	Nc	Plane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	MBike	Person	Plant	Sheep	Sofa	Train	TV	Global
SE-MCG	100	71.2	38.6	74.7	64.5	53.7	70.6	48.9	82.7	55.4	79.1	66.7	78.8	70.6	65.4	60.4	49.9	71.9	73.7	71.3	76.2	63.8
HED-MCG	100	73.0	39.3	79.9	70.0	56.2	68.8	50.7	86.9	55.1	81.6	62.7	85.0	73.7	63.9	59.2	56.5	76.2	72.6	68.6	73.6	64.9
IRHED-MCG	100	77.4	41.6	84.3	71.6	57.1	72.2	52.4	89.3	56.1	82.0	65.4	87.4	76.7	70.2	61.4	55.6	79.0	78.3	73.8	72.0	**67.2**
SE-MCG	5138	83.0	51.2	86.4	79.7	78.1	81.8	77.4	90.8	74.5	88.7	84.2	88.1	81.4	78.6	79.7	77.5	86.7	87.8	81.9	90.3	80.8
HED-MCG	3051	83.3	52.2	87.8	81.5	77.3	80.4	78.3	94.2	74.2	90.8	82.6	92.2	84.0	76.9	78.4	75.5	90.0	87.6	80.0	87.9	80.8
IRHED-MCG	2687	86.4	53.2	89.9	82.4	77.2	83.3	81.1	94.6	76.2	90.6	83.4	92.5	84.9	79.8	79.8	78.1	91.2	89.1	84.1	89.1	**82.4**

Table 1. Object proposals performance on VOC12 val set with top-100 and all proposals. Mean Jaccard index at instance level for per class is reported. Our IRHED clearly outperforms other edge detectors.

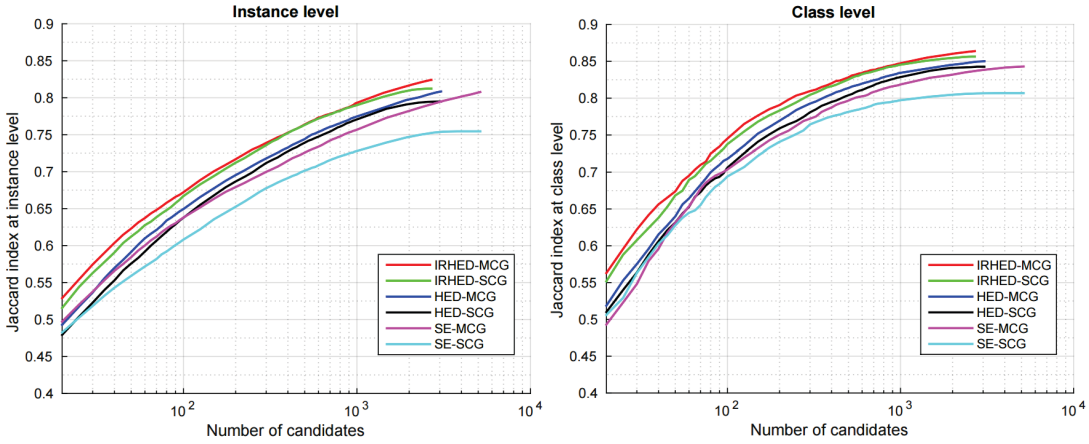

Fig. 4. Object proposals performance with different edge detectors (SE, HED, IRHED) on VOC12 val set at instance level (left) and class level (right). IRHED-MCG achieves the best results.

The final output edge map is the weighted fusion of the M side-predictions of the refinement network. By learning to fit the residual error of initial prediction in a deeply supervised way, our whole model effectively captures global structural constraints in input images. We note that our method differs significantly from [12], which simply concatenates the multiple side-predictions in earlier stage as the input to later stage.

3. EXPERIMENTS

We first benchmark our IRHED model on the widely-used BSDS500 dataset [5]. Then we demonstrate the benefit of structured edge map for object proposal generation.

3.1. Boundary Detection

Datasets and Metrics: We evaluate IRHED on BSDS500 dataset, which consists of 200 images for training, 100 for validation, and 200 for testing. In consistency with previous works [1, 13], we use the train and validation set for training and report results on the test set, evaluate the performance by three standard metrics: ODS, OIS and average precision (AP).

