Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Automated lesion segmentation in fundus images with many-to-many reassembly of features

Qing Liu[a,1], Haotian Liu[a,1], Wei Ke[b,*], Yixiong Liang[a,*]

[a] School of Computer Science, Central South University, Changsha, Hunan, 410083, China
[b] Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China

ARTICLE INFO

Article history:
Received 14 June 2022
Revised 20 October 2022
Accepted 18 November 2022
Available online 28 November 2022

Keywords:
Feature reassembly
Upsampling operator
Downsampling operator
Lesion segmentation
Fundus image analysis

ABSTRACT

Existing CNN-based segmentation approaches have achieved remarkable progresses on segmenting objects in regular sizes. However, when migrating them to segment tiny retinal lesions, they encounter challenges. The feature reassembly operators that they adopt are prone to discard the subtle activations about tiny lesions and fail to capture long-term dependencies. This paper aims to solve these issues and proposes a novel Many-to-Many Reassembly of Features (M2MRF) for tiny lesion segmentation. Our proposed M2MRF reassembles features in a dimension-reduced feature space and simultaneously aggregates multiple features inside a large predefined region into multiple output features. In this way, subtle activations about small lesions can be maintained as much as possible and long-term spatial dependencies can be captured to further enhance the lesion features. Experimental results on two lesion segmentation benchmarks, *i.e.*, DDR and IDRiD, show that 1) our M2MRF outperforms existing feature reassembly operators, and 2) equipped with our M2MRF, the HRNetV2 is able to achieve substantially better performances and generalisation ability than existing methods. Our code is made publicly available at https://github.com/CVIU-CSU/M2MRF-Lesion-Segmentation.

1. Introduction

Lesions such as microaneurysms (MAs), hemorrhages (HEs), hard exudates (EXs) and soft exudates (SEs) in fundus images (see Fig. 1(a-c)) are important manifestations for retinal fundus disease diagnosis and severity grading by ophthalmologists. For example, the presence of MAs is always treated as the early sign of diabetic retinopathy [1] while MAs together with HEs, EXs and SEs are symptoms for moderate nonproliferative diabetic retinopathy [2]. The number and locations of EXs are also evidences for severity grading of diabetic macular edema [2]. However, manual lesion segmentation in fundus images is time-consuming and labour intensive. To liberate ophthalmologists from heavy workload, automated lesion segmentation has become a trend [3].

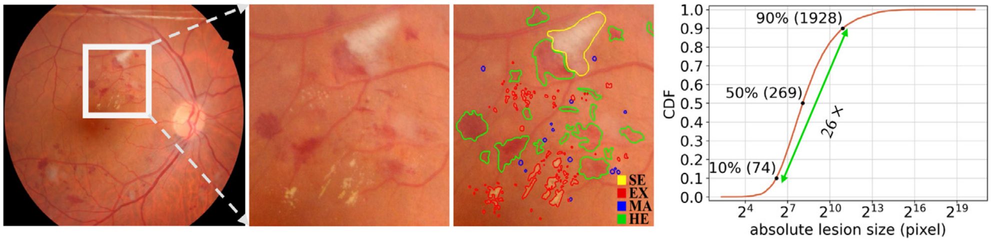
Inspired by the extraordinary success in natural scene image segmentation [5–7], a few modern semantic segmentation algorithms have been migrated to segment the retinal lesions in fundus images [3,8]. Despite their success, however, there is still a significant performance gap between natural and fundus images. Specifically, the HRNetV2 [5] can achieve 81% mean intersection over union (mIoU) on Cityscapes [9], which is a widely-used natural semantic segmentation dataset consisting of 19 classes. However, when fine-tuning the HRNetV2 [5] for lesion segmentation on IDRiD [4], the mIoU of four types of lesions decreases to 47%. Why is lesion segmentation so much harder than natural scene image segmentation?

Two possible factors behind this significant performance gap are the extreme small size of lesions and large size variation across them. For clear illustration, we count the lesion size in images size of 4288 × 2848 from IDRiD [4] and plot its cumulative distribution function in Fig. 1(d). As shown, 50% lesions are less than 269 pixels. Such small size of lesions rises an extreme challenge for CNN-based segmentation approaches to learn discriminative representations with enough spatial information. To make matters worse, the smallest 10% lesions in IDRiD [4] only contain less than 74 pixels while the largest 10% lesions contain more than 1928 pixels, which shows an enormous size variation. Intuitively, small lesions require that CNNs maintain as enough local information as possible within a small receptive field. On the contrary, large lesions require CNNs exploit long-range contexts over a large receptive field. This demand contradiction presents another challenge for retinal lesion segmentation. Moreover, in a fundus image, different types of

* Corresponding author.
E-mail addresses: qing.liu.411@gmail.com (Q. Liu), haotianliu@csu.edu.cn (H. Liu), wei.ke@xjtu.edu.cn (W. Ke), yxliang@csu.edu.cn (Y. Liang).

[1] Equal Contribution.

https://doi.org/10.1016/j.patcog.2022.109191
0031-3203/© 2022 Elsevier Ltd. All rights reserved.



(a) Retinal fundus image (b) Zoom into white box in (a) (c) Lesion regions (d) Cumulative distribution function of lesion sizes

Fig. 1. **(a-c)** An example for colour fundus images and lesion regions from IDRiD [4] dataset. Four types of lesions, i.e. soft exudates (SE), hard exudates (EX), microaneurysms (MA) and hemorrhages (HE) are delineated in yellow, red, blue and green respectively. **(d)** Cumulative distribution function of lesion size on IDRiD [4] dataset, which shows that many lesions are extremely small and the scale variation across them is enormous. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

lesions with large size variation often cluster together, which makes the representation learning more challenging.

Features reassembly operators, i.e. downsampling and upsampling operators, are the crucial components in lots of modern CNN architectures such as [5,10,11]. Downsampling via either max-pooling or strided convolution [5,12] can enlarge the size of CNN's receptive field and reduce the spatial resolution of feature maps [13,14]. Conversely, upsampling operators, such as bilinear interpolation or deconvolution [10,15], are essential to recover the spatial resolution for better segmentation. Both these downsapling and upsampling feature reassembly operators assume that output features at each location are independent of each other and follow a many-to-one pipeline, accordingly we name them as Many-to-One Reassembly of Features (M2ORF).

Specific to the implementation, for downsampling, M2ORF operators can be essentially treated as a compound of two successive operations: first filtering the input feature maps with small sized importance weighting kernels, e.g, max-pooling kernel in max-pooling operator and learnable kernels in strided convolution, and then uniformly subsampling the filtered feature maps to decrease the resolution according to the downsample factor. However, the activations that are not sampled are directly discarded by the uniform subsample. This may lead to the loss of discriminative information about small lesions, and further result in misidentification on small lesions. For upsampling, M2ORF upsample operators can also be decomposed into two successive operations: first interleaving the input feature maps with zeros to increase their resolution, then filtering with importance weighting kernels in small size, e.g., distance-based kernels in bilinear interpolation and learnable kernels in deconvolution, to recover the information. However, once subtle lesion activations are lost, it is almost impossible to recover them via upsampling operators. Additionally, the activations of some very small lesions are subtle and long-term dependencies across lesions are highly desired to enhance these subtle activations. However, M2ORF operators leverage small sized importance kernels to generate output features one-by-one independently, where long-term dependencies across small lesions are ignored.

To address above issues raised by M2ORF operators in previous CNN-based segmentation approaches within the context of lesion segmentation, we propose a unified RF operator, termed as Many-to-Many Reassembly of Features (M2MRF). It reassembles multiple features inside a large predefined region into multiple output features simultaneously via learning and the number of output features is controlled by the sample rate. In this way, our M2MRF can maintain the discriminative information about small lesions as much as possible as it bypasses the key step to decrease the feature resolution in M2ORF, i.e., the uniform subsample. Besides, our M2MRF can exploit long-term dependencies across lesions to mutually enhance lesion activations as the output features are reassembled with predefined feature region in large size simultaneously. We demonstrate the effectiveness of our M2MRF on two public lesion segmentation datasets, i.e. DDR [8] and IDRiD [4]. Experiments show that our M2MRF outperforms state-of-the-art feature reassembly operators [16–18]. Our M2MRF also shows competitive performances and generalisation ability comparing to state-of-the-art segmentation methods. Particularly, equipped with HRNetV2 [5], our M2MRF exhibits significant improvements with negligible increase of parameters and inference time.

The rest of this paper is organised as follows. The most related works are briefly reviewed in Section 2, and our proposed M2MRF and its application on lesion segmentation are described in Section 3. Section 4 presents the experiments and analysis. The conclusion is presented in Section 5.

2. Related work

Lesion segmentation falls into the research field of semantic segmentation. Therefore, we first give a brief overview of recent semantic segmentation approaches, and then we respectively elaborate existing lesion segmentation approaches and feature reassemble operators and explicitly distinguish them from the proposed method.

2.1. Deep semantic segmentation

The development of deep semantic segmentation can be mainly classified into four veins. The first vein focuses on how to produce and aggregate multi-scale representations. In fully convolutional networks (FCN) [10], a natural solution which reuses middle-level features to compensate for spatial details in high-level features is provided. In pyramid scene parsing network (PSPNet) [19], a pyramid pooling module is proposed to produce and fuse multiple features under different pyramid scales while Deeplabv3+ [20] adopts atrous spatial pyramid pooling for multi-scale feature map production. The second vein focuses on high-resolution representation learning. For example, 'encoder-decoder' style networks [11,21] gradually recover high-resolution representations from low-resolution representations with upsampling operations. Instead of resolution recovery, high-resolution network (HRNet) [5] maintains high-resolution representations via gradually adding high-to-low resolution convolution streams one-by-one and fusing them in parallel. The third vein introduces attention mechanism and variant modules such as non-local network [22] and disentangled non-local neural networks (DNL) [23] are developed to explore spatial dependencies. They aggregate pixel-level pairwise spatial dependencies with an attention map which is estimated based on self-similarity, e.g. dot-product similarity in non-local [22], to enhance the features. Usually, the attention map is only dependent on the feature maps and computationally intensive. Differently, our M2MRF directly reassembles features in local patch with importance kernels which are dependent on the whole dataset and computationally efficient. More recently, a new vein emerges which employs vision transformers, e.g., Swin [7] and Twins [6], for semantic segmentation. These approaches have not only achieved extraordinary success on natural scene image segmentation, but also paved the way to lesion segmentation in fundus images.

