IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2018, accepted March 21, 2018, date of publication April 6, 2018, date of current version May 9, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2823726

Detection of Curvilinear Structure in Images by a Multi-Centered Hough Forest Method

HANJIN ZHANG[1,2], YANG YANG[3,4], (Member, IEEE), AND HONGBIN SHEN[1,2]

[1]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai 200240, China
[2]Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai Jiao Tong University, Shanghai 200240, China
[3]Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[4]Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

Corresponding author: Hongbin Shen (hbshen@sjtu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61671288, Grant 91530321, Grant 61603161, and Grant 61725302 and in part by the Science and Technology Commission of Shanghai Municipality under Grant 16JC1404300, Grant 17JC1403500, and Grant 16ZR1448700.

ABSTRACT Automatic detection of curvilinear structure in images is a challenging task in computer vision applications. In this paper, we present a novel supervised learning method to identify curvilinear structure based on a modified Hough forest framework. We regard the curvilinear structure as a special object which has multiple object centers at centerline points, and construct a multi-centered Hough forest (MCHF) to cast confidence votes for each image point being a local curvilinear center in generalized Hough space. Considering the specific properties of curvilinear structure, we modify the classical Hough forest method in three aspects. First, the features are composed of a set of filter responses, and each filter is a base which is learned by sparse presentation techniques. Second, the offset of an image patch against its reference center is represented by a displacement function instead of Euclidean distance used in the classical Hough forest. Third, the centerline orientation is incorporated into the forest to reflect the local trend, which is considered in Hough voting. We conduct experiments on retinal vessel, neuron, and aerial road images. Both visualized and quantitative results demonstrate the good performance of MCHF in detecting complex curvilinear structures in various images.

INDEX TERMS Curvilinear structure, automatic detection, Hough forest.

I. INTRODUCTION

Curvilinear structure detection is a critical and challenging problem in image processing and computer vision applications. Curvilinear structures, also known as wiry or tube-like structures, widely exist in various images, such as roads and rivers in aerial or satellite images, neurons and vessels in medical images, tree branches and vines in nature images. Fig. 1 shows some typical curvilinear structures in different images.

Generally, curvilinear structures in images are irregular. Their curvature and orientation are changeful, and the line widths of different parts of the curvilinear structure also differ a lot. Most of the curvilinear structures have fractures and gaps, and the intensity and contrast varies frequently. Besides, in most cases, the curvilinear filaments intercross, overlap and enwind with each other, and multiple wiry objects form a complex tree or network structure. Therefore, automatic extraction of curvilinear structure is a difficult job in various image analyses.

To address this problem, numerous methods have been proposed and can be roughly grouped into three types. The first type of methods utilizes some characteristics of curvilinear structures, such as edges [1], ridges [2]–[4] and gradient vector flow [5], which are different from the background and other objects. We call them intuitive detection approaches. The second type of methods is learning-based. They regard the curvilinear structure as a special object and construct a machine learning model [6] to make a decision for each image unit, i.e., pixel, voxel, or local region. The third type of methods regards the whole curvilinear as a special graph and tries to analyze the topological structure to solve the problem [7]–[9], thus we name them graph-based approaches. Since the characteristics utilized in the models are not always peculiar to the curvilinear structure, the intuitive methods have some inherent shortcomings that are hard to overcome. Specifically, they tend to confound the curvilinear structure and other things in the background when the image context has a lot of ambiguous structures;

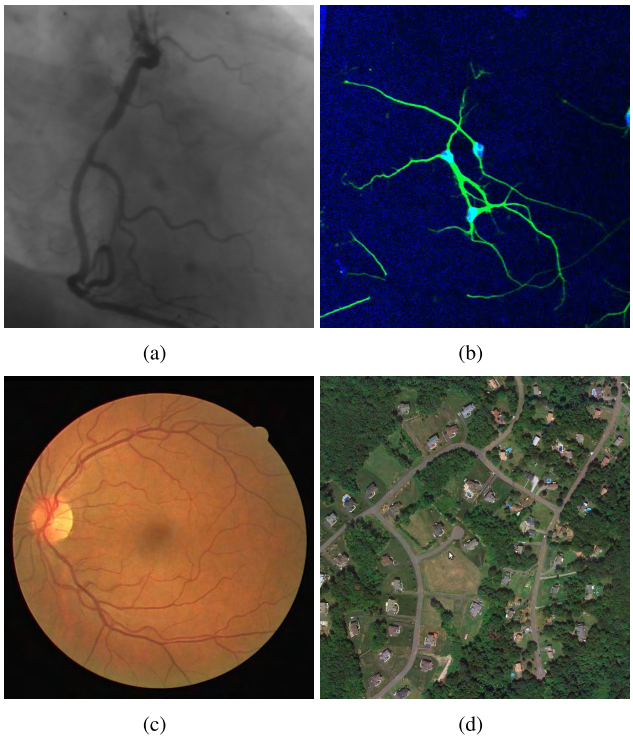


FIGURE 1. Examples of curvilinear structures. (a)-(d) are images of coronary angiogram, neuron confocal fluorescence, retinal vessel and aerial suburban roads, respectively.

while the graph-based methods are usually very effective when the curvilinear network is complicated. However, setting the initial vertices and edges of the graph model is not an easy job.

Compared with the other two types of methods, the learning-based methods have more flexibility in modeling and better performance for detecting complex curvilinear structures, thus have attracted a lot of research interests in recent years. Our model also falls into this category, in which the curvilinear structures are regarded as special objects. Unlike a normal object, the curvilinear structure consists of a large number of local line segments. To detect such object, we build an object recognition framework based on Hough forest [10], [11]. We extend the classical Hough forest into a multi-centered version where each centerline point is treated as an object center. Each tree in the multi-centered Hough forest (MCHF) is trained similarly as a decision tree in the random forest. At the detection stage, the patch centered at each point in the test image or FOV falls into a leaf node of every tree in the forest. By traversing all the trees in the forest, each point in the test image contains the accumulated votes about the probability that a tubular segment exists in this position. Thus the test image is converted into a voting map in Hough space. Then a directional non-maximum suppression is utilized to judge the position of each centerline point, and corresponding width and orientation can also be estimated by the accumulated position and orientation information of the leaf nodes in MCHF.

