Context-aware Spatio-recurrent Curvilinear Structure Segmentation

Feigege Wang[1],Yue Gu[1],Wenxi Liu[1],Yuanlong Yu[1],Shengfeng He[2],Jia Pan[3]
[1]*College of Mathematics and Computer Science, Fuzhou University*
[2]*School of Computer Science and Engineering, South China University of Technology*
[3]*Department of Computer Science, The University of Hong Kong*

Abstract

Curvilinear structures are frequently observed in various images in different forms, such as blood vessels or neuronal boundaries in biomedical images. In this paper, we propose a novel curvilinear structure segmentation approach using context-aware spatio-recurrent networks. Instead of directly segmenting the whole image or densely segmenting fixed-sized local patches, our method recurrently samples patches with varied scales from the target image with learned policy and processes them locally, which is similar to the behavior of changing retinal fixations in the human visual system and it is beneficial for capturing the multi-scale or hierarchical modality of the complex curvilinear structures. In specific, the policy of choosing local patches is attentively learned based on the contextual information of the image and the historical sampling experience. In this way, with more patches sampled and refined, the segmentation of the whole image can be progressively improved. To validate our approach, comparison experiments on different types of image data are conducted and the sampling procedures for exemplar images are illustrated. We demonstrate that our method achieves the state-of-the-art performance in public datasets.

1. Introduction

Due to the advances of deep learning techniques, image segmentation has been rapidly developed in recent years, with its main focus on semantically segmenting everyday objects from natural images. On the other hand, there are many other segmentation tasks about extracting objects with special shapes, which requires prior domain knowledge about the target objects. Some typical examples include the cell or organ segmentation from biomedical images [7, 53] and the aerial images semantic segmentation [32, 21, 29]. In the microscopic image data, objects with curvilinear structures, such as blood vessels, are the targets for segmentation, which have not attracted enough attention in previous works.

Prior methods for addressing the task of the curvilinear structure segmentation usually involved hand-crafted features [49], which cannot robustly handle a wide variety of complex curvilinear structures and could be sensitive to image noises. The recent success of Deep Convolutional Neural Networks (DCNN) has greatly improved the performance of image segmentation, and the standard DCNNs have also been applied to segmentation tasks involving curvilinear structures, such as blood vessel [24, 31] and neurons [4, 42, 35, 10].However, the fixed receptive fields of convolutional layers in DCNN models have difficulty to identify the curvilinear structures in biomedical images, because the curvilinear structures are complex, entangled, and multi-scaled but may take up a small portion in the local patches of the image. In addition, existing segmentation methods [39, 7] need to randomly crop small patches from the image using a fixed window size, in order to collect adequate amount of training samples. In the testing phase, these approaches have to sample patches with the same window size as in training and to produce the final segmentation result using the overlapped tiling strategy. Such a random sampling process does not take into account the context dependency in the image, which is important for identifying a complex network of multi-scale curvilinear

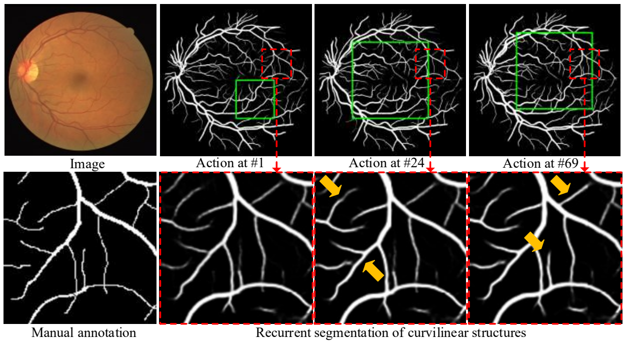


Figure 1. Our proposed approach sequentially and adaptively samples local patches from the input image, e.g. an retinal image as shown, and then the segmentation in these patches are enhanced. The green boxes represent the patches sampled at different time-steps (with 1^{st}, 24^{th}, and 69^{th} as examples). The second row shows the zoomed-in results for the first row's segmentation results within red regions and compares with a manual annotation as reference. The golden arrows highlight the major changes in the zoomed-in region. Whenever the red and green boxes overlap, the segmentation within red regions will be refined by taking into account the image contexts. The segmentation of the entire image will be progressively improved with more patches being sampled and refined.

structures.

To handle the above limitations, we propose a novel curvilinear structure segmentation approach, which is inspired by the human visual system. According to [23], it is widely believed that biological vision systems have a sequential process with changing retinal fixations that gradually accumulate evidence of certainty. Intuitively, to examine an object or an image, we may first glance over it and then watch attentively on the local regions to observe the details. For the task of detecting curvilinear structures that often exhibit multi-scale or hierarchical modality, such kind of visual models can be applied. Hence, we propose a spatio-recurrent segmentation approach that sequentially processes patches sampled from the target image. As shown in Fig. 1, the locations and sizes of the selected patches are adaptively determined based on the image context and the segmentation of these patches are locally refined, which is similar to the behavior of changing retinal fixations. By performing the sampling and the local segmentation recursively, the multi-scale curvilinear structures in the target image will be gradually extracted and refined, making our method attractive for tasks like segmenting blood vessels or cell boundaries from images.

Specifically, we adopt the Actor-Critic framework [22] to learn the sampling policy that automatically decides where to crop the patches and how large the patches shall be. To incorporate the context dependency cues, the sampling policy not only considers the contextual information based on the holistic feature of the image, but also considers the previous segmentation results via Long Short-Term Memory (LSTM). Hence, in each step of our proposed algorithm, the policy model infers the optimal action for sampling the patch given the latest segmentation mask. We also present an attentive feature extraction module to guide the policy to focus in the region where the segmentation decision is uncertain, in order to drive the sampling process more effectively by following the topology of the curvilinear structure networks. Finally, for validation and evaluation, we experiment our approach on datasets involving two types of curvilinear structures, including two retinal image datasets (i.e., DRIVE [47] and STARE [19]) and the Electron Microscopy dataset [2]. We also illustrate the sampling procedures on exemplar images.

The contribution of this paper is threefold:

- We propose a novel spatio-recurrent segmentation approach that sequentially segments proper-sized local patches and progressively refines the curvilinear structures of a target image.
- The patch sampling policy is learned by incorporating the spatial attention, the contextual dependency cues of the image, and the historical sampling experience.
- It accomplishes state-of-the-art performance in the retinal image datasets and the Electron Microscopy dataset.

2. Related Works

In this section, we survey related literatures on semantic segmentation, curvilinear structure segmentation, and deep reinforcement learning based vision works.