2.2. Lesion segmentation in fundus images

The development of lesion segmentation originates from one-type lesion segmentation with hand-crafted features [24,25] or deep features [26]. For example, the top three approaches participating in the 2018 ISBI grand challenge 'Diabetic Retinopathy - Segmentation and Grading' [4], i.e., VRT, PATech and iFLYTEK, follow the paradigm of one CNN model for one lesion type, and separately train four patch-level CNN models for the segmentation of SEs, EXs, MAs and HEs according to their characteristics. As a result, during inference phase, obtaining four-type lesion segmentation results requires four times forward propagations for each fundus image, which is high computational cost.

Recent solutions to multi-type lesion segmentation treat the task as a multi-label dense classification task and segment multi-type of lesions simultaneously in a unified model such that fundus images only need to be forwarded once during inference phase. In [8], two approaches for natural scene image analysis, i.e., holistically-nested edge detection network (HED) [27] and DeeplabV3+ [20], are directly fine-tuned as the benchmark of four-type lesion segmentation dataset DDR. To handle the class imbalance issue, Guo et al. [28] propose multi-channel bin loss function and develop L-Seg on the top of HED [27]. However, these methods adopt max-pooling or strided convolution as downsample operators, which may lead to loss of discriminative details, especially of small sized objects [14].

More recently, researchers rely on auxiliary databases to capture richer information for lesion segmentation. Particularly, [29,30] rely on both the lesion segmentation database and diabetic retinopathy grading database to capture relations between lesions and disease grades for collaborative learning of lesion segmentation and diabetic retinopathy grading. Similarly, in relation transformer network (RTNet) [3], Huang et al. rely on two auxiliary databases to train a vessel segmentation model and generate pseudo vessel masks for fundus images in lesion segmentation database. In this way, they are able to exploit inter-class relations between lesions and vessels for lesion segmentation. Capturing richer information from auxiliary databases contributes to lesion segmentation. However, labeling those auxiliary databases by ophthalmologists is expensive and time-consuming.

2.3. Feature reassembly operators in deep networks

The feature reassembly operator, including downsample and upsample operators, is essential in modern deep networks [5,10,20]. Among them, strided max-pooling [31] and strided convolution are widely adopted for feature downsampling, while bilinear interpolation, deconvolution and unpooling [32] are widely adopted for upsampling. The general idea of these operators is to generate a feature vector for each output location via reassembling multiple features inside a predefined region with importance kernels. Particularly, importance kernels for strided max-pooling, unpooling and bilinear interpolation are hand-crafted and feature maps are processed channel-by-channel efficiently. However, they ignore the context dependencies across channels and the diversity of local patterns. Instead, the importance kernels for strided convolution and deconvolution are learned. Their dimension depends on the input features, which is always high in CNNs. This makes the computation burden of reassembly heavy when large importance kernels are used. Thus it is difficult to reassemble features from a large region.

Recently, novel ideas about learning-based feature reassembly operators are proposed. Local importance-based pooling (LIP) [14] learns adaptive importance kernels based on inputs to enhance the discriminative features. In content-aware reassembly of features (CARAFE) [33] and CARAFE++ [16], content-aware kernels for each output position is learned according to input features for feature reassembly. Similarly, IndexNet [17] learns importance kernels from feature encoder to guide the feature reassembly in both feature encoder and decoder. Instead of reassembling features inside a predefined region, deformable RoI pooling [34] reassembles features in an adaptive region which is learned from the input features. To model the affinity information, affinity-aware upsampling (A^{2}U) [18] is proposed to learn importance kernels according to second-order features for feature reassembly. However, these learning-based operators usually follow the paradigm of many-to-one feature reassembly and may fail to capture long-range dependencies for lesion segmentation.

3. Method

In this section, we first give a simple analysis for M2ORF operators and then detail our M2MRF. Finally we take HRNetV2 [5] as an example and present how to integrate our M2MRF into CNN architectures for lesion segmentation.

3.1. Analysis for M2ORF operators

Given the input feature map $\mathbf{X} \in \mathcal{R}^{H\times W\times C}$ and sample rate δ where $\delta > 0$, the goal of feature reassembly is to generate output feature map $\mathbf{Y} \in \mathcal{R}^{\lfloor \delta H\rfloor \times \lfloor \delta W\rfloor \times C}$ via finding a function mapping Φ parametrised by importance kernels $\mathbf{W}$:

$$\mathbf{Y} = \Phi(\mathbf{X}; \mathbf{W}) \ . \tag{1}$$

Here $\delta < 1$ for downsampling and $\delta > 1$ for upsampling.

To make the computation efficient, it is usually degraded to a many-to-one local sampling problem, i.e. reassembling multiple features in a predefined local region to one output feature. Specifically, for any output feature $\mathbf{y} \in \mathcal{R}^C$ at location (i', j') in $\mathbf{Y}$, most existing methods assume that there is a corresponding source feature $\mathbf{x} \in \mathcal{R}^C$ at location (i, j) in $\mathbf{X}$, where $i = \lfloor i'/\delta \rfloor$ and $j = \lfloor j'/\delta \rfloor$. They follow three steps to obtain $\mathbf{y}$: (1) setting/learning a local region $\Omega_\mathbf{x}$ according to $\mathbf{x} \in \mathcal{R}^C$; (2) with $\Omega_\mathbf{x}$, setting/learning corresponding importance kernels $\mathbf{W}_{i',j'}$; (3) obtaining $\mathbf{y}$ via

$$\mathbf{y} = \Phi(\mathbf{P}_{\Omega_\mathbf{x}}; \mathbf{W}_{i',j'}) \ , \tag{2}$$

where $\mathbf{P}_{\Omega_\mathbf{x}}$ denotes features in $\Omega_\mathbf{x}$. Finally, the whole output feature map $\mathbf{Y}$ is constructed by simply putting $\mathbf{y}$ at each location together.

Obviously, M2ORF operators assume that each output feature $\mathbf{y}$ in the whole feature map $\mathbf{Y}$ is independent. For downsampling, the spatial locations of the output feature maps are obtained via uniformly subsampling the spatial locations of the input features, which may result in the loss of spatial details. Additionally, $\Omega_\mathbf{x}$ in (2) is usually small, which may make them fail to exploit long-term dependencies.

3.2. Many-to-many reassembly of features

To maintain the spatial details and exploit long-term dependencies as much as possible, we need to bypass the uniform subsample and increase the importance kernel size to enlarge the receptive field size. To this end, we propose many-to-many reassembly of features (M2MRF). Next, we will introduce our M2MRF module in detail.

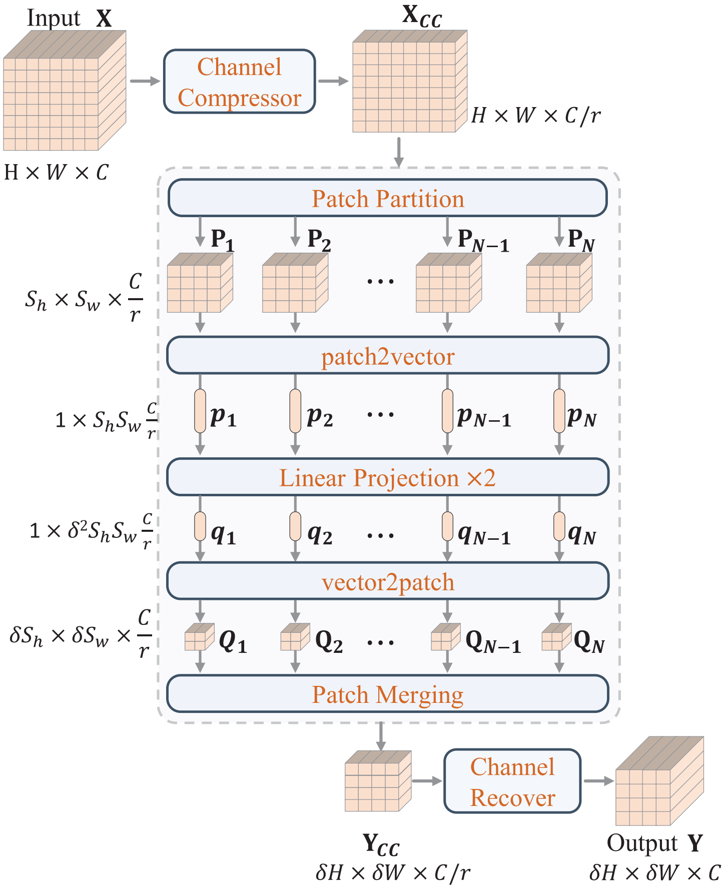


Fig. 2. An overview of our Many-to-Many Reassembly of Features (M2MRF). $\{\mathbf{P}_l\}_{l=1}^{L}$ and $\{\mathbf{Q}_l\}_{l=1}^{L}$ are feature patches of size $S_h \times S_w \times \frac{C}{r}$ and $\delta S_h \times \delta S_w \times \frac{C}{r}$ respectively, where $L(= \lceil H/S_h \rceil \cdot \lceil W/S_w \rceil)$ is the number of patches. $\{\mathbf{p}_l\}_{l=1}^{L}$ and $\{\mathbf{q}_l\}_{l=1}^{L}$ are feature vectors. In this figure, a feature map of size $H \times W \times C$ is downsampled by a factor of $\delta(= 1/2)$. (Best view in colour).