In summary, we construct a novel MCHF model specialized for the detection of curvilinear structure, where a new offset measure and the orientation information are used to estimate the position relationship of a patch with the curvilinear structure. The model is evaluated on four publicly available datasets, including the retinal blood vessels, the nerve and the aerial road images. The experimental results show that MCHF has promised performance in handling complex curvilinear structures in various images.

II. RELATED WORK

In Section I, we summarize the curvilinear detection methods into three types, intuitive methods, learning-based methods and graph-based methods. In this section, we describe these three types of the existing methods in details.

A. INTUITIVE DETECTION METHODS

The intuitive detection methods generally focus on the different characteristics of curvilinear structure and background in local region. Some of them extract discriminative features from edge [1], ridge [2]–[4] and gradient vector flow [5]. The active contour model [12] and threshold segmentation [13] have also been applied to the recognition of curvilinear structure. Besides, some researchers found out that the main difference of curvilinear structure and background come from second-order derivatives of their cross section [14]. Another representative method identifies curvilinear structures via the eigenvalues and eigenvectors of Hessian matrix [14]–[16]. These intuitive methods have laid foundation for many later studies, but they need Gaussian blur as a pretreatment, which may lead to confusion between lines and their adjacent points if the local structure is too wide or located in a branch region. To overcome this drawback, the optimally oriented flux (OOF) method [17] was proposed, which localizes the spherical region to compute the gradient flux by minimizing the inward flow. And later, Law and Chung [18] and Turetken *et al*. [19] further enhanced the performance. In addition, there are some other ideas of low-level linear extraction, such as tensor voting [20] and level sets variation [21]. They work well for many special applications but do not really overcome the inherent limitations.

B. LEARNING-BASED METHODS

The second type of methods consider the detection as a decision problem and employs supervised or unsupervised learning algorithms. These methods first construct a feature space, then build a learning or clustering model, and finally make a decision by classification or regression. Typical algorithms include boosted tree method for detecting the filament structures [22], [23], random forest model for curvilinear detection in Berks *et al.* [24] and higher-order CRF model for road extraction [25]. Becker *et al*. [26] proposed a gradient boosting based method that employs weak learners relying on the filter kernels, which are learned during boosting. The method obtains high accuracy but is time-consuming. Sironi *et al*. [27] considered the linear centerline extraction

as a regression problem. They trained a regressor in the scale space, which outperforms classification methods and is generally applicable to linear structures. However, if the curvilinear is dense, the locations of the points near centerlines may lead to ambiguity. A latest work [23] solves this problem by integrating structured and contextual features to construct a supervised learning framework. This method takes into consideration of local patch structure and the relation of local region with the whole object.

C. GRAPH-BASED METHODS

Since the wiry structure often looks like a tree or net, in recent years, some researchers adopted the graph or network topological structure to identify curvilinear structures. Turetken *et al*. [19], [28] modeled the task as a quadratic mixed integer programming problem in a graph of potential paths with weights. Chai *et al*. [8] developed a junction-point process by using Monte Carlo mechanism, and used the nodes to represent either a connection or an end in the line-network. De *et al*. [9] identified the filamentary structures via label propagation over directed graphs. This method can trace and separate filamentary structures into disjoint set of filaments. Most of the graph-based methods have achieved impressive results in their specific applications. However, almost all of the graph-based methods need some prior knowledge of the curvilinear structure before constructing the graph or network. For instance, in Chai's method [8], the junction of curvilinear objects must be known prior to defining the graph, and De's method [9] needs to segment the curvilinear structure first in the design of the directed graph.

III. MULTI-CENTERED HOUGH FOREST FOR CURVILINEAR STRUCTURE DETECTION

Previous work on curvilinear detection have two basic findings: i) the local appearance contains valuable information; ii) the orientation and multi-scale are helpful for identifying the structure. Our work is also guided by these two findings. In this section, the proposed MCHF method is described in details.

A. PROBLEM DESCRIPTION

As above-mentioned, we treat the curvilinear extraction as a special object detection problem. For an ordinary object, it is easy to demarcate the object region and the background by a bounding box in the image. However, the curvilinear structure often presents as a tree or network, and it is probably distributed in the entire image space or the field of view (FOV). Thus, it is hard to restrict the whole curvilinear into a specific bounding box.

Our aim is to detect all the curvilinear objects in a given image. In this study, we propose a multi-centered Hough forest (MCHF) model to extract curvilinear structures. The idea is based on the following assumption: no matter how complexity the curvilinear network is, it is just composed of many individual curves; and no matter how long or flexible the curves are, they are just composed of linear or tubular segments. Here, the individual curve means the local curvilinear structure from one junction or terminal to the nearest junction or terminal, which is composed of a set of local tubular segments. Specifically, a whole curvilinear network structure $\mathcal{N}$ consists of N individual curves $\mathcal{C}_i$, which are partitioned by line junctions or terminals, and an individual curve is a concatenation of M tubular segments $\mathcal{S}_{ij}(o, r, l, \theta)$, i.e.,

$$\mathcal{N} = \bigcup_{i=1}^{N}\{\mathcal{C}_i\} = \bigcup_{i=1}^{N}\{\bigcup_{j=1}^{M}\mathcal{S}_{ij}(o, r, l, \theta)\}, \tag{1}$$

where $\mathcal{S}_{ij}$ is a local tubular (we regard it has a rectangle shape) of center o, width $2r$, length l and orientation angle θ. Apparently, the area of $\mathcal{S}_{ij}(o, r, l, \theta)$ is approximately equal to $2rl$. Suppose the local tubular is narrow enough, i.e., $l \to 0$ and $M \to \infty$, then the local tubular segment center is equivalent to the centerline point at that segment, i.e., $o_j \in L_o$, where L_o is the centerline point set of the whole curvilinear structure $\mathcal{N}$. Thus, Eq. (1) can be reformulated as,

$$\mathcal{N} = \bigcup_{i=1}^{N}\int_{0}^{L_i}\mathcal{S}_i(o, r, l, \theta)dl, \tag{2}$$

where L_i is the length of the i-th individual curve in the curvilinear network. Besides, the local tubular segments are highly similar with each other, after adjusting their position, direction, and width by translation, scaling, or rotation.