Semantic segmentation is one of the most challenging tasks in computer vision, which attempts to predict pixel-level semantic labels of a given image or video frame. Inspired by the recent advance of Fully Convolutional Networks (FCN) [30], several techniques have been proposed which incorporate multi-scale feature ensemble and context information preservation, such as Dilated Convolution [52], DeepLab [8], RefineNet [26], PSPNet [56], and RAN [55]. The DCNN-based image segmentation techniques have also been used for segmenting special types of biological structures, including the liver [27, 28], the lungs [11], the kidney [25], the cells [7, 15], the pancreas [53], or the hippocampus [12, 48]. In contrast, our work focuses on extracting curvilinear structures from biomedical images.

Curvilinear structure segmentation has been studied for decades [45, 38, 3, 36], because these structures are ubiquitous in biological images, with typical examples as blood vessels, bronchial networks, and dendritic arbors. The segmentation of these structures is important for the medical analysis, but the accurate automated delineation of all curvilinear structures from an image is still challenging. There are literatures related to this problem in the community of biomedical image analysis [39, 24, 31, 49] and remote-sensing image analysis [33, 1, 57, 13, 50]. Among these works, the blood vessel segmentation [3, 36, 31] and the neuronal boundary detection [4, 42, 10] are extensively studied. In recent years, DCNNs have been used for curvilinear structure segmentation as well [39, 31, 42]. In particular, U-Net [39] is presented as an encoder-decoder architecture with skip connections, which demonstrates excellent performance for detecting the boundaries of neurons. Maninis et al. [31] fuse multi-scale deep features to extract blood vessels from retina images. In [42], a multi-stage fully convolutional neural network is presented for boundary detection. Mosinska et al. [35] propose a pixel-wise topology loss and a recursive refinement algorithm to perform delineation. Compared with prior methods that performs segmentation with fixed receptive fields, our approach recurrently samples and segments the proper-sized local patches incorporating with the contextual information of the image.

Deep reinforcement learning (DRL) is first introduced in [34], which applies a deep neural network as a function approximator to estimate the action-value function for reinforcement learning. Recently, there are some works attempting to apply DRL to computer vision tasks [5,

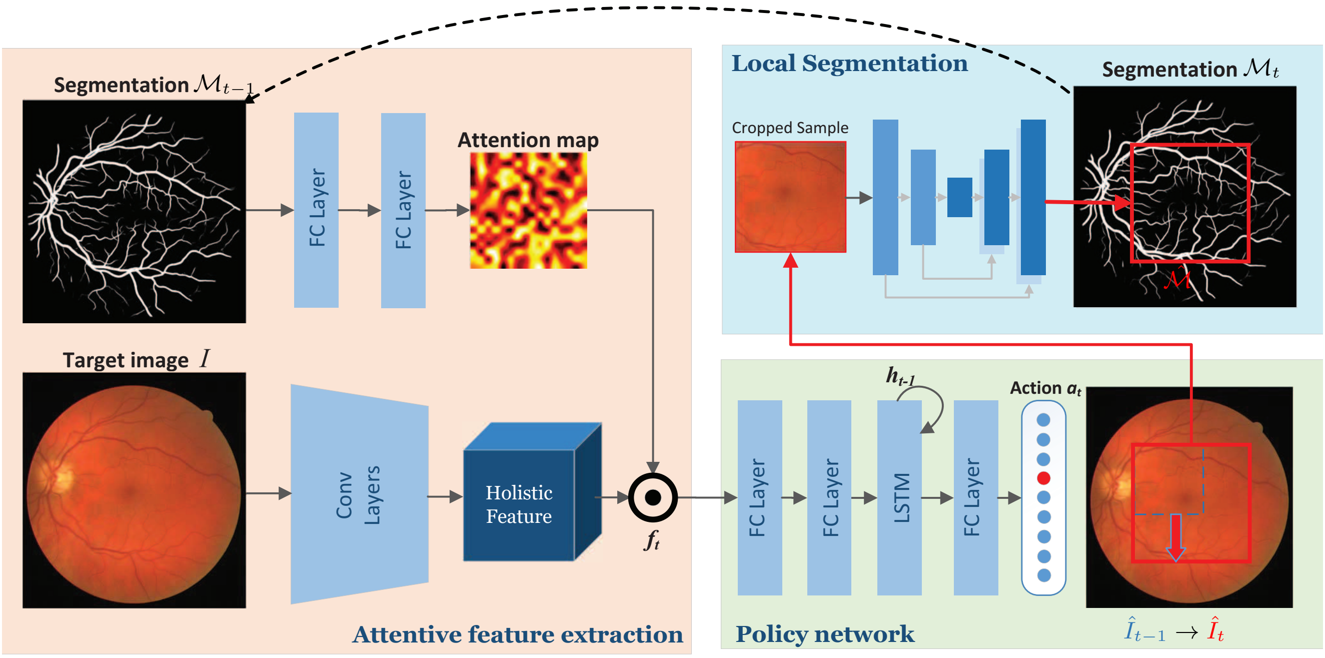


Figure 2. Our proposed approach consists of three modules. *Attentive feature extraction* first performs the feature extraction for the target image I, i.e., the holistic feature of I is masked by an attention map inferred from the latest segmentation $\mathcal{M}_{t-1}$ to obtain the feature f_t. Next, given the attentive holistic feature, the *policy network* will sample an action to select a proper patch $\hat{I}_t$ from the target image. In this figure, the sampled action is to enlarge the cropping window to obtain the new patch. Finally, in *local segmentation*, the selected patch is applied to generate a local segmentation $\hat{\mathcal{M}}$ and then update the global segmentation $\mathcal{M}_t$ that will be recursively used for sampling the next action.

54, 6, 20, 17]. Caicedo et al. [5] introduced a sequential algorithm that attentively localizes the objects in an image. Cao et al. [6] applied DRL for face hallucination, which sequentially discovers image patches that should be attached more attention and enhanced. Yun et al. [54] casted the problem of visual object tracking to a decision making process and applied DRL method to sequentially move the bounding box, achieving accurate tracking results. Han et al. [17] presented a Deep Q-learning based approach that simultaneously localizes and segments the target object in a video. There are also several methods attempting to apply RL techniques to the image segmentation problem [40, 37, 46, 16, 17], but they are limited to natural images or videos. In contrast to these previous works, the focus of this paper is about the progressive and adaptive refinement of curvilinear structure segmentation of particular biomedical images.