Module overview. Our goal is to reassemble the input feature map $\mathbf{X}$ of size $H \times W \times C$ to feature map $\mathbf{Y}$ of size $\lfloor \delta H \rfloor \times \lfloor \delta W \rfloor \times C$ according to the given sample rate δ. Fig. 2 illustrates an overview of our proposed M2MRF module. First, a channel compressor is performed on $\mathbf{X}$ to reduce its channel dimension from C to $\frac{C}{r}$ for computational efficiency. This also allows us to reassemble features within a large region. We denote the output as $\mathbf{X}_{CC}$. Then we partition $\mathbf{X}_{CC}$ into feature patches $\{\mathbf{P}_l\}_{l=1}^{L}$ of size $S_h \times S_w \times \frac{C}{r}$, where $L = \lfloor H/S_h \rfloor \cdot \lfloor W/S_w \rfloor$. Our proposed M2MRF is performed on each feature patch and outputs $\{\mathbf{Q}_l\}_{l=1}^{L}$ of size $\lfloor \delta S_h \rfloor \times \lfloor \delta S_w \rfloor \times \frac{C}{r}$ simultaneously. In this way, the uniform sample operation is naturally bypassed. Thereafter, those patches are merged into feature map $\mathbf{Y}_{CC}$ of size $\lfloor \delta H \rfloor \times \lfloor \delta W \rfloor \times \frac{C}{r}$. Finally, we recover the feature channel to C via channel recover. For channel compressor and recover, we simply implement them with a 1×1 regular convolution layer.

M2MRF. With a local feature patch $\mathbf{P} \in \{\mathbf{P}_l\}_{l=1}^{L}$, the goal is to generate $\mathbf{Q}$ of size $\lfloor \delta S_h \rfloor \times \lfloor \delta S_w \rfloor \times \frac{C}{r}$:

$$\mathbf{Q} = \Phi(\mathbf{P}; \mathbf{W}_{patch}) \ . \tag{3}$$

Here we let $M = \lfloor \delta S_h \rfloor \times \lfloor \delta S_w \rfloor, N = S_h \times S_w$, and treat this task as generating M features $\mathbf{Q} = \{\mathbf{y}_m\}_{m=1}^{M}$ from N source features $\mathbf{P} = \{\mathbf{x}_n\}_{n=1}^{N}$ where $\mathbf{y}_m, \mathbf{x}_n \in \mathcal{R}^{1 \times \frac{C}{r}}$. To achieve this, one option is to adopt linear projection, thus Eq. (3) can be expressed as:

$$[\mathbf{y}_1, \cdots, \mathbf{y}_M] = [\mathbf{x}_1, \cdots, \mathbf{x}_N][\mathbf{W}_1, \cdots, \mathbf{W}_M] \ , \tag{4}$$

where $\mathbf{W}_1, \cdots, \mathbf{W}_M$ are parameters of size $\frac{NC}{r} \times \frac{C}{r}$ to be learned. Therefore we have $\mathbf{W}_{patch} = [\mathbf{W}_1, \cdots, \mathbf{W}_M]$ whose size is $\frac{NC}{r} \times \frac{MC}{r}$. For simplicity, we denote $\mathbf{p} = [\mathbf{x}_1, \cdots, \mathbf{x}_N], \mathbf{q} = [\mathbf{y}_1, \cdots, \mathbf{y}_M]$, and rewrite Eq. (4) as:

$$\mathbf{q} = \mathbf{p}\mathbf{W}_{patch} \ . \tag{5}$$

To make use of long-range dependencies, $S_h S_w$ is required to be large. Accordingly, N and M are large, thus $\mathbf{W}_{patch}$ would be a large matrix. On one hand, it is always difficult to optimise such a large matrix. On the other hand, storing a large matrix results in high memory consumption. To reduce the number of parameters to be learned, we decompose $\mathbf{W}_{patch}$ to two small matrices via a two-layer linear projections. Thus Eq. (4) can be rewritten as:

$$\mathbf{q} = (\mathbf{p}\mathbf{W}'_{patch})\mathbf{W}''_{patch} \ , \tag{6}$$

where $\mathbf{W}'_p \in \mathcal{R}^{\frac{NC}{r} \times \frac{NC}{\alpha r}}$ and $\mathbf{W}''_p \in \mathcal{R}^{\frac{NC}{\alpha r} \times \frac{MC}{r}}$ are parameters in the two linear projections, and $\alpha \geq 1$ such that the matrix dimensions of $\mathbf{W}'_{patch}$ and $\mathbf{W}''_{patch}$ are less than $\mathbf{W}_{patch}$ far away.

3.3. Variants of M2MRF for arbitrary sample rate

Feature reassembly is an omnipresent part of modern CNN architectures and widely used for either downsampling or upsampling or both. Usually, in CNN architectures, the sample rate δ is designed as $\frac{1}{2^t}$ for downsampling and 2^t for upsampling where $t \in \{1, 2, \cdots\}$. Particularly, to reach the goal of decreasing feature resolution to $\frac{1}{2^t}$, we propose two options. The first one decreases the feature resolution gradually via successively performing our M2MRF with $\delta = 1/2$ t times. We term it as cascade M2MRF. The second one directly decreases the feature resolution to $1/2^t$ via performing our M2MRF with $\delta = 1/2^t$ once. We term it as one-step M2MRF. Similarly, there are also two options to upsample features

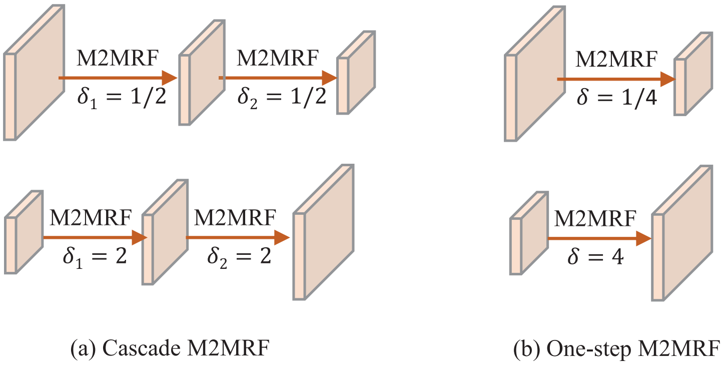


Fig. 3. Two variants of our M2MRF. Here we take $\delta = 1/4$ for downsampling and $\delta = 4$ for upsampling as examples.

with scale factor of 2^t. One is to cascade M2MRF with $\delta = 2$ t times to gradually increase the feature resolution to 2^t. The other is one-step M2MRF with $\delta = 2^t$ which directly increases the feature resolution to 2^t. Taking $t = 2$ for example, Fig. 3 illustrates the cascade M2MRF and one-step M2MRF for downsampling and upsampling respectively.

3.4. Application to lesion segmentation

To segment lesions from fundus images, we propose to incorporate our M2MRF into a high-resolution network, named HRNetV2 [5] as it achieves state-of-the-art performances on semantic segmentation task. To build variants of HRNetV2 [5] with our M2MRF for lesion segmentation, we replace the repeated strided convolution and bilinear interpolation in HRnetV2 with our M2MRF. For the segmentation head, we treat the task as a multi-label classification problem and employ multiple binary classifiers to obtain probability maps for each lesion class. We term this variant as M2MRF-HRNetV2. Considering the extremely class imbalance between lesion and background pixels, we adopt Dice loss [35] to train M2MRF-HRNetV2.

4. Experiments

In this section, we perform a thorough comparison of our M2MRF-HRNetV2 to the state-of-the-art segmentation approaches along with a comprehensive comparison of our M2MRF to state-of-the-art feature reassembly operators on two publicly available datasets for lesion segmentation, i.e. DDR [8] and IDRiD [4].

4.1. Datasets and experiment setup

4.1.1. Datasets

DDR [8] dataset contains 757 colour fundus images of size ranging from 1088 × 1920 to 3456 × 5184 pixels, among which 383 images are used for training, 149 for validation and 225 for testing. In DDR [8], 24154, 13035, 1354 and 10563 connected regions are annotated by ophthalmologists as EX, HE, SE and MA respectively. In our experiments and the reproducing of compared methods, we follow this official dataset division and train models with the training set and validate with the validation set to select optimal hyper-parameters. Then results on testing set are used to make comparisons with state-of-the-arts.

IDRiD [4] dataset contains 81 colour fundus images of size 4288 × 2848 pixels, among which 54 images are used for training and 27 for testing. It is provided by a grand challenge on “Diabetic Retinopathy” – Segmentation and Grading” in 2018. In IDRiD [4], 11716, 1903, 150 and 3505 connected regions are annotated by ophthalmologists as EX, HE, SE and MA respectively. For experiments and reproducing of compared methods on IDRiD, we train models with training set and test on testing set. As IDRiD does not provide validation set, we directly keep the same setting as DDR for hyper-parameters.

4.1.2. Evaluation metrics

We follow the protocol suggested by DDR [8] and IDRiD [4], and report standard metrics including class-wise IoU, mean class-wise IoU (mIoU), class-wise Area Under Precision-Recall curve (AUPR) and mean class-wise AUPR (mAUPR). As Dice coefficient is widely adopted for evaluation of medial image segmentation [26,36], we also report the class-wise Dice coefficient and mean class-wise Dice (mDice).