B. SAMPLE COMPOSITION

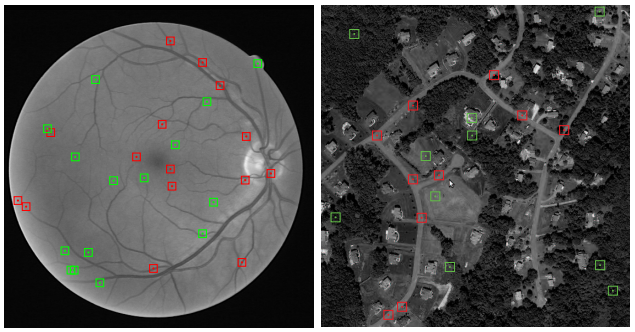
Extended from the settings of classical HF, a sample in MCHF is defined by a quadri-tuple $\mathcal{P} = \{F(x, I), c, \mathcal{O}(d, r), \theta\}$, where $F(x, I)$ is the feature vector which represents the local patch appearance, $c \in C$ indicates the label of the patch and $C = \{0, 1\}$, $\mathcal{O}(d, r)$ is the offset description function associated with the distance d from the nearest centerline point whose local tubular segment radius is r, and θ is the line direction of the nearest centerline point. The positive samples are the patches whose centers are located in the curvilinear regions, i.e., $x \in \mathcal{N}$; while negative samples are the patches whose centers are in the background, i.e., $x \notin \mathcal{N}$, as shown in Fig. 2.

1) PATCH APPEARANCE DESCRIPTOR

Here we adopt a set of features to describe the local appearance of curvilinear structure. References [29] and [30] have demonstrated that learning a set of filters using sparse presentation can produce good features for representing linear structure. And, these filters have rotational invariance that is suitable for curvilinear structure detection [31]. Here we denote the feature vector by $F(x, I)$,

$$F(x, I) = [(f_1 * I_1)(x), \ldots, (f_i * I_j)(x), \ldots, (f_M * I_N)(x)]^T, \tag{3}$$

where x is the local position of the image, f_i is one of the learned filter bank via sparse coding, and I_j indicates one of



(a) (b)

FIGURE 2. The positive and negative sample patches. The red rectangles are positive samples and green ones are negative samples.

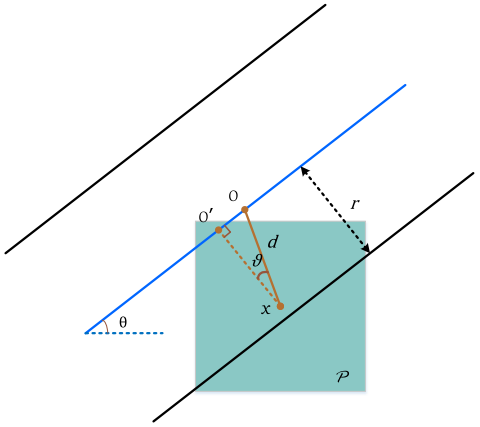


FIGURE 3. Illustration of the offset descriptor.

the scaled image which is generated by Gaussian smoothing and sampling. The feature vector of a patch consists of the convolutional responses at the patch center with a set of scales. As a set of complete basis, in each feature channel k ($k \in \{1, 2, \ldots, MN\}$), the component $(f_i * I_j)(x)$ can represent an aspect or status of local appearance, and the whole feature vector characterizes the local appearance from an arbitrary point with a group of selected scales.

2) OFFSET DESCRIPTOR

Since the curvilinear object centers are set by the centerline points, the definition of offset needs to be modified correspondingly. Let x, o and d denote a patch center, a centerline point and the distance between these two points, respectively. Since the offset estimation only relies on the adjacent centerline points, as [31] and [32] mentioned, the valid offset d only depends on the distance between x and its nearest centerline point o_k. In addition, the offset component of d along the centerline orientation θ is useless for the curvilinear object detection, thus only the normal orientation component of d is kept, i.e.,

$$d^\star = d \cos \vartheta, \tag{4}$$

where ϑ is the angle between the local centerline orientation and the connecting line joining the patch center point and the centerline point, as Fig. 3 shows. In most cases, the shortest distance $d^\star$ from the patch center x to o is the length of the straight line connecting these two points and perpendicular to the local centerline orientation θ, i.e., $\vartheta = 0$ and $d^\star = d$.

The classical Hough Forest defines the offset of positive samples as the Euclidean distance between the patch center and the object center, while the offset of negative samples is set as 0. However, for a curvilinear structure, we can't define the offset in that way. First, $d^\star$ is very short because the positive patch is in or cross the curvilinear structure which is very close to the centerline. And $d^\star$ could even be equal to 0 if the patch center is located at the centerline. Thus it is hard to distinguish the offsets for positive and negative samples. Second, as the width of curvilinear structure varies with location, the $d^\star$s of the thick and thin curvilinear structures are different. For measuring the position relationship between a patch center and its nearest centerline point, we use

$$\mathcal{O}(d, r) = \eta(1 - \frac{d^\star}{r}) \tag{5}$$

to represent the offset information, where r is half the width of the local tubular segment $S(o, r, l, \theta)$, and η indicates the coefficient, which is usually greater than 1 to emphasize the impact of displacement. Eq. (5) can fully represent the position relationship of the curvilinear structure. First, $\frac{d^\star}{r}$ reflects the ratio between the distance $d^\star$ and the curvilinear radius r, which can effectively avoid the error caused by the direct calculation of Euclidean distance. Second, for a special tubular segment $\mathcal{S}_k$, r is fixed, and Eq. (5) is a monotonic decreasing function of $d^\star$. The maximum value is achieved when the patch center is located at the centerline, and the offset description value decreases as the distance between centerline point and patch center increases. Third, it is easy to distinguish the positive and negative samples: for a positive sample, the patch center is located in the curvilinear structure, i.e., $d^\star \leq r_k$ and $\mathcal{O}(d, r_k) \geq 0$; while for a negative patch, apparently $\mathcal{O}(d, r_k) < 0$.

3) ORIENTATION DESCRIPTOR

There is a significant difference from the classical Hough forest (HF) method and the MCHF method. In HF, a specific object has one and only one center corresponding to it, and the relative positions of samples are determined by this center point; while in MCHF, since we assume that the curvilinear structure is composed by a set of local tubular segments and each tubular segment has its own center, the relative positions of samples are determined by their closest tubular structure centers. Accordingly, the tubular segment $\mathcal{S}_k$ which is centered at o_k is arranged in a special direction θ in the curvilinear structure $\mathcal{N}$. We mark the direction of the patch with the orientation of its nearest centerline point, i.e., θ, since the valid structure part in a positive patch is approximatively parallel to the nearest centerline orientation. And, for a negative patch, its direction can also be represented

by the direction of its nearest centerline point, because the class information of patches is not considered when they are grouped by the orientation information in the splitting process in a Hough tree.