3. Proposed Approach

In this work, we formulate the image segmentation as a sequential decision-making process, in which an agent takes a sequence of actions to accomplish a global optimization task. Particularly, given an initial coarse segmentation, we introduce a segmentation agent that sequentially samples proper-sized patches from the target image for local segmentation and finally obtain the optimal global segmentation result. As visualized in Fig. 2, our proposed pipeline can be briefly formulated as:

$$f_t \leftarrow \mathcal{F}_e(I, \mathcal{M}_{t-1}), \tag{1}$$

$$a_t \leftarrow \mathcal{F}_\pi(f_t, h_{t-1}), \tag{2}$$

$$\hat{I}_t \leftarrow \mathcal{F}_a(I, \hat{I}_{t-1}, a_t), \tag{3}$$

$$\mathcal{M}_t \leftarrow \text{Merge}(\mathcal{F}_s(\hat{I}_t), \mathcal{M}_{t-1}). \tag{4}$$

Specifically, at each step t, the segmentation agent receives observations from the target image I, the latest segmentation mask $\mathcal{M}_{t-1}$, and the historical experience h_t. According to these observations, the *attentive feature extraction* module $\mathcal{F}_e$ extracts an attentive holistic feature f_t for representing the current state of the segmentation task, as shown in Eq. 1. Next, the *policy network* module uses the learned policy $\mathcal{F}_\pi$ in Eq. 2 to sample an action a_t from the action space $\mathcal{A}$, given the extracted feature f_t. The execution of this action (i.e., $\mathcal{F}_a(\cdot)$ in Eq. 3) will crop a new patch $\hat{I}_t$ from the image based on previously selected local patch $\hat{I}_{t-1}$. After that, the *local segmentation* model $\mathcal{F}_s$ will perform segmentation on the local patch $\hat{I}_t$ and merge it into the global segmentation mask $\mathcal{M}_t$ that will be used in the next time-step. After the execution of each action, the agent will receive a reward according to the pixel-wise error between the updated segmentation mask and the ground-truth. To sum up, the goal of the agent is to find an optimal patch that maximizes the accumulated reward, by iteratively performing Eq. 1, 2, 3 and 4.

3.1. Policy network

As the core of our framework, the policy network performs the sequential local patch mining. In the following part, we introduce our state and action formulation.

State: In our formulation, the state consists of two parts: the current segmentation mask and the historical experience, i.e. $s_t = \{\mathcal{M}_t, h_t\}$, where $\mathcal{M}_t$ indicates the global segmentation mask at the current time step t, and h_t is one LSTM latent layer which encodes the historical experience by incorporating all previous actions and is computed by forwarding the encoded history action vector h_{t-1}. In this way, the state of the decision making procedure not only considers the current observation but also involves the historical experience, which enables the segmentation agent to perceive the complete contextual information in a human-k'z'm'blike manner.

Actions: Given the input image I with size $W \times H$, the agent first selects one patch $\hat{I}_t = (x, y, w, h)$ where (x, y) refers to the top-left position of the patch and (w, h) refers to its width and height, and then updates the segmentation mask in the patch. We define the action space $\mathcal{A}$ as a group of transformation selections for adapting the size and the position of $\hat{I}_t$, including the translation actions, the scaling actions, and a termination action. In particular, the translation refers to the four actions which move $\hat{I}_t$ leftward, rightward, upward, and downward by 16 pixels, respectively. The four scaling actions will scale the width and the height of the patch window size by $\times 1$, $\times 0.5$, $\times 0.375$, and $\times 0.25$, respectively. In practice, we use a square patch, i.e,. $w \equiv h$, and both w and h are assigned with the default 512 pixels. Thus, the scaled patch sizes could be 512×512, 256×256, 192×192, or 128×128. The termination action will stop the segmentation procedure whenever it is chosen. In total, we have 9 actions, and each of them is then represented as a dim-9 one-hot action vector $a_t \in \{0, 1\}^9$.

3.2. Network architectures

Global segmentation. At the beginning, our model is given a coarse initial segmentation. This initial global segmentation mask $\mathcal{M}_0$ is produced using a U-Net architecture [39] that has been widely used in many vision tasks. This U-Net is trained using the complete images from training datasets.

Attentive feature extraction. To incorporate holistic cues for encoding the contextual dependency, we extract the holistic feature of the image using the network $\mathcal{F}_{global}$, which is the Conv1-5 layers of the pretrained VGG model [43] as shown in Fig. 2. Since the extracted global feature of the image does not vary over time, we need to incorporate it with the latest global segmentation mask $\mathcal{M}_t$ in order to extract a feature that is sensitive to the temporal variations of the segmentation mask in the image. However, sometimes, the update difference between $\mathcal{M}_t$ and $\mathcal{M}_{t-1}$ can be small, which slows the progress of the reinforcement learning. To extract the temporal-varying feature more effectively, we learn a binary mask to attend the feature extraction, where the attention module $\mathcal{F}_{att}$ consists two fully-connected layers as shown in Fig. 2. The attention module produces a binary attention map with the same spatial size as the holistic feature. In practice, the dimension of the holistic feature is $16 \times 16 \times 512$. The output dimension of the second FC layers in $\mathcal{F}_{att}$ is 256, which is reshaped into 16×16 as the attention map. As illustrated in Fig. 2, the holistic feature and the attention map are multiplied to compute the feature as:

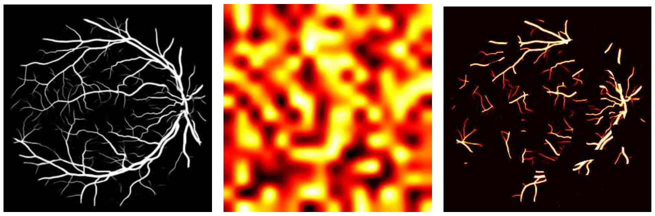


Figure 3. Visualization of the segmentation, the attention map, and the attentive regions of the segmentation. As shown, the entangled and the thin structures are highlighted.

$$f_t = \mathcal{F}_e(I, \mathcal{M}_{t-1}) = \mathcal{F}_{att}(\mathcal{M}_{t-1}) \odot \mathcal{F}_{global}(I), \quad (5)$$

where $\odot$ indicates the element-wise multiplication. The attention map tends to focus on the regions where the segmentation decision is still uncertain, as shown in Fig. 3.

Policy sampling. Next, the feature f_t is passed forward to the recurrent module, which encodes or memorizes the historical experience h_t. The recurrent module is composed of three fully-connected layers and an LSTM layer, as shown in Fig. 2. It outputs a transformation action a_t in the form of a one-hot vector. The action will then be used to crop a patch $\hat{I}_t$ from the image I.