4.1.3. Network architecture

Our segmentation model M2MRF-HRNetV2 is built on the top of HRNetV2 [5] provided by MMSegmentation [37] with only slight modifications to the feature reassembly operator in the backbone. According to the setting of the number of channels in convolutional layers in HRNetV2 [5], there are several variants, including HRNetV2-W18, HRNetV2-W32, and HRNetV2-W48 etc. In our work, we adopt the widely used one, i.e. HRNetV2-W48 and without extra illustration, HRNetV2-W48 is the default in our experiments. The vanilla HRNetV2 [5] adopts strided convolution layers and bilinear interpolation layers for downsampling and upsampling, respectively. We replace them with our M2MRFs and build our M2MRF-HRNetV2. As there are two variants of M2MRF, i.e one-step and cascade M2MRF, we are able to build four different RF pairs. We denote them by M2MRF-A (one-step M2MRF for both downsampling and upsampling), M2MRF-B (one-step M2MRF for downsampling and cascade M2MRF for upsampling), M2MRF-C (cascade M2MRF for downsampling and one-step M2MRF for upsampling), M2MRF-D (cascade M2MRF for both downsampling and upsampling). Accordingly, there are four variants of our M2MRF-HRNetV2.

4.1.4. Training details

Before feeding images into models, we follow [26,28] and scale images in DDR [8] such that the long side is 1024 pixels. Thereafter, zero padding is used on short side to enlarge its length to 1024 pixels. Following [26,28], images in IDRiD [4] are resized to 1440 × 960 pixels. We follow [5,26,28] and use three data augmentation tricks: multi-scaling (0.5-2.0), rotation (90°, 180° and 270°) and flipping (horizontal and vertical).

We initialize parameters associated with both M2MRF and dense classification layers with Gaussian distribution with zeros mean and standard deviation of 0.01 and the rest with the pre-trained model on ImageNet. We adopt stochastic gradient descent (SGD) as optimiser. Hyper-parameters include: initial learning rate (0.01 poly policy with power of 0.9), weight decay (0.0005), momentum (0.9), batch size (4), and iterations (60k on DDR and 40k

Table 1
Segmentation results on DDR test set [8]. M2MRF-A: (One-Step/One-Step), M2MRF-B:(One-Step/Cascade), M2MRF-C:(Cascade/One-Step), M2MRF-D: (Cascade/Cascade). †: RTNet leverages two extra vessel segmentation datasets during training. All results are averaged over three repetitions. Results marked in **bold** are best ones while those in *italic* are second best ones.

Methods	AUPR					IoU					Dice				
	mAUPR	EX	HE	SE	MA	mIoU	EX	HE	SE	MA	mDice	EX	HE	SE	MA
L-seg [28]	32.08	55.46	35.86	26.48	10.52	-	-	-	-	-	-	-	-	-	-
Dual-PSPNet [26]	-	54.24	-	-	-	-	39.85	-	-	-	-	56.99	-	-	-
RTNet † [3]	33.62	56.71	36.56	29.43	11.76	-	-	-	-	-	-	-	-	-	-
HED [27]	42.97	61.40	43.19	46.68	20.61	27.17	39.50	27.09	29.46	12.63	41.79	56.63	42.61	45.50	22.43
DNL [23]	40.14	56.05	47.81	42.01	14.71	24.33	36.39	27.15	25.33	8.46	38.02	53.36	42.71	40.40	15.60
Deeplabv3+ [20]	42.34	62.32	40.79	41.83	24.39	26.47	41.44	23.44	26.46	14.55	40.95	58.59	37.97	41.83	25.40
PSPNet [19]	39.23	57.04	42.71	42.32	14.85	24.31	37.31	24.51	26.64	8.75	37.97	54.35	39.37	42.08	16.09
SPNet [38]	31.91	44.10	38.22	32.93	12.37	16.47	24.19	13.76	20.55	7.38	27.66	38.78	24.13	34.00	13.74
HRNetV2 [5]	45.21	61.55	45.68	46.91	26.70	28.84	41.82	29.01	28.94	15.60	43.95	58.98	44.96	44.86	26.99
Swin-base [7]	46.72	62.71	54.39	46.12	23.67	30.07	42.64	**33.82**	30.62	13.19	45.10	59.79	**50.53**	46.77	23.31
Twins-SVT-B [6]	46.11	59.71	49.96	*52.72*	22.03	29.28	39.70	29.08	**36.24**	12.07	44.15	56.83	45.04	**53.19**	21.54
M2MRF-A (Ours)	**49.94**	**64.17**	54.20	**53.19**	28.21	**31.16**	43.35	30.03	*35.22*	16.06	**46.60**	60.47	46.18	*52.10*	27.67
M2MRF-B (Ours)	*49.42*	*63.88*	**55.47**	50.01	28.33	*30.41*	43.06	*30.56*	32.08	15.95	*45.77*	60.20	*46.81*	48.58	27.51
M2MRF-C (Ours)	48.94	63.59	54.43	49.35	*28.38*	30.09	*43.49*	29.17	31.39	**16.31**	45.40	*60.62*	45.16	47.78	**28.04**
M2MRF-D (Ours)	49.25	**64.17**	*54.72*	49.64	**28.46**	30.27	**44.04**	29.28	31.60	*16.15*	45.57	**61.15**	45.29	48.02	*27.81*

on IDRiD). In what follows, the same setting is adopted to train lesion segmentation models of our M2MRF-HRNetV2 as well as the state-of-the-art methods for a fair comparison.

4.2. Comparison to state-of-the-art segmentation methods

4.2.1. Results on DDR [8] dataset

We compare our M2MRF-HRNetV2 with 11 state-of-the-art segmentation methods: L-seg [28], Dual-PSPNet [26], RTNet [3], HED [27], DNL [23], Deeplabv3+ [20], PSPNet [19], SPNet [38], HRNetV2 [5], Swin-base [7] and Twins-SVT-B [6]. The first three are specifically for lesion segmentation and the rest are methods for general semantic segmentation, among which six are CNN-based and the last two are transformer-based. Comparative results on DDR testing set are listed in Table 1, in which performances of L-Seg [28] and RTNet [3] are directly borrowed from original papers and the rest are obtained via fine-tuning models on DDR [4] training set. All experiments are performed on a high-performance computing center with many Linux system backend computation nodes. For DNL [23], DeepLabv3+ [20], Swin-base [7] and Twins-SVT-B [6] which require massive GPU memory, we conduct experiments with four NVIDIA Tesla V100 SXM2 GPUs with 32GB of memory. For rest of compared methods and our proposed method, experiments are performed with four NVIDIA GeForce RTX 2080 Ti GPUs with 11GB of memory.

From Table 1, we can see that comparing with approaches specifically designed for lesion segmentation, our four variants of M2MRF-HRNetV2 achieve significantly better performances. Among previous CNN-based segmentation methods, the vanilla HRNetV2 [5] achieves the best performance with 45.21% on mAUPR, 28.84% on mIoU and 43.95% on mDice. The reason is that compared CNN-based methods are mitigated from image classification networks where highly abstract feature representations are learned via consecutive combination of convolution and downsampling operations. However, this may impede semantic segmentation tasks, where detailed spatial information is desired [13]. HRNetV2 is originally designed for high-resolution representation learning where multi-resolution representations are learned in parallel such that spatial detail information can be maintained via high resolution representation branch and highly abstract information can be learned via low-resolution representation learning branch. Thus, HRNetV2 outperforms compared CNN-based semantic segmentation methods on lesion segmentation. Comparing with vanilla HRNetV2 [5], the four variants of ours get better performances on the four lesion classes consistently in terms of mAUPR, mDice and mIoU. Particularly, for MAs whose sizes are very tiny, our M2MRFs are able to obtain at least 1.51% improvement in AUPR. The possible reason is that our M2MRFs bypass the naive uniform subsampling, thus can maintain more informative activations about tiny lesions. For EXs and SEs whose appearances are very similar as they both belong to bright yellow lesions, our M2MRFs are able to gain at least 2.01% and 2.44% improvements in AUPR. The possible reason is that our M2MRFs reassemble features inside large regions such that long-term dependencies are exploited to mutually enhance their features, which further facilitates the subsequent classification. Comparing with two most recent transformer-based methods, i.e. Swin-base [7] and Twins-SVT-B [6], our four variants of M2MRF achieve better on mAUPR and particularly our M2MRF-A outperforms them by a large margin consistently on mAUPR, mIoU and mDice.