Therefore, we incorporate the orientation θ into the sample to cast a corresponding location vote in the generalized Hough space. By the way, the normal orientation $\theta_\perp$ of the tubular segment $\mathcal{S}_k$ can be an alternative to represent the local direction of the curvilinear structure.

C. TRAINING

The MCHF is an ensemble of trees, and each tree $\mathcal{T}$ is constructed from a subset of training samples through random sampling with replacement. As the tree builds up, each node receives a group of patches from their parent node and are splitted continuously until they reach a leaf node, where the leaf node is a node which satisfies either of the following criteria: i) the depth of the node reaches the maximal depth d_{max}, ii) the number of samples assigned to this node is no more than a pre-defined value, i.e., the minimum number of leaf node samples N_{min}.

The binary test at a splitting node is defined in Eq. (6),

$$\mathcal{B}(f_i(x, I)) = \begin{cases} 1, & \text{if } f_i(x, I) \geq \tau, \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where $f_i(x, I)$ is a channel feature, and τ is the threshold. The key issue is how to purify the samples of each node during the training process, in order to minimize the uncertainty of the samples for selecting the best attribute in each internal node. According to Gall's definition [10], suppose S denotes the sample set $\{\mathcal{P}\}$ in a node, the class uncertainty can be calculated as,

$$U_1(S) = -|S| \sum_{c \in C} p(c|S) \ln p(c|S), \tag{7}$$

where the summation item is the class entropy S and it measures the impurity of the class. Eq. (8) defines the uncertainty of offset in the local line segment region,

$$U_2(S) = \sum_{p \in S} \|\mathcal{O}(d_p, r_p) - \mathcal{O}(d_{mean}, r_{mean})\|^2, \tag{8}$$

where d_{mean} and r_{mean} are the mean displacement of d_i and r_i over $\mathcal{S}$ in such node. The summation is over all the patches in the node. To measure the skewing of the local line segment direction, the direction uncertainty is defined as,

$$U_3(S) = \sum_{p \in S} \|\theta_p - \theta_{mean}\|^2, \tag{9}$$

where the summation is also over all the patches in the node. The object of the splitting process is to minimize either of the three uncertainty measures. Specifically, a binary test T^k is conducted on each node, which minimizes the following objective function,

$$\arg\min_k (U_*(\{(p_i|t^k(I_i)) = 0\}) + U_*(\{(p_i|t^k(I_i)) = 1\})), \tag{10}$$

where $*$ can take the value of 1, 2 or 3. This process is conducted on each tree to decrease the class label uncertainty, offset uncertainty and direction skewing. The training process is depicted in Fig. 4.

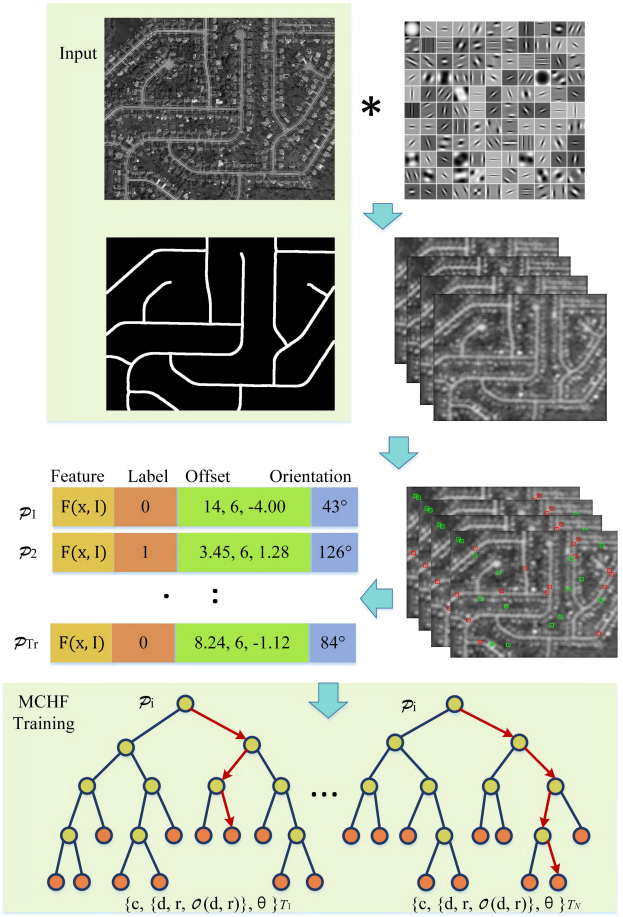


FIGURE 4. The flowchart of the training process.

The training samples are splitted as the tree grows until the tree reaches a maximum depth or the number of samples within the group is no more than the minimum number. Once the samples reach a leaf node, the statistical information of this node is calculated and stored. The information includes a class label proportion probability $p(c|\mathcal{L})$, the sample offsets $D = \{d_i|i = 1, 2, \ldots, M_\mathcal{L}\}$ and offset measures $O_\mathcal{L} = \{\mathcal{O}(d_i, r_i)|i = 1, 2, \ldots, M_\mathcal{L}\}$ and respective orientations $A_\mathcal{L} = \{\theta_i|i = 1, 2, \ldots, M_\mathcal{L}\}$, where $M_\mathcal{L}$ is the number of the patches in leaf node $\mathcal{L}$. For precisely estimating spatial distribution of the sample patches in this node, the radius set for the nearest local tubulars, $R_\mathcal{L} = \{r_i|i = 1, 2, \ldots, M_\mathcal{L}\}$, is also stored. In fact, the samples in each leaf node have some common characteristics. Take a leaf node with a high probability of positive label as an example, the patches of this node tend to be the different tubular segment parts which have similar orientations, appearances, widths and offsets in the whole curvilinear structure.