Local segmentation. Finally, the selected patch $\hat{I}_t$ is fed into a local segmentation model to obtain a new segmentation mask $\mathcal{M}_t$. In practice, we set up the local segmentation model as the same U-Net architecture as the global segmentation model, except that it is trained using multi-scale patches randomly sampled from the training images. Generally, the local segmentation model does not have to be a U-Net, and it could be replaced with any encoder-decoder architectures. After the patch $\hat{I}_t$ is fed to the local segmentation model, a local segmentation mask $\hat{\mathcal{M}}$ is computed, which is of the same size as $\hat{I}_t$ and is used to update the segmentation mask by Merge$(\cdot)$. In particular, the Merge$(\cdot)$ in Eq. 4 is implemented as follows. Within the region of $\hat{I}_t$, the segmentation mask $\mathcal{M}_t$ is computed as the linear fusion of the local segmentation $\hat{\mathcal{M}}$ and the last step segmentation mask $\mathcal{M}_{t-1}$:

$$\mathcal{M}_t|_{\hat{I}_t} = \gamma \mathcal{M}_{t-1}|_{\hat{I}_t} + (1-\gamma)\hat{\mathcal{M}}, \quad (6)$$

while for the region outside $\hat{I}_t$, $\mathcal{M}_t$ just copy from the last step result: $\mathcal{M}_t|_{I \setminus \hat{I}_t} = \mathcal{M}_{t-1}|_{I \setminus \hat{I}_t}$. In this way, the current local segmentation will not completely replace the previous one in a single step and thus the smooth structures may be preserved.

Reward. The reward is defined as the improvement of the local segmentation from time-step $t-1$ to t, which encourages the agent to locate a patch that brings the maximum improvement. In the training stage, since the process is a trial-and-error learning, the segmentation may not be improved in each step. But the learned policy will eventually let the segmentation reaches the global optimal state. We adopt L_1 distance to measure the error of the segmentation. In particular, when the L_1 distance between the segmentation and its ground-truth decreases from $t-1$ to t, a positive reward will be obtained. Otherwise, the reward will be negative. Hence, the reward function R is designed as:

$$R = \lambda(||\hat{\mathcal{M}}^{gt}_{t-1} - \hat{\mathcal{M}}_{t-1}||_1 - ||\hat{\mathcal{M}}^{gt}_{t} - \hat{\mathcal{M}}_{t}||_1), \quad (7)$$

where $\hat{\mathcal{M}}_t = \mathcal{M}_t|_{\hat{I}_t}$ is the restriction of the segmentation mask $\mathcal{M}_t$ in the patch region defined by $\hat{I}_t$, $\hat{\mathcal{M}}^{gt}_t$ denotes the ground-truth of the local segmentation for $\hat{\mathcal{M}}_t$. λ is set to 3 and ϵ is set to 10.

3.3. Actor-Critic based learning

We use Temporal Difference based Actor-Critic algorithm [22] to learn a policy to sample patches from target images for optimally improving the segmentation performance. In the training stage, the algorithm experiences lots of training examples of high rewards from good actions and negative rewards from bad actions, by selecting an action from the existing policy network using a softmax output layer. When the training progresses, the algorithm can iteratively update the policy network, which is called the Actor in the Actor-Critic framework, and a value function referred as the Critic module, which guides the gradient update of the Actor module according to a Temporal Difference (TD) error signal. The TD signal is an approximation to the advantage function, i.e., how good the action is compared to the average of all the actions in terms of the segmentation quality. After convergence, the Actor-Critic algorithm can provide a high-quality policy for performing high-quality patch sampling of a given target image.

4. Experiments

4.1. Datasets and implementation details

Datasets. We evaluate our approach on two types of biomedical image data featuring different linear structures.

- **Blood vessel images**. We experiment on DRIVE [47] and STARE [19], containing 40 and 20 images, respectively. Both contain manual segmentations of the blood vessels by expert annotators. Following [31], for DRIVE, we use the standard train/test split. For STARE, we use the first 10 images as the training set and the last 10 as the test set.

- **Electron Microscopy images** (EM). The task of the EM dataset [2] is to detect neuronal boundaries. There are 30 training images with ground truth annotations and 30 test images for which the ground-truth is withheld by the organizers. Following the practice of [35], we split the training set into 15 training and 15 test images.

Implementation details. We have built our networks with Tensorflow and trained on a single NVIDIA GTX 1080Ti. Particularly, in the attention learning module, the output sizes of the two FC layers are 512 and 256. In the FC layers of the policy network, the output sizes are 30, 30, and 10, while the output dimension of LSTM is 64. Besides, we deploy the same U-Net architecture for global and local segmentation, which consists of 5 convolutional layers for downsampling and 5 deconvolutional layers for upsampling. As mentioned, the global segmentation model is trained on the complete images only, while the local segmentation model is trained on multi-scale patches cropped from the training image. Their input spatial resolution is 512×512, and thus all training samples are resized to 512×512 before being fed into the network. Furthermore, we perform data augmentation by mirroring and rotating the training images. Additionally, in the EM dataset, we apply elastic deformations as suggested in [39] to compensate for the small amount of training data.

In the training phase, we iteratively train over all images for each training dataset. We treat each image as a unique training scenario for the Actor-Critic framework. It runs for maximally 500 steps for each image to collect adequate observations. We choose Adam as the optimization solver in the training stage and the learning rates of the Actor and the Critic are set to 10^{-4} and 10^{-3} respectively.

Metrics. For evaluating the comparison results on DRIVE, we use F1-measure, which is often applied for evaluating the segmentation results. We also compute the pixel-wise precision-recall for reference. For the EM and STARE dataset, we follow the metrics used in [35] that adopts F1-measure along with correctness, completeness, and quality [18]. They are metrics designed specifically for linear structures, which measure the similarity between predicted structures and ground truth. All metrics are computed at the optimal point of comparison methods.

4.2. Results analysis

Learning progress analysis. To analyze the learning progress, we evaluate the our model at different training stages on the DRIVE dataset. We use the global segmentation model as the baseline model without learning any policy sampling, denoted as *Global*. We compare