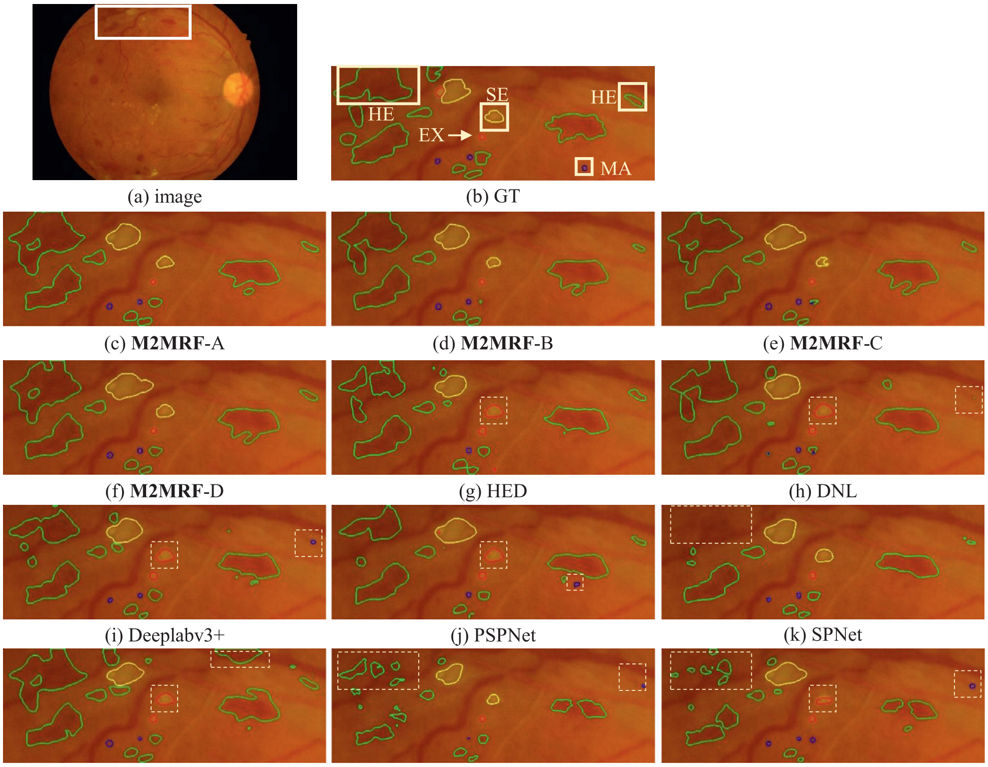
4.2.2. Results on IDRiD [4] dataset

We compare our M2MRF-HRNetV2 with 14 approaches, including the top three methods in ISBI-2018 grand challenge (VRT, PATech, iFLYTEK), three methods specifically designed for lesion segmentation (L-seg [28], Dual-PSP [26], RTNet [3]), six CNN-based methods for general semantic segmentation (HED [27], DNL [23], Deeplabv3+ [20], PSPNet [19], SPNet [38], HRNetV2 [5]) and two transformer-based segmentation methods (Swin-base [7] and Twins-SVT-B [6]). The comparative results on IDRiD testing set are listed in Table 2, in which performances of the first five methods are directly borrowed from original papers and the rest are obtained by fine-tuning models on IDRiD training set.

From Table 2, we have following observations. (1) RTNet [3] outperforms our four variants of M2MRF-HRNetV2 significantly in terms of mAUPR. The possible reason is that RTNet [3] utilises more data to train the lesion segmentation model than ours. Specifically, RTNet [3] utilises two extra datasets, i.e. DRIVE [39] and STARE [40] with pixel-level vessel annotations to boost the performances of lesion segmentation while our M2MRF only leverages the training set of IDRiD [4]. (2) Our four variants of M2MRF-HRNetV2 surpass all the CNN-based and transformer-based methods consistently in terms of mAUPR, mDice and mIoU. (3) Among M2MRF variants, M2MRF-C achieves the best performance, which surpasses the most recent transformer-based segmentation method Swin-base [7] by 2.76%, 2.18% and 2.18% in mAUPR, mIoU and mDice respectively. (4) M2MRF variants contribute significantly to MA segmentation and outperform vanilla HRNetV2 [5] by more than 4% on AUPR, 4% on IoU and 5% on

Table 2
Segmentation Results on IDRiD [4] testing set. M2MRF-A: (One-Step/One-Step), M2MRF-B:(One-Step/Cascade), M2MRF-C:(Cascade/One-Step), M2MRF-D: (Cascade/Cascade). §: These methods use a separate network to train and predict each type of lesion. †: RTNet leverages two extra vessel segmentation datasets during training. All results are averaged over three repetitions. Results marked in **bold** are best ones while those in *italic* are second best ones.

Methods	AUPR					IoU					Dice				
	mAUPR	EX	HE	SE	MA	mIoU	EX	HE	SE	MA	mDice	EX	HE	SE	MA
VRT (1st) § [4]	64.69	71.27	68.04	*69.95*	*49.51*	-	-	-	-	-	-	-	-	-	-
PATech (2nd) §[4]	-	*88.50*	64.90	-	47.70	-	-	-	-	-	-	-	-	-	-
iFLYTEK (3rd) §[4]	64.84	87.41	55.88	65.88	**50.17**	-	-	-	-	-	-	-	-	-	-
L-seg [28]	65.15	79.45	63.74	71.13	46.27	-	-	-	-	-	-	-	-	-	-
Dual-PSPNet [26]	-	77.48	-	-	-	-	60.95	-	-	-	-	75.74	-	-	-
RTNet † [3]	**70.76**	**90.24**	**68.80**	**75.02**	48.97	-	-	-	-	-	-	-	-	-	-
HED [27]	63.94	80.81	66.41	68.09	40.45	46.66	64.74	47.43	50.38	24.07	62.18	78.60	64.33	67.00	38.81
DNL [23]	59.09	75.12	64.04	64.73	32.48	42.28	57.67	44.80	47.03	19.61	57.94	73.15	61.87	63.96	32.78
Deeplabv3+ [20]	63.19	81.93	64.66	63.04	43.14	45.21	66.10	44.90	44.39	25.45	60.90	79.60	61.96	61.48	40.57
PSPNet [19]	58.73	75.21	63.36	63.65	32.71	41.70	57.78	43.71	45.81	19.50	57.38	73.24	60.81	62.83	32.63
SPNet [38]	50.25	64.61	54.11	52.14	30.14	30.30	44.40	28.50	33.58	14.72	45.26	61.45	44.33	49.59	25.66
HRNetV2 [5]	65.01	82.09	65.50	68.68	43.76	47.52	*66.57*	45.56	50.99	26.98	63.14	*79.93*	62.58	67.53	42.49
Swin-base [7]	64.48	81.34	66.57	64.91	45.10	47.76	66.26	48.36	47.54	28.86	63.53	79.71	65.19	64.43	44.79
Twins-SVT-B [6]	63.84	80.09	63.12	68.86	43.27	47.07	64.68	44.91	**51.76**	26.92	62.79	78.56	61.98	**68.19**	42.42
M2MRF-A (Ours)	66.48	82.04	67.80	68.23	47.86	49.07	66.16	*48.87*	50.03	31.23	64.89	79.64	*65.66*	66.68	47.59
M2MRF-B (Ours)	66.00	81.98	67.41	66.68	47.91	48.56	66.07	48.58	48.16	31.42	64.45	79.57	65.39	65.01	47.81
M2MRF-C (Ours)	*67.24*	82.16	*68.69*	69.32	48.80	**49.94**	66.46	**49.72**	*51.43*	**32.13**	**65.71**	79.85	**66.42**	*67.92*	**48.63**
M2MRF-D (Ours)	66.66	82.29	66.94	69.00	48.43	*49.36*	**66.62**	48.04	50.98	*31.81*	*65.15*	**79.97**	64.88	67.50	*48.26*



(a) image (b) GT
(c) **M2MRF**-A (d) **M2MRF**-B (e) **M2MRF**-C
(f) **M2MRF**-D (g) HED (h) DNL
(i) Deeplabv3+ (j) PSPNet (k) SPNet
(l) HRNetV2 (m) Swin-base (n) Twins-SVT-B

Fig. 4. Segmentation results on IDRiD test set [4]. From (a) to (n) are original image, patch with lesion annotations by ophthalmologists, results by our M2MRF-A/B/C/D, HED [27], DNL [23], Deeplabv3+ [20], PSPNet [19], SPNet [38], HRNetV2 [5], Swin-base [7] and Twins-SVT-B [6]. Regions delineated in green, blue, yellow and red are haemorrhage (HE), microaneurysm (MA), soft exudate (SE) and hard exudate (EX) by ophthalmologists or segmentation approaches. Lesions marked with solid golden boxes in (b) are challenging lesions. Four variants of HRNetV2 [5] equipped with our M2MRFs are almost able to correctly segment those challenging lesions while compared methods are prone to encounter either miss or wrong identifications which are marked with dotted golden boxes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Dice, which indicates that our M2MRF benefits the segmentation of small lesions.

We also make qualitative comparisons with state-of-the-art methods and visualise their results in Fig. 4. We can see that the four variants of M2MRFs are powerful in recognising lesions. For example, they can correctly recognise the soft exudate (SE) marked out by golden box in (b) although SE and hard exudate (EX) are confusing due to their similar appearances. On the contrary, the baseline method, i.e., HRNetV2 [5], is confused and wrongly identifies SE as EX. The possible reason is that the four variants of our M2MRF can capture the dependencies between the SE in golden box in (b) and the left top SE to enhance the features of the SE in golden box while the vanilla HRNetV2 fails. For the rest compared methods, only SPNet [38] and Swin-base [6] can correctly

Table 3
Generalisation study of different segmentation methods on DDR [8] and IDRiD [4] datasets. All models are trained on DDR [8] training set and their performances on IDRiD [4] testing set are reported. Results marked in **bold** are best ones while those in *italic* are second best ones.

Methods	AUPR				
	mAUPR	EX	HE	SE	MA
L-seg [28]	39.88	65.01	44.05	30.59	19.86
RTNet [3]	42.05	67.99	45.04	34.01	21.14
HRNetV2 [5]	56.33	68.71	60.12	54.83	41.66
Swin-base [7]	56.90	71.58	**62.72**	56.98	36.35
Twins-SVT-B [6]	56.26	71.84	58.04	**58.24**	36.93
M2MRF-A (Ours)	58.32	74.04	59.97	55.20	*44.09*
M2MRF-B (Ours)	58.41	**74.69**	58.81	56.13	44.03
M2MRF-C (Ours)	**59.30**	73.94	*61.01*	*57.61*	**44.64**
M2MRF-D (Ours)	*58.86*	*74.18*	60.14	57.48	43.64

recognise the SE in golden box in (b). Similarly, correctly recognising haemorrhage (HE) and microaneurysm (MA) is also challenging as both of them belong to dark red lesions and always exhibit similar appearances. Additionally, they are also prone to be confused by the vessels. As is shown in Fig. 4(l), the baseline method HRNetV2 [5] wrongly recognises the vessels in dot golden box as HE. For the HE marked out by golden box at rightmost in (b), it is wrongly recognised as MA by Deeplabv3+ [20] and Twins-SVT-B [6] and missed by DNL [23] and Swin-base [7]. For the large HE at leftmost marked out by box in (b), SPNet [38] completely fails to recognise it (see the dot golden box in (k)), and Swin-base [7] and Twins-SVT-B [6] segment it poorly. On the contrary, our M2MRFs are able to successfully segment these HEs.