D. DETECTION

In a test image or FOV, each point x is considered as a patch center, and the corresponding feature vector $f(x, I)$ which represents local appearance can be extracted by convolving with the same filter bank in the sample construction of the training process. All samples are passed through every tree and fall into leaf nodes in MCHF. And such samples are used to cast votes for their local spatial center in Hough space. As mentioned in Section III-A, a whole curvilinear structure is composed of a set of individual curves, and each individual curve is composed of a set of tubular segments. According to local linearity, a tubular segment $\mathcal{S}(o, r, l, \theta)$ can be regarded as a rectangle. It means that $\mathcal{S}$ is bounded in a box of size $2rl$ (width r and length l) and orientation θ, where we assume that the width r is fixed and the length l is very small. Thus, regardless of the orientation, the only parameter crucial for defining a tubular segment (or bounding box) is the location of its center. What's more, since each patch has a closest centerline point, each sample patch has a corresponding object bounding box.

Considering a patch $\mathcal{P} = \{F(x, I), c, \mathcal{O}(d, r), \theta\}$ centered at x in the test image, where $F(x, I)$ denotes the feature vector, c is the hidden class label, $\mathcal{O}(d, r)$ is the offset against the hidden local centerline point, and θ is the hidden object orientation. Our primary concern is the conditional probability $p(E(y)|f(x, I))$, where $E(y)$ denotes the random event of the local tubular segment object (bounding box) centered at y, with the condition that the patch feature $f(x, I)$ exists. As [10] and [11] mentioned, this probability can be expressed as,

$$\begin{aligned} p(E(y)|f(x,I)) &= p(E(y), c=1|f(x,I)) \\ &= p(E(y)|c=1, f(x,I)) \cdot p(c=1|f(x,I)) \\ &= p(\mathcal{O}(d,r)|c=1, f(x,I)) \cdot p(c=1|f(x,I)), \end{aligned} \tag{11}$$

where $p(E(y)|f(x, I))$ is factorized into two terms, both of which can be estimated through the information stored at the leaf nodes $\mathcal{L}$ in MCHF. The first item can be approximated using the Parzen-window with the offset vector set $D_{\mathcal{L}}$ which is collected from the training leafs. And the second item is estimated to be the proportion of the object patches, denoted by $C_{\mathcal{L}}$). For a single tree, the probability is expressed as,

$$\begin{aligned} &p(E(y)|f(x,I);\mathcal{L}) \\ &= \frac{p(c|\mathcal{L})}{|D^c_{\mathcal{L}}|} \sum_{d \in D^c_{\mathcal{L}}} \frac{1}{2\pi\sigma^2} \exp\left(\frac{\|(\mathcal{O}(d,r) - \mathcal{O}(d_{mean}, r_{mean}))\|^2}{2\sigma^2}\right), \end{aligned} \tag{12}$$

where σ^2 denotes the variance of the Gaussian Parzen-window, $|D^c_{\mathcal{L}}|$ is the number of the offsets in $D^c_{\mathcal{L}}$. For the entire MCHF $\{\mathcal{T}_t\}_{t=1}^T$, the probability is averaged over all trees,

$$p(E(y)|f(x,I);\{\mathcal{T}_t\}_{t=1}^T) = \frac{1}{T}\sum_{t=1}^{T} p(E(y)|f(x,I);\{\mathcal{T}_t\}). \tag{13}$$

Eq. (13) represents the probabilistic votes of a single patch existing in the curvilinear structure. In order to aggregate the votes cast from different patches, the votes can be accumulated in the Hough space $V(y)$ through each pixel location y, i.e.,

$$V(y) = \sum_{x \in S} p(E(y)|f(x,I);\{\mathcal{T}_t\}_{t=1}^T). \tag{14}$$

To make the computation efficiently, we also employ the strategy used in [10], i.e., passing every sample $f(x, I)$ to each tree in Hough Forest, and accumulating the value $\frac{C_{\mathcal{L}}}{|D_{\mathcal{L}}|}$ to all pixels $\{\mathcal{O}(d, r)|\mathcal{O}(d, r) \in D_{\mathcal{L}}\}$ on each Hough image $V(y)$. Considering the various widths of the curvilinear structure, we extend the original HF into a multi-scale version, MCHF. To handle different scales, we employ a range of scales $\{r_1, r_2, \ldots, r_Q\}$, where Q is the number of scale factors. Thus each leaf node in the MCHF votes in all voting spaces to get the offset measures in re-scaled space. A final voting map can be constructed by accumulating the multi-scale offsets, which are weighted by the object probability of each leaf node, and Gaussian filtering. The value of a special location in generalized Hough space is directly proportional to the probability of a local object centered here. The local peak values in the voting space can be regarded as the most probable local tubular segment centers.

In the classical HF, an ordinary object center can be readily located through the non-maximum suppression operation in the voting space, but a curvilinear object has a large number of connective center points, thus it is impracticable to get the centerline point through the ordinary non-maximum suppression directly. In our model, additional information is stored on each leaf node during training, including offsets $D = \{d_i|i = 1, 2, \ldots, M_{\mathcal{L}}\}$, offset measures $O_{\mathcal{L}} = \{\mathcal{O}(d_i, r_i)|i = 1, 2, \ldots, M_{\mathcal{L}}\}$, their nearest tubular segment radii $R_{\mathcal{L}} = \{r_i|i = 1, 2, \ldots, M_{\mathcal{L}}\}$, and orientations $A_{\mathcal{L}} = \{\theta_i|i = 1, 2, \ldots, M_{\mathcal{L}}\}$. The properties of a test patch in a leaf node can be approximately expressed through the mean values of these properties. While a patch traverses all the trees in MCHF, the corresponding mean value of the properties can be used to represent the patch properties, denoted by d_{mean}, $\mathcal{O}(d_{mean}, r_{mean})$, r_{mean} and θ_{mean}, respectively. Therefore, a whole test image can be mapped into four additional spaces. In order to specify the centerline, we just consider the patches which are located in curvilinear structure with high probability, i.e., $\mathcal{O}(d_{mean}, r_{mean}) \geq 0$. Then we use the directional non-maximum suppression to get the centerline points. The directional non-maximum suppression is akin to classical non-maximum suppression. The only difference is that the former just compares the maximum value along the profile direction, while the latter considers the maximum value of the whole local patch. Once the location of a centerline point o is determined, the only problem is to estimate the width and orientation of the local tubular segment (the length of the tubular is very small, which is usually set as 1). The radius r_{mean} can be approximated as half the width of the tubular segment, and the orientation can be approximated by the mean

orientation θ_{mean}. Thus the location, shape and orientation of the tubular segment are determined. The whole curvilinear structure is the concatenation of all of these tubular structures. The detection process is shown in Fig. 5.