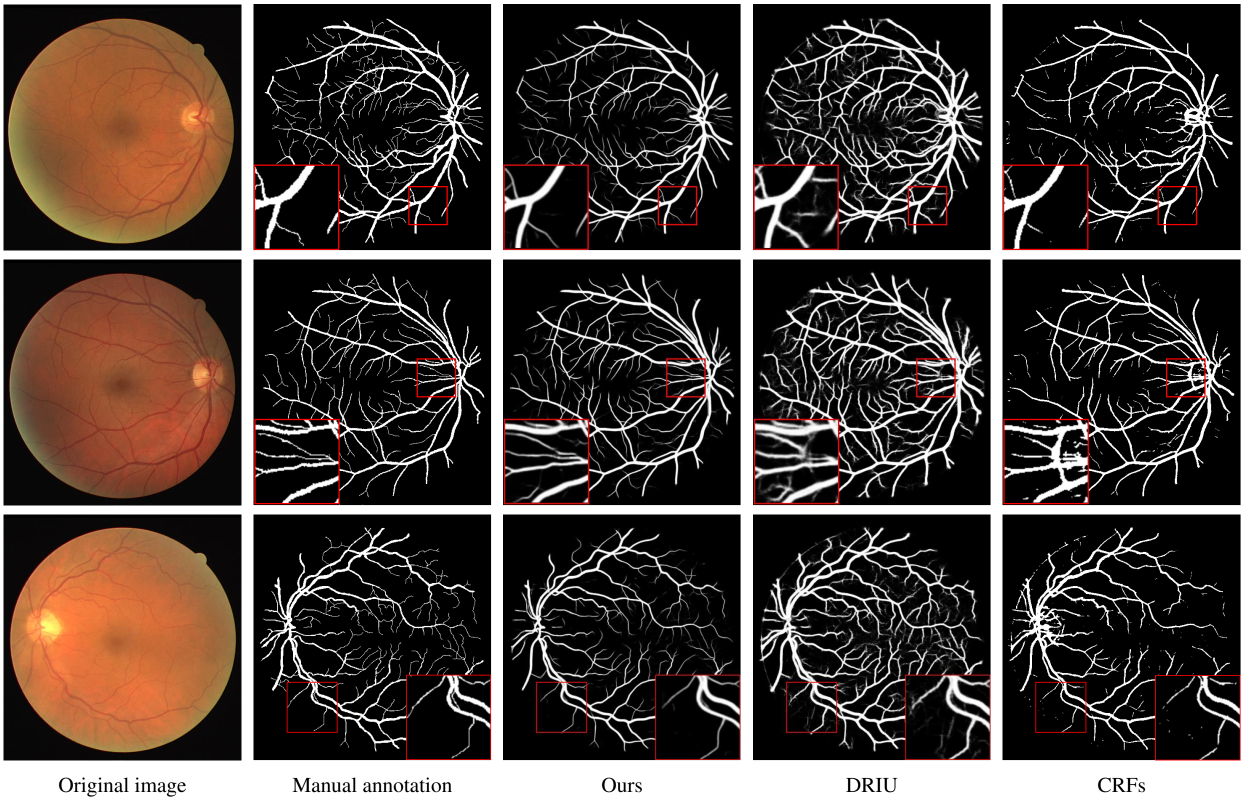


Figure 4. Our approach is compared against DRIU and CRFs on three exemplar images from the DRIVE dataset. The first two columns refer to the target image and the manual annotation. The third to fifth columns refer to the segmentation results of ours, DRIU, and CRFs, respectively. In addition, we highlight several local patches for a more detailed comparison. The zoomed-in patches on the 1^{st} and 2^{nd} row are shown at the bottom-left corner of each image, while the ones on the 3^{rd} row are at the bottom-right corner.

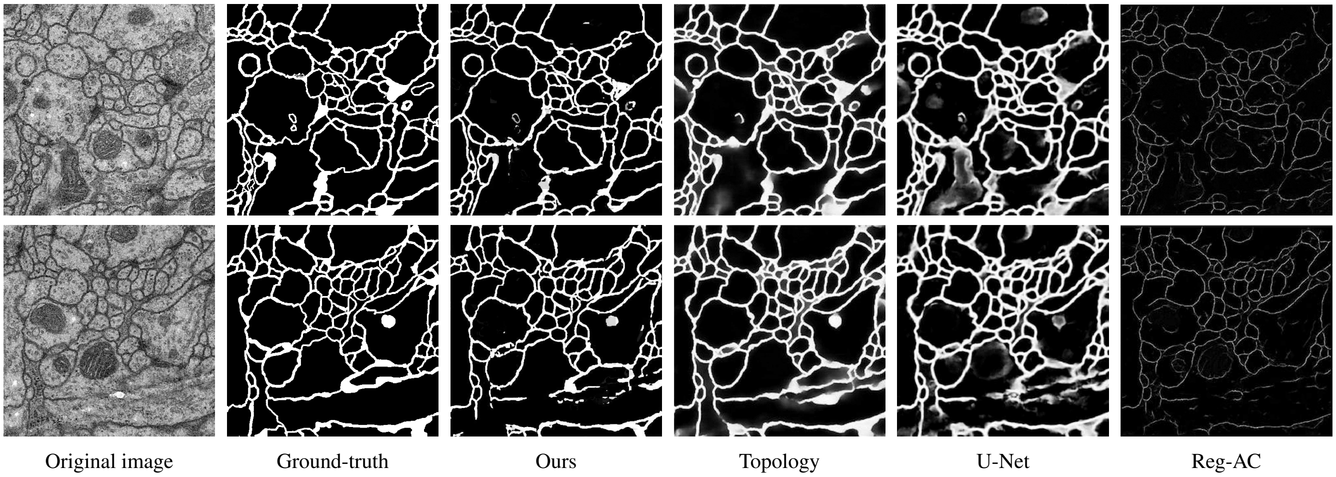
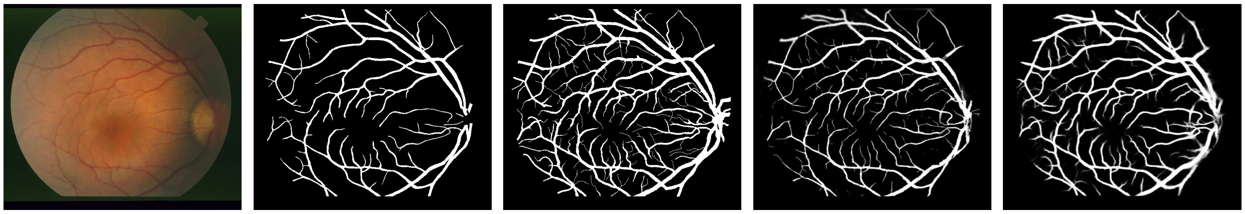


Figure 5. Comparison with Topology, U-Net, and Reg-AC based on two images in the EM dataset. Results are from the original work of [35].