4.2.3. Generalisation study

Following [3], we train lesion segmentation models with DDR [8] training set and test their performances on IDRiD [4] testing set to validate the model generalisation. Comparison results with five methods are listed in Table 3, in which results of L-seg [28] and RTNet [3] are obtained from Huang et al. [3] while others are reproduced by us. We can see that our M2MRF variants achieve better mAUPR performance comparing to lesion segmentation methods L-seg [28] and RTNet [3] as well the two recent transformer-based methods, i.e., Swin-Base [7] and Twins-SVT-B [6] by a large margin.

4.3. Comparison to state-of-the-art operators of feature reassembly

4.3.1. Results and analysis on DDR [8] dataset

Here we make comparisons to alternative RF operators to verify the effectiveness of our proposed M2MRF on DDR [8] testing set. In what follows, we first compare our M2MRF with RF operators that can/must be used as pairs, then we investigate which contribute more on lesion segmentation for RF operators that can be used as both downsampling and upsampling operators.

Comparisons to paired RF operators. We compare our four M2MRF variants with eight paired RF operators, i.e. StrideConv/Bilinear, MaxPool/Bilinear, MaxPool/Unpooling, StrideConv/Deconv, StrideConv/LIP [14], CARAFE++ [16], IndexNet [17] and A^{2}U [18]. Among them, StrideConv/Bilinear is the default setting in vanilla HRNetV2 [5], in which the kernel size and stride associated to StrideConv are set to 3 × 3 and 2, respectively. MaxPool/Unpooling and IndexNet [17] have to utilise the indices generated during downsampling for upsampling. For MaxPool, 3 × 3 max-pooling kernel with stride of 2 is adopted. For LIP [14], CARAFE++ [16], IndexNet [17] and A^{2}U [18], suggested settings by original papers are adopted. We replace StrideConv/Bilinear in vanilla HRNetV2 [5] with these alternative paired RF operators and retrain models on training set of DDR [8]. Table 4 reports quantitative results.

As listed, among four variants of our M2MRF, M2MRF-A achieves best in mAUPR, mDice and mIoU. Comparing with vanilla HRNetV2 [5], our M2MRF-A improves the mAUPR by a large margin from 45.21% to 49.94% and the mIoU from 28.84% to 31.16%. Comparing with existing feature reassembly operators, our M2MRF-A achieves best in both mAUPR and mIoU and the rest three variants achieve better mAUPR and competitive mIoU. Specifically, comparing with the recent learning-based feature reassembly operators IndexNet [17] and CARAFE++ [16], our M2MRF-A and M2MRF-B achieve better performance in mAUPR and mIoU. Comparing with A^{2}U [18], our four variants of M2MRF outperform it on both mAPUR and mIoU.

In terms of the number of parameters and inference speed, the four variants of our M2MRF are inferior to both the baseline and two rule-based RFs, i.e., MaxPool/Bilinear and MaxPool/Unpooling. Particularly, comparing with the baseline, our M2MRF-A introduces extra 4.08M parameters and slows down the inference speed from 11.31FPS to 9.21FPS. On the contrary, as MaxPool/Bilinear and MaxPool/Unpooling are parameter-free, their number of parameters are less than the baseline. Also they improve the inference speed from 11.31FPS to 11.88FPS and 11.62FPS respectively.

Fig. 5 shows visualized segmentation results. We can see that four variants of our M2MRF make less mistakes on small lesions than those compared RF operators, which may further demonstrate that our M2MRFs preserve more discriminative details about small lesions. We note that, for the smaller SE in Fig. 5(b), both our M2MRFs and the compared methods fail to segment it. The possible reason is that its contrast to background is too subtle to segment it.

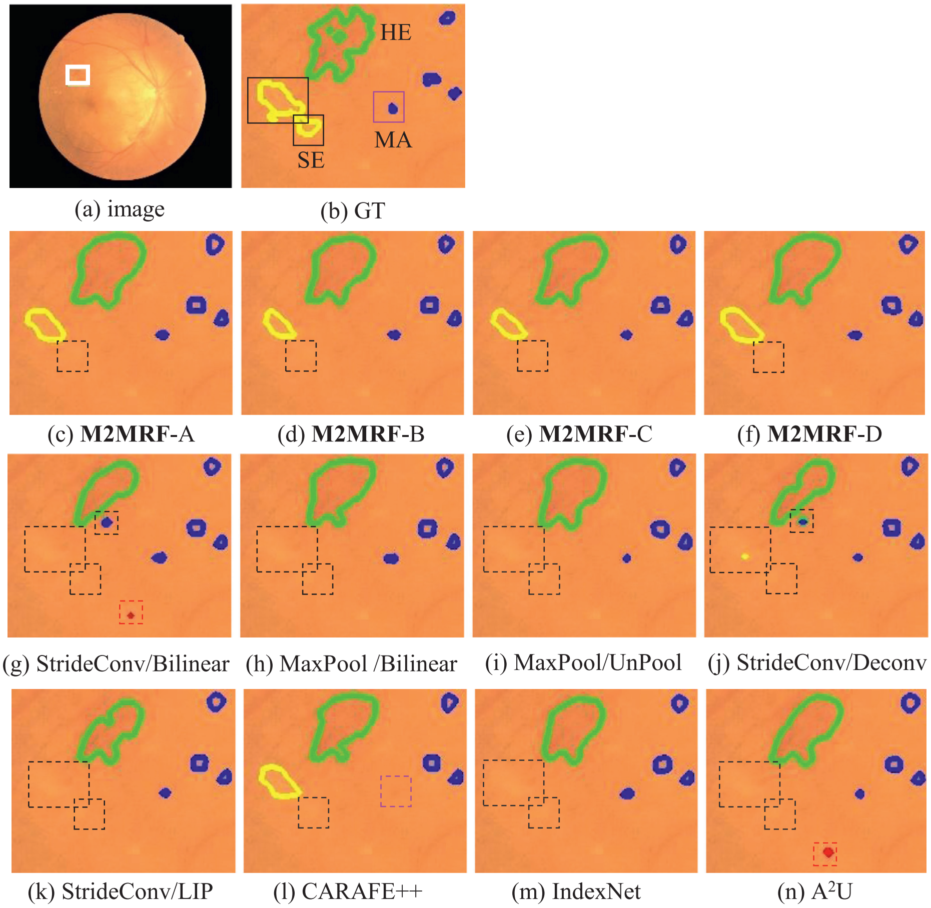
Which contributes more: downsampling, upsampling or in combination? Recent two novel RF operators CARAFE++ [16] and A^{2}U [18] and our proposed M2MRFs can be used alone or in combination for downsampling and upsampling. Here we further conduct experiments to investigate the effectiveness of those RF operators used as alone and in combination. We list the results in Table 5. We can see that: (1) CARAFE++ [16] as downsampling operator achieves inferior performances on mDice and mIoU to the baseline. For A^{2}U [18] as downsampling operator, performance increments on mDice and mIoU are 0.54% and 0.50%, respectively. Both our one-step and cascade M2MRF gain performance improvement on mDice by 1.73% and 1.04%, on mIoU by 1.52% and 0.87%, respectively. The possible reason is that our M2MRFs bypass uniform subsample operation so that they are able to preserve more spatial details about small lesions than CARAFE++ [16], A^{2}U [18] as well as baseline. (2) CARAFE++ [16] and A^{2}U [18] achieve better performances when replacing upsampling operators than downsampling operators while ours are just the opposite. (3) When using them in combination, our M2MRFs achieve best performances consistently on mAUPR, mDice as well as mIoU. Conversely, both CARAFE++ [16] and A^{2}U [18] achieve inferior performances when using in combination than only using for upsampling in terms of mDice and mIoU.

4.3.2. Comparisons to state-of-the-art feature reassembly operators on IDRiD [4] dataset

To further validate the effectiveness of our proposed M2MRF, we also conduct experiments on IDRiD [4] and make comparisons with existing eight pairs of feature reassembly operators. Table 6 reports the results. It shows that M2MRF-C, i.e., cascade M2MRF for downsampling and one-step M2MRF for upsampling ranks first while our M2MRF-D, i.e., cascade M2MRF for both downsampling and upsampling ranks second among listed

Table 4
Comparing with different paired RF operators on DDR [8] testing set. The first row is the baseline, i.e. the default setting in Vanilla HRNetV2 [5]. All results are averaged over three repetitions. Results marked in **bold** are best ones while those in *italic* are second best ones.