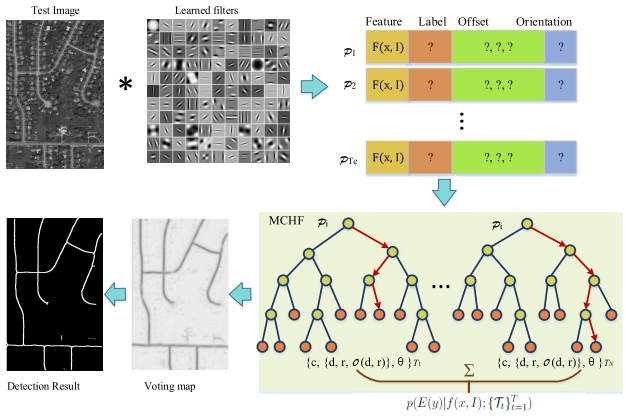


FIGURE 5. The flowchart of the detection process.

IV. EXPERIMENTS

A. DATASETS AND CONFIGURATION

To evaluate the proposed method, we have conducted a series of experiments on several tasks of curvilinear structure detection, including retinal vessel segmentation, aerial road extraction and 2D neuronal delineation.

1) THE RETINAL VESSEL IMAGES

Retinal vessel images consist of two widely used benchmark datasets, namely STARE [33] and DRIVE [2]. The DRIVE dataset includes 40 retinal images, half of which are used for training and others for test. Each image in the dataset has two manually segmented results of vessels as well as masks that delineate the FOV. The STARE dataset consists of 20 color images of retinal fundus, half of which contain signs of pathologies, while others are healthy. This dataset also provides two groups of manually segmented images. Generally, in both the two datasets, the first observer's manual labelling is considered as the ground truth for assessment, while the segmentation results of the second observer are used as the comparison benchmark. In our experiment, we use the green channel of the image of the two datasets because it has the highest contrast [34]. We preprocess the data as [6] suggested, i.e., removing the central light reflex, homogenizing the background of the FOV, and whitening the images as the classic sparse presentation procedure [30]. Our filter bank has 121 filters and each filter size is 21 × 21, following the settings in [30]. We use 3×10^5 positive and 3×10^5 negative examples for training and validation for both the two datasets respectively. Unlike the standard Hough forest, which only trains a small amount of regression trees, here we train 300 trees to construct the forest, where each tree has a maximum depth of 6, and the minimum number of samples stored at a leaf node (N_{min}) is equal to 30. However, these two datasets have no information of the centerline location, thus we need to skeletonize the ground truth images to get approximate locations.

2) THE 2D NEURONAL IMAGES

The microscopic neuronal image dataset was collected by IMB and IMCB of A*star [9], [22]. This dataset includes 210 confocal fluorescence images of mouse embryonic neural stem cells and corresponding labels. Neurons are stained for neurite channel (GFP) marker and for nuclei channel (DAPI) marker, respectively. The dateset is composed by 2 batches, namely NeuB1 and NeuB2, including 112 images and 98 images respectively. Images of the second batch have more noticeable noise in the background and are more blurred in the foreground. In our experiment, for both NeuB1 and NeuB2, we select 1/3 of the images for training and the rest 2/3 for test as in [9]. We set the learned filter sized $s = 21 \times 21$. For training a forest, we extract 1×10^5 positive samples and the same number of negative samples. To construct Hough forest, we use 200 trees with the maximum depth of 6, and $N_{min} = 30$.

3) THE AERIAL ROAD IMAGES

This dataset contains 26 color aerial images of road networks in rural area [28]. We also use half of them as training set and the other half as test set. The ground truth of each image is a list of centerline points with their corresponding radii. Before the experiments, we preprocess the dataset as Sironi *et al*. suggested [27], i.e., discretizing the coordinates of centerline points into integers, and converting the swc file into a binary image file according to the centerline locations and their corresponding radii. There are 220 trees in the Hough forest, while other settings are the same as the experiments for 2D Neuronal images.

For all the experiments, the Gaussian kernel size $\sigma^2 = 9$ is adopted, and for multi-scale setting, the scale dimension is filtered with $\sigma^2 = 1$, which follows Gall's setting [11]. The scale range R is $\{r_1, r_2, \ldots, r_M\}$, which varies for different datasets: on DRIVE and STARE, $M = 4$ and $R = \{1, 2, 4, 4\sqrt{2}\}$; on neuronal dataset, $M = 4$ and $R = \{1, 2, 4, 4\sqrt{2}\}$; on aerial road dataset, there is only one scale, i.e. $M = 1$, and $R = 6$.

B. EVALUATION CRITERIA

To evaluate the curvilinear structure detection results, multiple criteria are employed, including sensitivity, specificity and accuracy, as defined in Eq. (15), (16) and (17), respectively,

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \tag{15}$$

$$\text{Specificity} = \frac{TN}{TN + FP}, \tag{16}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP}, \tag{17}$$

where true positive (TP) is the number of curvilinear points correctly detected; false positive (FP) is the number of non-curvilinear points detected as curvilinear points; true negative (TN) is the number of non-curvilinear points that are not detected as curvilinear points; and false negative (FN) is the number of curvilinear points that are not detected.

C. EXPERIMENTAL RESULTS AND COMPARISON

In this section, we show both visual and statistical detection results on the above-mentioned datasets respectively. To access the performance of MCHF, we compare it with the state-of-the-art curvilinear structure detection methods, including Eigen [15], Optimally Oriented Flux (OOF) [17], Kernel Boost [29], B-COSFIRE [35], and LSF [22] as well as its modified version [23]. Eigen uses eigenvalues and eigenvectors of multi-scale Hessian to identify the local point position and orientation of curvilinear structure [15]. Optimally Oriented Flux (OOF) is a widely used detector of curvilinear structures, which uses the second derivatives of an image convolved with the intensity indicator function of a local circular area or spherical volume in the image [17]. Kernel Boost learns discriminative convolutional filters from images based on the gradient boosting framework [29]. B-COSFIRE extracts curvilinear structure through calculating the weighted geometric mean of the Gabor filters by applying a pool of Difference-of-Gaussians filters [35]. LSF learns structured and contextual features with tree structured classifiers to segment the curvilinear structures [22]. And the modified version, SF+dist, incorporates local spatial label patterns into the feature space [23].