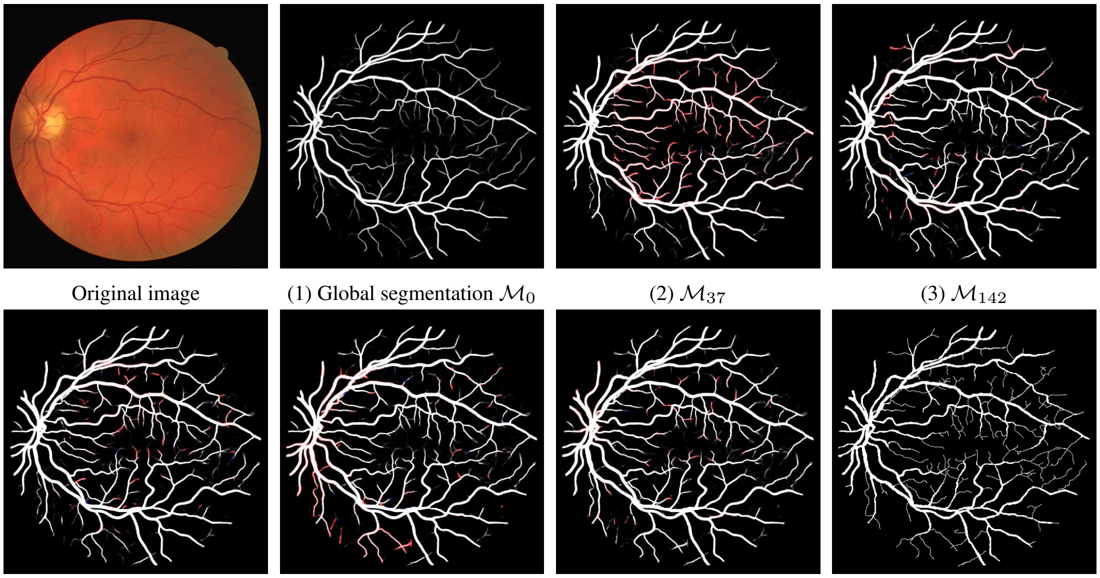
the models with policy sampling trained for 1, 2, and 3 iterations over all of the 40 training images, respectively. They are denoted as *Ours-iter1*, *Ours-iter2*, and *Ours-iter3*. Results in Tab. 1 demonstrate that our model with policy sampling obviously improves the performance of the global model, since *Ours-iter1* and *Ours-iter2* outperform it in all metrics. However, the model can hardly be improved for more than 3 iterations, since *Ours-iter3* degrades the precision with little gain in other metrics.

Quantitative results. We evaluate our proposed approach in the tasks of blood vessel segmentation in DRIVE and STARE as well as the neuron boundary detection in the EM dataset. For the EM dataset, we compare against the methods: CHM-LDNN [41], Reg-AC [44], U-Net [39], and [35] (denoted as *Topology* for convenience), according to the original results published in the latest work [35].



Original image Manual annotation 1 Manual annotation 2 Ours DRIU

Figure 6. Comparison against DRIU with reference to two manual annotations based on an image from STARE dataset.



Original image (1) Global segmentation $\mathcal{M}_0$ (2) $\mathcal{M}_{37}$ (3) $\mathcal{M}_{142}$

(4) $\mathcal{M}_{238}$ (5) $\mathcal{M}_{309}$ (6) $\mathcal{M}_{409}$ Manual annotation

Figure 7. An example of how our approach progressively refines the segmentation from (1) to (6). We highlight the changes from the previous segmentation using colors of red and blue. The red color indicates the enhanced regions, while the blue color indicates the opposite.

Model	F1	Prec.	Recall
Global	0.7691	0.8652	0.7003
Ours-iter1	0.8074	0.8802	0.7297
Ours-iter2	0.8195	**0.8914**	0.7674
Ours-iter3	**0.8203**	0.8783	**0.7729**

Table 1. The comparison of the performances for models with different training configurations on the DRIVE dataset.

For the blood vessel segmentation, our model is compared with the state-of-the-art approaches including DRIU [31], N^4 fields [14], Kernel Boost [3], HED [51], CRFs [36], Wavelets [45], and SE [9]. Note that these state-of-the-art results are provided in the public benchmark [1].

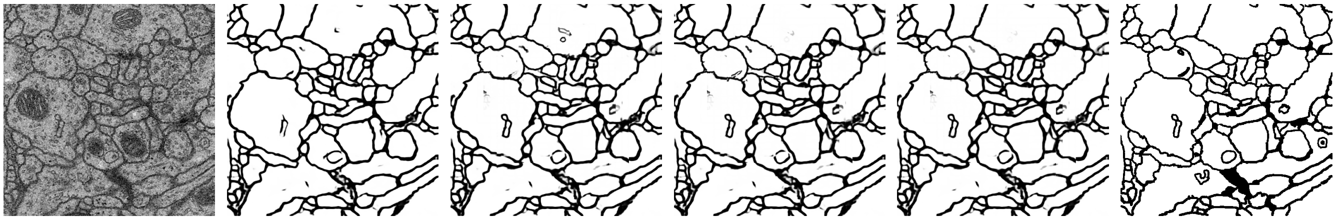
In Tab. 2, we first conduct comparison experiments on the EM dataset and the STARE dataset. We not only measure the F1 score of methods, but also evaluate the correctness, the completeness, and the quality of extracted curvilinear structures by thinning the segmentation. As illustrated, our approach outperforms other approaches, thanks to our context-aware segmentation's capability in preserving smooth and continuous structures.

Dataset	Methods	F1	Corr.	Comp.	Quality
STARE	HED	0.8050	0.3801	0.5021	0.2749
	Wavelets	0.7733	0.3761	0.4529	0.2590
	DRIU	0.8307	0.4725	0.5227	0.3306
	Ours	**0.8415**	**0.5473**	**0.5563**	**0.3810**
EM	HED	0.7227	-	-	-
	CHM-LDNN	0.8072	-	-	-
	Reg-AC	-	0.7110	0.6647	0.5233
	U-Net	0.7952	0.6911	0.7128	0.5406
	Topology	0.8230	0.7227	0.7358	0.5722
	Ours	**0.8345**	**0.7671**	**0.9152**	**0.7160**

Table 2. Experimental results on the STARE and the EM dataset. The comparison metrics include F1-measure and three line structure assessment metrics: correctness (Corr.), completeness (Comp.), and quality.

In addition, in Tab. 3, we compare our approach with several state-of-the-art approaches on the DRIVE dataset, including the DCNN-based approach DRIU [31] incorporated with the technique of deep feature fusion, which achieves the top performance in their public benchmark. It is demonstrated that our approach outperforms others in terms of F1-measure and the balanced precision and recall,

[1] http://www.vision.ee.ethz.ch/ cvlsegmentation/driu/index.html



Original image (1) $\mathcal{M}_0$ (2) $\mathcal{M}_{41}$ (3) $\mathcal{M}_{56}$ (3) $\mathcal{M}_{58}$ Ground-truth

Figure 8. From (1) to (4), an examplar image from the EM dataset shows how our approach progressively refines the segmentation.