Abalation Sutudy: First, we conduct ablation analysis of IRHED network. Table 2 shows the detailed results. For ODS, Enhanced-HED improves over HED by 1.5%, and IRHED outperforms Enhanced-HED by another 0.1%. Moreover, we explore strategies of multi-scale testing. We experiment with MultiScale (1/2x, 1x, 2x) for Enhanced-HED and IRHED. We first resize the image and feed it into the network. The outputs are resized to the original image resolution and averaged to produce final boundary map. With multi-scale processing, both Enhanced-HED and IRHED improves over the single scale version. In addition, IRHED-MultiScale outperforms Enhanced-HED-MultiScale by 0.4%. This further shows the effectiveness of our iterative learning scheme.

Method	ODS	OIS	AP
HED	.780	.797	.829
Enhanced-HED	.795	.812	.850
IRHED	.796	.814	.838
Enhanced-HED-MultiScale	.800	.818	.867
IRHED-MultiScale	**.804**	**.824**	**.869**

Table 2. Generic boundary detection: ablation study.

Comparison with State-of-the-arts: We further compare the best performing IRHED (IRHED-MultiScale) to state-of-the-art methods in Table 3. Figure 2 shows the Precision-Recall curves. IRHED achieves an ODS of 0.804 on the standard BSDS500, higher than all previous top-performing methods [13, 14]. This result also slightly surpasses human

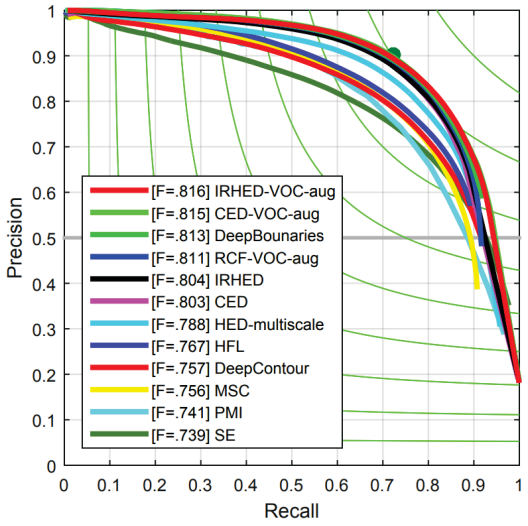

Fig. 2. Generic boundary detection: Precision-Recall curves of different methods on BSDS500 dataset.

Method	ODS	OIS	AP
Human	.8027	.8027	-
gPb-owt-ucm[5]	.726	.757	.696
SE-Var[16]	.746	.767	.803
PMI[17]	.741	.769	.799
MSC [18]	.756	.776	.787
CSCNN [19]	.756	.775	.798
DeepContour [20]	.757	.776	.790
HFL [21]	.767	.788	.795
HED [1]	.788	.808	.840
RDS [14]	.792	.810	.818
RCF-VOC-aug [2]	.811	.830	-
DeepBounaries (VOC-aug) [15]	.813	.831	.866
CED[13]	.803	.820	**.871**
CED-VOC-aug[13]	.815	**.833**	**.889**
IRHED	**.804**	**.824**	.869
IRHED-VOC-aug	**.816**	.831	.885

Table 3. Generic boundary detection: comparison to state-of-art methods on BSDS500.

benchmark on BSDS500 dataset with ODS 0.8027. After augmenting the standard training set with VOC images as [13, 15], IRHED gives better results with ODS 0.816. This new form of IRHED is denoted as IRHED-VOC-aug. Finally, Figure 3 shows visual comparison of boundaries from HED and IRHED before non-maximal suppression (NMS). Clearly, IRHED can produce crisp image boundaries.

3.2. Benefits for Object Proposal Generation

In this part, we show that the task of object proposal generation can benefit from the structured edge map. Object proposal generation is an important mid-level vision task, which is the first step for other higher-level tasks, such as object segmentation. Multi-scale Combinatorial Grouping (MCG) and its single scale version (SCG) [22] are employed to generate object proposals. In order to generate object proposals, MCG builds a hierarchical grouping of object boundaries, which employs the Structured Edge (SE) [23] as the default edge detector. We simply replace SE with HED [1] and our IRHED. Note that these edge detectors are trained only on the BSDS500 dataset. We benchmark three edge detectors (SE, HED, IRHED) with both MCG and SCG.