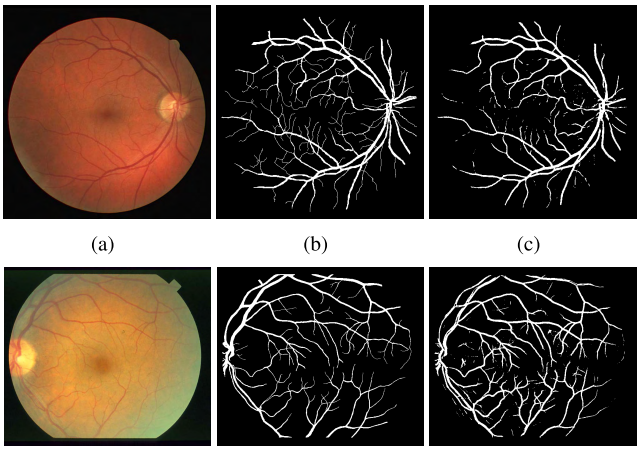


FIGURE 6. Detection results on retinal vessel images. (a) and (d) are two original retinal images in DRIVE and STARE respectively, (b) and (e) are the manual segmentation results of the first observer, and (c) and (f) are the detection results.

We provide the visualized results in Figs. 6-8. Fig. 6 shows the segmentation results of retinal images on DRIVE and STARE datasets. As can be observed, the detection results are highly consistent with the manual segmentation results. Fig. 7 shows two examples of the detection results of neuronal

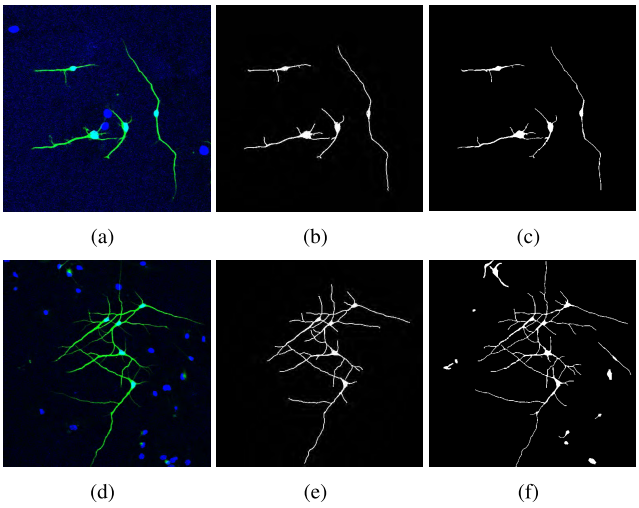


FIGURE 7. Detection results on 2D neuron images. (a) and (d) are the original neuron images, (b) and (e) are the ground truth of the nervus, and (c) and (f) are the detection results.

images from NeuB1 and NeuB2, respectively. Fig. 8 shows an example of the delineation on aerial road image, where the test results of our method are almost consistent with the ground truth from the observer's point of view.

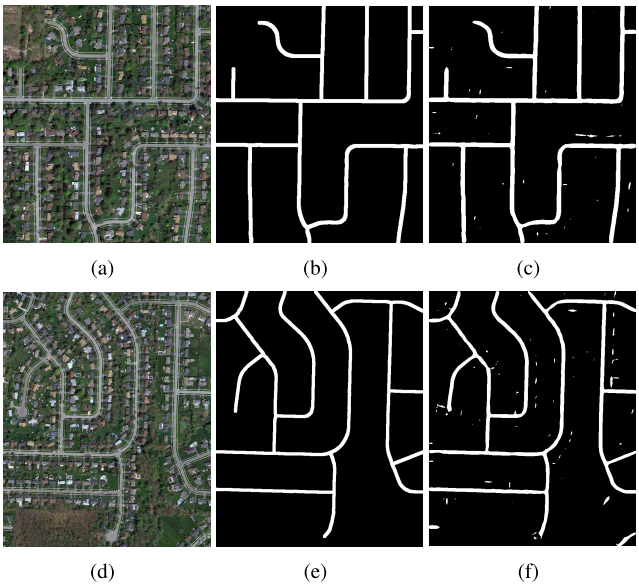


FIGURE 8. Detection results on aerial road images. (a) and (d) are the original images, (b) and (e) are the ground truth of the roads, and (c) and (f) are the detection results.

Apparently, most of the curvilinear structures can be accurately detected, but some of the detection results are susceptible to small targets. In Fig. 8, it can be found that the detected results are not exactly matched to the ground truth roads. This means that our approach still suffers interference of some small objects such as houses. There may be two reasons for this deviation. First, it is difficult to distinguish between the local curvilinear segments and some of the house parts. Some of the local curvilinear segments and house parts

have similar feature vectors representing the appearance. And they even have similar relative position relationship. Second, the curvilinear structure region and background region holds different proportions in the image. The negative sample has a variety of appearances, and the area of background is much larger than the area of the curvilinear object. Yet during the training process, we have selected the same number of positive and negative samples. This setting may lead to the insufficient number of negative samples during training, and further increase the detection error in the process of Hough forest testing.

TABLE 1. Detection accuracies of 7 methods on DRIVE dataset.

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)
2nd manual[a]	94.70	77.63	97.23
Eigen	92.32	64.82	95.43
OOF	90.57	68.42	94.81
KernelBoost	90.95	78.30	95.50
B-COSFIRE	94.66	78.67	96.93
LSF	95.27	75.95	97.11
SF+dist	95.54	77.33	97.28
MCHF	**95.61**	**80.71**	**97.77**

[a] 2nd manual denotes the manual segmentation results of the second observer provided by the dataset

TABLE 2. Detection accuracies of 7 methods on STARE dataset.

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)
2nd manual[a]	93.48	89.51	93.84
Eigen	94.62	66.10	96.67
OOF	94.08	68.98	96.63
KernelBoost	95.93	75.94	97.60
B-COSFIRE	95.12	79.18	97.36
LSF	95.32	77.91	97.41
SF+dist	96.00	**79.24**	97.74
MCHF	**96.14**	77.96	**98.23**

[a] 2nd manual denotes the manual segmentation results of the second observer provided by the dataset

TABLE 3. Detection accuracies of 7 methods on 2D neuron images.