Paired RFs	mAUPR	mDice	mIoU	Param(M)	FPS
StrideConv / Bilinear (baseline)	45.21	43.95	28.84	*65.85*	11.31
MaxPool [31] / Bilinear	45.97	44.28	29.17	**59.45**	**11.88**
MaxPool [31] / Unpooling [41]	48.81	*46.17*	*30.73*	**59.45**	*11.62*
StrideConv / Deconv [32]	46.24	44.59	29.37	73.12	10.86
StrideConv / LIP [14]	43.14	40.68	26.29	75.44	6.80
CARAFE+ [16]	47.64	44.80	29.54	72.57	8.69
IndexNet [17]	48.06	45.64	30.28	70.33	10.75
A^{2}U [18]	45.89	44.44	29.27	66.51	3.97
M2MRF-A (ours)	**49.94**	**46.60**	**31.16**	69.93	9.21
M2MRF-B (ours)	*49.42*	45.77	30.41	67.15	8.54
M2MRF-C (ours)	48.94	45.40	30.09	70.30	8.43
M2MRF-D (ours)	49.25	45.57	30.27	67.52	8.01



(a) image (b) GT

(c) **M2MRF**-A (d) **M2MRF**-B (e) **M2MRF**-C (f) **M2MRF**-D

(g) StrideConv/Bilinear (h) MaxPool /Bilinear (i) MaxPool/UnPool (j) StrideConv/Deconv

(k) StrideConv/LIP (l) CARAFE++ (m) IndexNet (n) A^{2}U

Fig. 5. Visualization results on DDR test set [8] by variants of HRNetV2 with paired feature reassembly operators: M2MRF-A, M2MRF-B, M2MRF-C, M2MRF-D, StrideConv/bilinear, MaxPool/Bilinear, MaxPool/UnPool [41], StrideConv/Deconv, StrideConv/LIP [14], CARAFE++ [16], IndexNet [17] and A^{2}U [18]. Regions delineated in green, blue, red and yellow are HE, MA, EX and SE by ophthalmologists or segmentation approaches. In GT map (b), challenging lesions are marked with solid boxes. In visualised segmentation result maps (c-n), wrong and miss identifications are marked with dotted boxes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

competitors. Particularly, our M2MRF-C surpasses the baseline consistently in terms of mAUPR (67.24% *vs.* 65.01%), mDice (65.71% *vs.* 63.14%) and mIoU (49.94% *vs.* 47.52%) by a large margin. Interestingly, MaxPool/Unpooling, as rule-based operators, achieves better than previous learning-based operators. The possible reason is that the location information preserved when performing MaxPool operator helps Unpooling operator to recover spatial details about lesions, thus further facilitates to the segmentation.

4.4. Extension to other backbones

To further validate the effectiveness of our M2MRF, we integrate it into different backbones for lesion segmentation. They are ResNet50 [42], SKNet [43], HRNet32 [5] as well as HRNet48 [5]. For ResNet50 [42] as the segmentation backbone, UperNet [44] is usually attached as the segmentation head. We directly replace the downsampling operators, i.e., strided convolution layers, in ResNet50 [42] and upsampling operators, i.e., bilinear interpolation

Table 5
Comparing with different feature reassembly operators for downsampling, upsampling, and both on DDR [8] testing set. The first row is the baseline i.e. vanilla HRNetV2 [5]. The best results of each group are highlighted in boldface. All results are averaged over three repetitions.

RF operators	Downsample	Upsample	mAUPR	Δ_{mAUPR}	mDice	Δ_{mDice}	mIoU	Δ_{mIoU}
StrideConv (baseline)	✓		45.21	0	43.95	0	28.84	0
Bilinear (baseline)		✓						
CARAFE+[16]	✓		46.27	1.06	43.51	-0.44	28.51	-0.33
		✓	46.68	1.47	**45.32**	**1.38**	**29.97**	**1.13**
	✓	✓	**47.64**	**2.43**	44.80	0.85	29.54	0.70
A^{2}U [18]	✓		46.74	1.53	44.49	0.54	29.34	0.50
		✓	**47.71**	**2.50**	**46.17**	**2.22**	**30.71**	**1.87**
	✓	✓	45.89	0.68	44.44	0.49	29.27	0.43
one-step M2MRF	✓		49.41	4.20	45.68	1.73	30.36	1.52
		✓	45.44	0.23	44.03	0.08	28.88	0.04
	✓	✓	**49.94**	**4.73**	**46.60**	**2.65**	**31.16**	**2.32**
cascade M2MRF	✓		48.92	3.71	44.99	1.04	29.71	0.87
		✓	45.37	0.16	44.46	0.51	29.27	0.44
	✓	✓	**49.25**	**4.04**	**45.57**	**1.65**	**30.27**	**1.43**

Table 6
Comparing with different feature reassembly operators on IDRiD [4] testing set. The first row is the baseline i.e. vanilla HRNetV2 [5]. All results are averaged over three repetitions. Results marked in **bold** are best ones while those in *italic* are second best ones.

Paired RF operators	mAUPR	mDice	mIoU
StrideConv / Bilinear (baseline)	65.01	63.14	47.52
MaxPool [31] / Bilinear	66.24	65.01	49.28
LIP [14] / Bilinear	63.04	61.50	45.78
StrideConv / Deconv [32]	64.64	62.86	47.23
MaxPool [31] / Unpooling [41]	66.17	64.98	49.22
CARAFE+ [16]	66.35	64.67	48.91
IndexNet [17]	65.77	64.30	48.74
A^{2}U [18]	64.93	63.23	47.52
M2MRF-A (ours)	66.48	64.89	49.07
M2MRF-B (ours)	66.00	64.45	48.56
M2MRF-C (ours)	**67.24**	**65.71**	**49.94**
M2MRF-D (ours)	*66.66*	*65.15*	*49.36*

Table 7
Results of different backbones with our M2MRF for downsampling and upsampling on DDR [8]. Results are averaged over three repetitions.

	mAUPR	mDice	mIoU
ResNet50	41.97	40.95	26.45
w/M2MRF	**47.39**	**43.32**	**28.37**
SKNet	41.48	39.62	25.56
w/M2MRF	**47.94**	**44.31**	**29.25**
HRNet32	45.40	44.03	28.94
w/M2MRF	**48.17**	**44.31**	**29.18**
HRNet48	45.21	43.95	28.84
w/M2MRF	**48.94**	**45.40**	**30.09**

Table 8
Results of different backbones with our M2MRF for downsampling and upsampling on IDRiD [4]. Results are averaged over three repetitions.

	mAUPR	mDice	mIoU
ResNet50	64.10	61.95	46.18
w/M2MRF	**66.78**	**65.11**	**49.25**
SKNet	66.08	64.41	49.05
w/M2MRF	**67.18**	**65.86**	**50.23**
HRNet32	64.74	62.98	47.30
w/M2MRF	**67.24**	**65.45**	**49.62**
HRNet48	65.01	63.14	47.52
w/M2MRF	**67.24**	**65.71**	**49.94**

Table 9
M2MRF(Cascade/One-Step) with different hyper-parameter settings on DDR validation set [8]. Results are averaged over three repetitions.

S_h, S_w	r	α	mAUPR	mDice	mIoU
4	4	64	60.93	59.77	43.33
8	**4**	**64**	**61.49**	**60.26**	**43.80**
16	4	64	60.69	59.42	42.93
8	2	64	61.13	59.89	43.45
8	**4**	**64**	**61.49**	**60.26**	**43.80**
8	8	64	61.43	60.18	43.78
8	4	32	60.34	59.24	42.71
8	**4**	**64**	**61.49**	**60.26**	**43.80**
8	4	128	61.31	60.06	43.63

layers, in UperNet [44], with our proposed M2MRFs for segmentation. For SKNet [43] where 3×3 convolution layer with stride of 2 and 3×3 dilation convolution with stride of 2 and dilation rate of 2 are performed on feature maps for downsampling and then the proposed selective module is performed to select the two types of the downsampled feature maps, we replace the two downsampling layers with our M2MRFs with patch size of 4×4 and 8×8 respectively. For HRNet32 [5] and HRNet48 [5], we replace the strided convolution layers [42] and bilinear interpolation layers with our M2MRFs. Results on DDR [8] and IDRiD [4] are listed in Table 7 and Table 8 respectively. We can see that equipped with our M2MRF, the performances on lesion segmentation are improved consistently.

4.5. Ablation study for M2MRF

We conduct ablation studies to investigate the influences of hyper-parameters in our M2MRF on lesion segmentation. Totally, there are four hyper-parameters in our M2MRF, i.e. the patch size S_h and S_w, r in channel compressor, and α in Eq. (6). For S_h and S_w, we directly let them equal and conduct experiments with setting $\{4, 8, 16\}$. For r and α, we conduct experiments with setting r to $\{2, 4, 8\}$ and α to $\{32, 64, 128\}$. We simply replace the strided convolution layer in HRNetV2 [5] with our cascade M2MRF and the bilinear interpolation layer in HRNetV2 [5] with one-step M2MRF. As DDR [8] provides validation set, we conduct ablation experiments on it and the results are listed in Table 9, from which we can see that $S_h = S_w = 8$, $r = 4$ and $\alpha = 64$ yield best performances. It is worth mentioning that, except for extra illustration, we directly use

$S_h = S_w = 8$, $r = 4$, and $\alpha = 64$ as the default setting for all above experiments on DDR [8] and IDRiD [4].

5. Conclusion and future work

In this paper, we unify the feature downsampling and upsampling in one framework and propose a simple feature reassembly operator named M2MRF for small lesion segmentation in retinal fundus images. It simultaneously reassembles multiple features in a large region to multiple output features so that rich long-range spatial dependencies are exploited and activations about small lesions are enhanced. Extensive experiments are conducted to validate the effectiveness of our M2MRF on two public lesion segmentation datasets, i.e. DDR [8] and IDRiD [4]. On one hand, our M2MRF significantly improves the sate-of-the-art CNN-based method, i.e., the vanilla HRNetV2 [5] by 4.73% and 2.23% in mAUPR on DDR and IDRiD respectively while only marginal extra parameters and inference time are increased. On the other hand, our M2MRF also outperforms the two sate-of-the-art transformer-based segmentation methods, i.e., Swin-base [7] by 3.22% and 2.76% in mAUPR on DDR and IDRiD respectively, and Twins-SVT-B [6] by 3.83% and 3.40% in mAUPR on DDR and IDRiD respectively. Moreover, our M2MRFs also show more powerful generalisation ability which improves the baseline, i.e. vanilla HRNetV2 [5] from 56.33% to 59.30% in mAUPR when directly transferring models trained with DDR [8] to IDRiD [4]. Nevertheless, for tiny lesions whose contrasts to background are very subtle, how to extract discriminative information about them is still a challenging problem in lesion segmentation task.