Methods	F1	Prec.	Recall	Corr.	Comp.	Quality
Wavelets	0.7618	0.7479	0.7763	0.4918	0.2134	0.1782
SE	0.6584	0.6784	0.6396	0.3794	0.1603	0.1259
HED	0.7959	0.7981	0.7938	0.4383	0.4157	0.2703
Kernel Boost	0.8003	0.8098	0.7929	0.4689	0.4201	0.2840
N^4 Fields	0.8052	0.8081	0.8024	0.5650	0.3646	0.2843
CRFs	0.7812	0.7783	0.7842	0.4939	0.4031	0.2856
DRIU	0.8221	0.8179	0.8264	0.4734	**0.4725**	0.3137
Ours	**0.8353**	**0.8288**	**0.8419**	**0.5768**	0.4639	**0.3432**

Table 3. Experimental results on the DRIVE dataset. The comparison metrics include F1-measure, precision, recall, correctness, completeness, and quality.

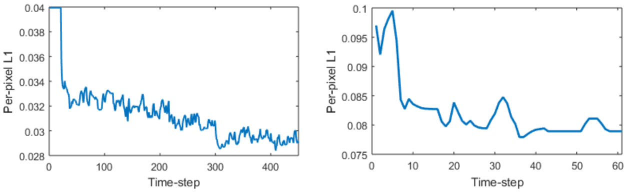


Figure 9. Illustration of the per-pixel $L1$ loss in the process of our segmentation for the examples in Fig. 7 (left) and Fig. 8 (right).

since our model can provide more consistent segmentation in blood vessels.

Qualitative results. We illustrate some typical results from these datasets. First, in Fig. 4, provided three retinal images of the DRIVE dataset, our approach is compared against DRIU and CRFs. Generally, DRIU produces noisy results with false positive predictions, while the ones created by CRFs have a lot of missed detections. As highlighted in the zoomed-in red boxes, the vessel structures are well preserved and smooth with very few noise. Next, in Fig. 5, we use the two examples from [35] for comparison. Generally, our produced results show higher quality than others. Since the test image is challenging, although there are some mis-detections observed in our results, our results have generated clearer boundaries and fewer noises than other methods. Lastly, we experiment on the image from the STARE dataset. Note that the training data is labeled by two persons. Our model is trained on data from the first annotator, but we observe in Fig. 6 that our model manages to extract thin structures similar to the second annotation.

Context-aware sampling. We first experiment on an example from DRIVE to demonstrate our sampling procedure. Our approach runs for 450 steps on this image and its performance is illustrated as the curve of per-pixel $L1$ loss of Fig. 9 (left). In Fig. 7, we select several key frames to show how our approach recurrently refines the segmentation at the 37^{th}, 142^{nd}, 238^{th}, 309^{th}, and 409^{th} steps. The improvements for the segmentation between frames are highlighted in colors. We observe that the agent first enhances the thin structures in the image center, but it also leaves discontinuity in the segmentation mask. Then, the agent moves around to remove the discontinuity and tries to filter out noises in the empty space. In the last few steps, the agent moves to image boundary to enhance the segmentation of the main branches. In Fig. 5, we illustrate the procedure of our segmentation on another image from the EM dataset. We run for 60 steps in this image, shown in Fig. 9 (right). In 41^{st} step, small circles are closed, but some of the neuronal boundaries are distorted. Then, the boundaries are refined, while noises inside the neuron are removed at the 56^{th} and 58^{th} steps.

5. Conclusion

In this paper, we propose a curvilinear structure segmentation algorithm using context-aware recurrent networks. Rather than segmenting the entire image or fixed-sized local patches, our method sequentially attends to segment a local patch from the target image with a proper size, where the strategy of choosing local patches is learned based on the contextual information of the image. Hence, the segmentation result can be progressively refined.

Due to the limited training data, we only adopt a small discrete set of actions, which may increase the number of sampled actions for each image. Additionally, since we apply the naive U-Net as the segmentation model, it is difficult to accurately identify all of those thin structures. As the future work, a richer set of actions may be applied and an advanced segmentation network could be applied to improve the performance and the versatility of our model for more vision tasks.

Acknowledgement This project is supported by the National Natural Science Foundation of China (No. 61702104, 61873067, and 61702194), the Natural Science Foundation of Fujian Province (No. 2016J05155 and 2018J01798), and the Guangzhou Key Industrial Technology Research fund (No. 201802010036).

References

[1] R. Alshehhi and P. R. Marpu. Hierarchical graph-based segmentation for extracting road networks from high-resolution satellite images. *ISPRS*, 2017. 2

[2] I. Arganda-Carreras, S. C. Turaga, D. R. Berger, D. Ciresan, A. Giusti, L. M. Gambardella, J. Schmidhuber, D. Laptev, S. Dwivedi, J. M. Buhmann, T. Liu, M. Seyedhosseini, T. Tasdizen, L. Kamentsky, R. Burget, V. Uher, X. Tan, C. Sun, T. D. Pham, E. Bas, M. G. Uzunbas, A. Cardona, J. E. Schindelin, and H. S. Seung. Crowdsourcing the creation of image segmentation algorithms for connectomics. *Frontiers in Neuroanatomy*, 2015. 2, 5

[3] C. J. Becker, R. Rigamonti, V. Lepetit, and P. Fua. Supervised feature learning for curvilinear structure segmentation. In *MICCAI*, 2013. 2, 7

[4] T. Beier, B. Andres, U. Kthe, and F. A. Hamprecht. An efficient fusion move algorithm for the minimum cost lifted multicut problem. In *ECCV*, 2016. 1, 2

[5] J. C. Caicedo and S. Lazebnik. Active object localization with deep reinforcement learning. In *ICCV*, 2015. 3

[6] Q. Cao, L. Lin, Y. Shi, X. Liang, and G. Li. Attention-aware face hallucination via deep reinforcement learning. In *CVPR*, 2017. 3

[7] H. Chen, X. Qi, L. Yu, and P.-A. Heng. Dcan: Deep contour-aware networks for accurate gland segmentation. In *CVPR*, 2016. 1, 2

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2018. 2

[9] P. Dollar and C. L. Zitnick. Structured forests for fast edge detection. In *ICCV*, 2013. 7

[10] M. Drozdzal, G. Chartrand, E. Vorontsov, M. Shakeri, L. D. Jorio, A. Tang, A. Romero, Y. Bengio, C. Pal, and S. Kadoury. Learning normalized inputs for iterative estimation in medical image segmentation. *Medical Image Analysis*, 2018. 1, 2

[11] P. P. R. Filho, P. C. Cortez, and V. H. C. de Albuquerque. 3d segmentation and visualization of lung and its structures using ct images of the thorax. *Journal of Biomedical Science and Engineering*, 2013. 2

[12] B. Fischl, D. H. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. van der Kouwe, R. Killiany, D. Kennedy, S. Klaveness, A. Montillo, N. Makris, B. Rosen, and A. M. Dale. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 2002. 2