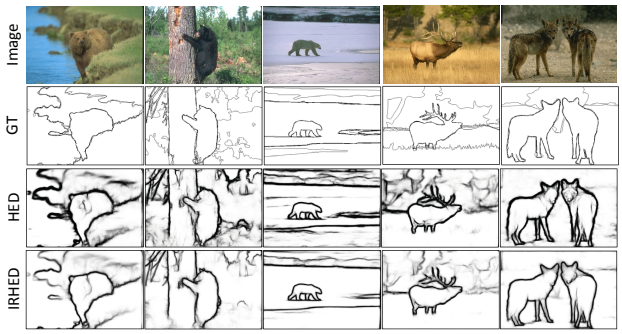

Fig. 3. Visualization of edge detection results from different methods. First two rows show the original images and ground-truths edges. The next two rows are the raw edge maps (before NMS) of HED, IRHED, respectively. Edge maps from IRED are sharper and cleaner than HED.

As shown in Figure 4, we report the mean Jaccard index at both instance level and class level with respect to the number of proposals. IRHED-MCG achieves the best results at both metrics for all number of proposals. We also shows the Jaccard index at instance level for each class in Table 1. Our IRHED-MCG outperforms HED-MCG by 2.3% with top-100 proposals, and by 1.6% with all proposals. These results demonstrate the benefits of structured edges for object proposal generation.

4. CONCLUSION

We address the problem of enforcing the structural constraints into the ConvNet-based edge detector. We present a novel IRHED network to learn the inherent structural correlations. Extensive experimental results demonstrate the validity of our deeply supervised iterative refinement scheme. IRHED network also achieves state-of-the-art results on the BSDS500 dataset.

Acknowledgement

This work is funded by the National Key Research and Development Program of China (Grant No.2016YFB1001005), the National Natural Science Foundation of China (Grant No.61673375, Grant No.61602485), and the Projects of Chinese Academy of Science (Grant No.QYZDB-SSW-JSC006, Grant No.173211KYSB20160008).

5. REFERENCES

[1] Saining Xie and Zhuowen Tu, "Holistically-nested edge detection," in *ICCV*, 2015.

[2] Yun Liu, Ming Ming Cheng, Xiaowei Hu, Kai Wang, and Xiang Bai, "Richer convolutional features for edge detection," in *CVPR*, 2017.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[4] David R Martin, Charless C Fowlkes, and Jitendra Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *TPAMI*, 2004.

[5] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, 2011.

[6] Qibin Hou, Ming Ming Cheng, Xiao Wei Hu, Ali Borji, Zhuowen Tu, and Philip Torr, "Deeply supervised salient object detection with short connections," in *CVPR*, 2017.

[7] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun, "Large kernel matters – improve semantic segmentation by global convolutional network," in *CVPR*, 2017.

[8] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[9] Zhuowen Tu and Xiang Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation," *TPAMI*, 2010.

[10] Daniel Munoz, J Andrew Bagnell, and Martial Hebert, "Stacked hierarchical labeling," in *ECCV*, 2010.

[11] Ke Li, Bharath Hariharan, and Jitendra Malik, "Iterative instance segmentation," in *CVPR*, 2016.

[12] Wei Shen, Bin Wang, Yuan Jiang, Yan Wang, and Alan Yuille, "Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection," in *ICCV*, 2017.

[13] Yupei Wang, Xin Zhao, and Kaiqi Huang, "Deep crisp boundaries," in *CVPR*, 2017.

[14] Yu Liu and Michael S Lew, "Learning relaxed deep supervision for better edge detection," in *CVPR*, 2016.

[15] Iasonas Kokkinos, "Pushing the boundaries of boundary detection using deep learning," *ICLR*, 2016.

[16] Piotr Dollár and C Lawrence Zitnick, "Fast edge detection using structured forests," *TPAMI*, 2015.

[17] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson, "Crisp boundary detection using pointwise mutual information," in *ECCV*, 2014.

[18] Amos Sironi, Engin Türetken, Vincent Lepetit, and Pascal Fua, "Multiscale centerline detection," *TPAMI*, 2016.

[19] Jyh-Jing Hwang and Tyng-Luh Liu, "Pixel-wise deep learning for contour detection," *arXiv preprint arXiv:1504.01989*, 2015.

[20] Wei Shen, Xinggang Wang, Yan Wang, Xiang Bai, and Zhijiang Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *CVPR*, 2015.

[21] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani, "High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision," in *ICCV*, 2015.

[22] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik, "Multiscale combinatorial grouping," in *CVPR*, 2014.

[23] Piotr Dollár and C. Lawrence Zitnick, "Structured forests for fast edge detection," in *ICCV*, 2013.