In future, we will explore wider applications of our M2MRF on diabetic retinopathy grading and lesion segmentation in other medical images such as CTs and OCTs. Besides, we also intend to exploit the mutual dependencies between tasks of lesion segmentation and diabetic retinopathy grading and develop deep multi-task learning framework for joint lesion segmentation and diabetic retinopathy grading.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the research are openly available from third party https://doi.org/10.21227/H25W98 and https://doi.org/10.1016/j.ins.2019.06.011. The code is publicly available at github.com.

Acknowledgments

This work is supported in part by the High Performance Computing Center of Central South University. Qing Liu is partially supported by the National Natural Science Foundation of China (NSFC No. 62006249), Changsha Municipal Natural Science Foundation (kq2014135) and Hunan Provincial Natural Science Foundation of China (2021JJ40788). Wei Ke is partially supported by the National Natural Science Foundation of China (NSFC No.62006182). Yixiong Liang is partially supported by the National Key R&D Program of China (No.2021YFF1201202) and the Hunan Provincial Science and Technology Innovation Leading Plan (No.2020GK2019).

References

[1] M.U. Akram, S. Khalid, S.A. Khan, Identification and classification of microaneurysms for early detection of diabetic retinopathy, Pattern Recognit. 46 (1) (2013) 107–116.

[2] T.Y. Wong, J. Sun, R. Kawasaki, P. Ruamviboonsuk, N. Gupta, V.C. Lansingh, M. Maia, W. Mathenge, S. Moreker, M.M.K. Muqit, S. Resnikoff, J. Verdaguer, P. Zhao, F. Ferris, L.P. Aiello, H.R. Taylor, Guidelines on diabetic eye care: the international council of ophthalmology recommendations for screening, follow-up, referral, and treatment based on resource settings, Ophthalmology 125 (10) (2018) 1608–1622.

[3] S. Huang, J. Li, Y. Xiao, N. Shen, T. Xu, RTNet: relation transformer network for diabetic retinopathy multi-lesion segmentation, IEEE Trans. Med. Imaging (TMI) 41 (6) (2022) 1596–1607.

[4] P. Porwal, S. Pachade, M. Kokare, G. Deshmukh, J. Son, W. Bae, L. Liu, J. Wang, X. Liu, L. Gao, et al., IDRiD: diabetic retinopathy–segmentation and grading challenge, Med. Image Anal. 59 (2020) 101561.

[5] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, et al., Deep high-resolution representation learning for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 43 (10) (2020) 3349–3364.

[6] X. Chu, Z. Tian, Y. Wang, B. Zhang, H. Ren, X. Wei, H. Xia, C. Shen, Twins: revisiting the design of spatial attention in vision transformers, Adv. Neural Inf. Process. Syst. (NeurIPS), Vol. 34, 2021.

[7] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: hierarchical vision transformer using shifted windows, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2021, pp. 10012–10022.

[8] T. Li, Y. Gao, K. Wang, S. Guo, H. Liu, H. Kang, Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening, Inf. Sci. 501 (2019) 511–522.

[9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 3213–3223.

[10] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2015, pp. 3431–3440.

[11] C. Beeche, J.P. Singh, J.K. Leader, N.S. Gezer, A.P. Oruwari, K.K. Dansingani, J. Chhablani, J. Pu, Super U-Net: a modularized generalizable architecture, Pattern Recognit. 128 (2022) 108669.

[12] J.T. Springenberg, A. Dosovitskiy, T. Brox, M. Riedmiller, Striving for simplicity: the all convolutional net, in: Int. Conf. Learn. Represent. (ICLR) workshop, 2015.

[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 40 (4) (2017) 834–848.

[14] Z. Gao, L. Wang, G. Wu, Lip: local importance-based pooling, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 3355–3364.

[15] M.D. Zeiler, D. Krishnan, G.W. Taylor, R. Fergus, Deconvolutional networks, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), IEEE, 2010, pp. 2528–2535.

[16] J. Wang, K. Chen, R. Xu, Z. Liu, C.C. Loy, D. Lin, Carafe++: unified content-aware reassembly of features, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 44 (9) (2022) 4674–4687.

[17] H. Lu, Y. Dai, C. Shen, S. Xu, Index networks, IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) 44 (1) (2020) 242–255.

[18] Y. Dai, H. Lu, C. Shen, Learning affinity-aware upsampling for deep image matting, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2021, pp. 6841–6850.

[19] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 2881–2890.

[20] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 801–818.

[21] K. Wang, X. Zhang, X. Zhang, Y. Lu, S. Huang, D. Yang, EANet: iterative edge attention network for medical image segmentation, Pattern Recognit. 127 (2022) 108636.

[22] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 7794–7803.

[23] M. Yin, Z. Yao, Y. Cao, X. Li, Z. Zhang, S. Lin, H. Hu, Disentangled non-local neural networks, in: Eur. Conf. Comput. Vis. (ECCV), Springer, 2020, pp. 191–207.

[24] B. Zhang, X. Wu, J. You, Q. Li, F. Karray, Detection of microaneurysms using multi-scale correlation coefficients, Pattern Recognit. 43 (6) (2010) 2237–2248.

[25] Q. Liu, B. Zou, J. Chen, W. Ke, K. Yue, Z. Chen, G. Zhao, A location-to-segmentation strategy for automatic exudate segmentation in colour retinal fundus images, Comput. Med. Imaging Graph. 55 (2017) 78–86.

[26] Q. Liu, H. Liu, Y. Zhao, Y. Liang, Dual-branch network with dual-sampling modulated dice loss for hard exudate segmentation in colour fundus images, IEEE J. Biomed. Health Inform. 26 (3) (2022) 1091–1102.

[27] S. Xie, Z. Tu, Holistically-nested edge detection, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2015, pp. 1395–1403.

[28] S. Guo, T. Li, H. Kang, N. Li, Y. Zhang, K. Wang, L-Seg: An end-to-end unified framework for multi-lesion segmentation of fundus images, Neurocomputing 349 (2019) 52–63.

[29] Y. Zhou, X. He, L. Huang, L. Liu, F. Zhu, S. Cui, L. Shao, Collaborative learning of semi-supervised segmentation and classification for medical images, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 2079–2088.

[30] X. Wang, M. Xu, J. Zhang, L. Jiang, L. Li, Deep multi-task learning for diabetic retinopathy grading in fundus images, in: Proc. AAAI Conf. Artif. Intell. (AAAI), Vol. 35, 2021, pp. 2826–2834.

[31] Y. LeCun, B.E. Boser, J.S. Denker, D. Henderson, R.E. Howard, W.E. Hubbard, L.D. Jackel, Handwritten digit recognition with a back-propagation network, in: Adv. Neural Inf. Process. Syst. (NeurIPS), 1990, pp. 396–404.

[32] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Eur. Conf. Comput. Vis. (ECCV), Springer, 2014, pp. 818–833.

[33] J. Wang, K. Chen, R. Xu, Z. Liu, C.C. Loy, D. Lin, CARAFE: content-aware reassembly of features, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 3007–3016.

[34] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2017, pp. 764–773.

[35] F. Milletari, N. Navab, S. Ahmadi, V-Net: fully convolutional neural networks for volumetric medical image segmentation, in: 2016 Fourth Int. Conf. 3D Vis. (3DV), 2016, pp. 565–571.

[36] O. Alpar, R. Dolezal, P. Ryska, O. Krejcar, Nakagami-fuzzy imaging framework for precise lesion segmentation in MRI, Pattern Recognit. 128 (2022) 108675.

[37] M. Contributors, MMSegmentation: OpenMMLab semantic segmentation toolbox and benchmark, 2020, (https://github.com/open-mmlab/mmsegmentation).

[38] Q. Hou, L. Zhang, M.-M. Cheng, J. Feng, Strip pooling: rethinking spatial pooling for scene parsing, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 4003–4012.

[39] J. Staal, M.D. Abràmoff, M. Niemeijer, M.A. Viergever, B. Van Ginneken, Ridge-based vessel segmentation in color images of the retina, IEEE Trans. Med. Imaging (TMI) 23 (4) (2004) 501–509.

[40] A.D. Hoover, V. Kouznetsova, M. Goldbaum, Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response, IEEE Trans. Med. Imaging (TMI) 19 (3) (2000) 203–210.

[41] M.D. Zeiler, G.W. Taylor, R. Fergus, Adaptive deconvolutional networks for mid and high level feature learning, in: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), IEEE, 2011, pp. 2018–2025.

[42] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.

[43] X. Li, W. Wang, X. Hu, J. Yang, Selective kernel networks, in: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 510–519.

[44] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, J. Sun, Unified perceptual parsing for scene understanding, in: Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 418–434.

Qing Liu received the PhD degree in Computer Science from Central South University in 2017. Currently, she is a lecturer with Central South University. Her research interests include medical image analysis and computer vision and has published 25+ papers in peer-reviewed journals and conferences including IEEE-TIP, IEEE-JBHI and ICASSP.

Haotian Liu received the Bachler degree in Computer Science from Central South University in 2020. Currently, he is a master student with Central South University. His research interest is medical image analysis.

Wei Ke received the PhD degree from the University of Chinese Academy of Sciences in 2018. He is currently an Associate Professor with Xi'an Jiaotong University. He has published about 20 papers in refereed conferences and journals including CVPR and ECCV. His research interests include computer vision and deep learning.

Yixiong Liang is currently a Professor of Computer Science with Central South University. He received the PhD degree from Chongqing University in 2005. His research interests include computer vision and machine learning. He has published 50+ papers in peer-reviewed journals and conferences including ICCV, Pattern Recognition and IEEE-JBHI.