[13] R. Gaetano, J. Zerubia, G. Scarpa, and G. Poggi. Morphological road segmentation in urban areas from high resolution satellite images. In *International Conference on Digital Signal Processing*, 2011. 2

[14] Y. Ganin and V. S. Lempitsky. N4-fields: Neural network nearest neighbor fields for image transforms. *ACCV*, 2014. 7

[15] F. C. Ghesu, E. Krubasik, B. Georgescu, V. Singh, Y. Zheng, J. Hornegger, and D. Comaniciu. Marginal space deep learning: efficient architecture for volumetric image parsing. *IEEE transactions on medical imaging*, 2016. 2

[16] V. Goel, J. Weng, and P. Poupart. Unsupervised video object segmentation for deep reinforcement learning. *NIPS*, 2018. 3

[17] J. Han, L. Yang, D. Zhang, X. Chang, and X. Liang. Reinforcement cutting-agent learning for video object segmentation. In *CVPR*, 2018. 3

[18] C. Heipke, H. Mayer, C. Wiedemann, and O. Jamet. Empirical evaluation of automatically extracted road axes. 1998. 5

[19] A. W. Hoover, V. Kouznetsova, and M. H. Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical Imaging*, 2000. 2, 5

[20] Z. Jie, X. Liang, J. Feng, X. Jin, W. F. Lu, and S. Yan. Tree-structured reinforcement learning for sequential object localization. *NIPS*, 2016. 3

[21] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *CVPRW*, 2016. 1

[22] V. Konda and J. N. Tsitsiklis. *Actor-critic algorithms*. 2002. 2, 5

[23] H. Larochelle and G. E. Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *NIPS*, 2010. 2

[24] M. W. K. Law and A. C. S. Chung. Three dimensional curvilinear structure detection using optimally oriented flux. In *ECCV*, 2008. 1, 2

[25] D.-T. Lin, C.-C. Lei, and S.-W. Hung. Computer-aided kidney segmentation on abdominal ct images. *International conference of the engineering in medicine and biology society*, 2006. 2

[26] G. Lin, A. Milan, C. Shen, and I. D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017. 2

[27] H. Ling, S. Zhou, Y. Zheng, B. Georgescu, M. Suehling, and D. Comaniciu. Hierarchical, learning-based automatic liver segmentation. In *CVPR*, 2008. 2

[28] M. G. Linguraru, J. K. Sandberg, Z. Li, F. Shah, and R. M. Summers. Automated segmentation and quantification of liver and spleen from ct images using normalized probabilistic atlases and enhancement estimation. *Medical Physics*, 2010. 2

[29] Y. Liu, D. M. Nguyen, N. Deligiannis, W. Ding, and A. Munteanu. Hourglass-shapenetwork based semantic segmentation for high resolution aerial imagery. *Remote Sensing*, 2017. 1

[30] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2

[31] K.-K. Maninis, J. Pont-Tuset, P. A. Arbelaez, and L. J. V. Gool. Deep retinal image understanding. *MICCAI*, 2016. 1, 2, 5, 7

[32] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla. Semantic segmentation of aerial images with an ensemble of cnns. *ISPRS*, 2016. 1

[33] V. Mnih and G. E. Hinton. Learning to label aerial images from noisy data. In *ICML*, 2012. 2

[34] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 2

[35] A. Mosinska, P. Marquez-Neila, M. Kozinski, and P. Fua. Beyond the pixel-wise loss for topology-aware delineation. In *CVPR*, 2018. 1, 2, 5, 6, 8

[36] J. I. Orlando and M. B. Blaschko. Learning fully-connected crfs for blood vessel segmentation in retinal images. In *MICCAI*, 2014. 2, 7

[37] M. A. Reza and J. Kosecka. Reinforcement learning for semantic segmentation in indoor scenes. *arXiv preprint arXiv:1606.01178*, 2016. 3

[38] E. Ricci and R. Perfetti. Retinal blood vessel segmentation using line operators and support vector classification. *IEEE Transactions on Medical Imaging*, 2007. 2

[39] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *MICCAI*, 2015. 1, 2, 4, 5, 6

[40] F. Sahba, H. R. Tizhoosh, and M. M. A. Salama. A reinforcement learning framework for medical image segmentation. In *International Joint Conference on Neural Networks*, 2006. 3

[41] M. Seyedhosseini, M. Sajjadi, and T. Tasdizen. Image segmentation with cascaded hierarchical models and logistic disjunctive normal networks. In *ICCV*, 2014. 6

[42] W. Shen, B. Wang, Y. Jiang, Y. Wang, and A. L. Yuille. Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection. In *ICCV*, 2017. 1, 2

[43] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015. 4

[44] A. Sironi, E. Turetken, V. Lepetit, and P. Fua. Multiscale centerline detection. *TPAMI*, 2016. 6

[45] J. V. B. Soares, J. J. G. Leandro, R. M. C. Jr., H. F. Jelinek, and M. J. Cree. Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification. *IEEE Transactions on Medical Imaging*, 2006. 2, 7

[46] G. Song, H. Myeong, and K. M. Lee. Seednet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation. In *CVPR*, 2018. 3

[47] J. Staal, M. Abramoff, M. Niemeijer, M. Viergever, and B. van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 2004. 2, 5

[48] T. Tong, R. Wolz, P. Coupe, J. V. Hajnal, and D. Rueckert. Segmentation of mr images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. *NeuroImage*, 2013. 2

[49] E. Turetken, C. J. Becker, P. Glowacki, F. Benmansour, and P. Fua. Detecting irregular curvilinear structures in gray scale and color imagery using multi-directional oriented flux. In *ICCV*, 2013. 1, 2

[50] C. Ventura, J. Pont-Tuset, S. Caelles, K.-K. Maninis, and L. V. Gool. Iterative deep learning for road topology extraction. *BMVC*, 2018. 2

[51] S. Xie and Z. Tu. Holistically-nested edge detection. *IJCV*, 2017. 7

[52] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *ICLR*, 2016. 2

[53] Q. Yu, L. Xie, Y. Wang, Y. Zhou, E. K. Fishman, and A. L. Yuille. Recurrent saliency transformation network: Incorporating multi-stage visual cues for small organ segmentation. *CVPR*, 2018. 1, 2

[54] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Y. Choi. Action-decision networks for visual tracking with deep reinforcement learning. In *CVPR*, 2017. 3

[55] Y. Zhang, Z. Qiu, T. Yao, D. Liu, and T. Mei. Fully convolutional adaptation networks for semantic segmentation. *CVPR*, 2018. 2

[56] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017. 2

[57] M. Ziems, M. Gerke, C. Heipke, and U. Stilla. Automatic road extraction from remote sensing imagery incorporating prior information and colour segmentation. *ISPRS*, 2007. 2