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)
Eigen	96.49	72.57	98.03
OOF	96.31	69.95	98.01
KernelBoost	98.12	78.41	98.61
B-COSFIRE	97.83	77.32	99.16
LSF	98.30	83.71	99.18
SF+dist	**98.38**	**85.28**	**99.23**
MCHF	97.40	77.96	98.83

Besides, Tables 1-4 present the statistical results of different detectors on the four data sets, respectively. The reported accuracies are averaged over all images. For the retinal vessel images, MCHF generally achieves the best performance on both DRIVE and STARE dataset among all the 7 methods. Especially, MCHF enhances the sensitivity more significantly than other two criteria, indicating that it can

TABLE 4. Detection accuracies of 7 methods on aerial road images.

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)
Eigen	54.38	32.18	55.69
OOF	57.64	38.93	58.74
KernelBoost	**91.86**	79.15	**92.61**
B-COSFIRE	80.37	77.32	89.02
LSF	85.73	78.19	88.18
SF+dist	88.16	80.19	88.63
MCHF	89.95	**81.56**	90.44

effectively detect more curvilinear structure points. For the neuron images, KernelBoost and the structure feature based methods obtained better results than other methods; while for the aerial road images, MCHF and KernelBoost rank top two among all the methods. Compared with KernelBoost, MCHF has better sensitivity (81.56% vs. 79.15%) but worse accuracy and specificity (∼2% lower).

From these tables, we can draw the following conclusions. First, performance of the intuitive detection approaches is much worse than the learning-based methods. It is because that Eigen and OOF algorithms just analyze the local profile property of curvilinear structure, both of which are implemented by solving the eigenvalues of elliptic equations. In a real image, such analytical models often encounter a considerable interference in the heterogeneous background. Furthermore, the curve structure itself is complex and cannot be completely simulated by these analytical models. For instance, there is a significant difference on the orientation flux property between the junction and the ordinary curvilinear local region. If the eigenvalues and eigenvectors are calculated, it is hard to determine whether the junction region belongs to the curvilinear structure. These shortcomings can be effectively avoided by learning-based methods. These methods avoid the direct analysis of local characteristics and detect the curvilinear object through the learned classifier or the retractor. Second, our approach is superior to most other learning-based methods. It is partly because that most of the existing learning-based methods tend to focus solely on the local appearance other than the spatial distribution patterns of curvilinear structure. Thus, they just consider the curvilinear structure as a common classification problem, and discriminate the foreground and background. We note that [23] also takes the context distance into consideration, and improves the performance, which treats the context distance as a group of feature vectors for identifying the curvilinear structure. Our approach also takes advantage of the spatial relationship of local part in training and voting. Compared with the structural feature learning-based methods, MCHF performs better on three datasets. These results demonstrate the advantages of MCHF in identifying complex curvilinear structures in various types of images. In addition, the MCHF method performs slightly worse than Kernel Boost and structural feature learning-based methods in neural image dataset. The visual results show that a small number of the scattered cell nuclei were detected as linear structures. Errors may

occur during training. The nucleus of a neuron is a part of the whole curvilinear structure. Obviously, it's a blob-like local object, while we consider it as the local tubular segment in the training stage. This process results in a blob-like structure in the MCHF leaf nodes after learning. Then the scattered nucleus centers voted as a high probability centerline points of curvilinear structure. By contrast, the two best existing method, Kernel Boost and structural feature learning-based methods do not divided the whole curvilinear object into many parts. These treatments can effectively avoid the error of which sees the scattered nuclei as the curvilinear structure.

D. DISCUSSION

In the field of image detection, an object is usually considered as a combination of many different parts arranged with a certain spatial position relationship. Based on this idea, the generalized Hough transform (GHT) [36] detects an arbitrary-shaped object through a mapping which consists of the relative position between an edge point and an object reference point. The implicit shape model (ISM) [37] considers the local appearance parts as a visual codebook, then uses it to index votes of object position in Hough space. The classical Hough forest (HF) [10], [11] originates from ISM, which vots the object center location through a random forest framework [38], and determines the whole object through the displacement of local object parts (codebook). In the classical HF model, a visual object has an explicit center (reference point), and the whole object can be restricted into a bounding box. Each part cast probabilistic votes for possible positions of the object center. Accordingly, the final object center is equal to the maxima of the Hough image which accumulates all of the votes.

However, because of the difference between curvilinear object and ordinary foreground, classical HF method for object detection is not directly applicable to the extraction of curvilinear structures. First, it is hard to determine the bounding box center or the location and size of the bounding box, thus the displacement/offset defined in the classic HF method is meaningless. Second, because the curvilinear structure has limited width, the local patch often exceeds the local object region. Hence the appearance of a local patch often includes both curvilinear and background factor. The proposed MCHF method aims to address the curvilinear detection problems, and it is significantly different from the classical HF. More over, the object and its center are one to one correspondence in the classical HF; while in MCHF, the relationship of curvilinear structure and its centers are one to many mapping. In addition, the sample composition, training and detection are also unlike classical HF method.

V. CONCLUSION

In this paper, we propose a new method for the detection of curvilinear structure in images. The fundamental idea of this framework is regarding the curvilinear structure as a special object, which has multiple object centers, and each centerline point can be considered as a local center of such object. Based on this idea, we extend the original Hough forest into a multi-centered version specialized for curvilinear structure detection. The MCHF model generates the voting map in Hough space through accumulating the offset of each leaf node of all the trees. For constructing the MCHF model, each sample is represented by a quadri-tuple, including feature vector, class label, offset factor and orientation. By using directional non-maximum suppression, each center point of a tubular segment is determined, and the whole curvilinear structure can be detected through its width and orientation estimation. We perform our method on several publicly available datasets and compare with the state-of-the-art methods. Experimental results demonstrate that the proposed method is capable to detect various curvilinear or tubular structures, and achieves competitive performance in different types of images.

ACKNOWLEDGEMENT

(Hanjin Zhang and Yang Yang contributed equally to this work.)

REFERENCES

[1] S. Berlemont and J. C. Olivo-Marin, "Combining local filtering and multiscale analysis for edge, ridge, and curvilinear objects detection," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 74–84, Jan. 2010.

[2] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Trans. Med. Imag.*, vol. 23, no. 4, pp. 501–509, Apr. 2004.

[3] M. Sofka and C. V. Stewart, "Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures," *IEEE Trans. Med. Imag.*, vol. 25, no. 12, pp. 1531–1546, Dec. 2006.

[4] M. S. Hassouna and A. A. Farag, "Robust centerline extraction framework using level sets," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 458–465.

[5] J. V. D. Weijer, L. J. V. Vliet, P. W. Verbeek, and R. V. Ginkel, "Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 9, pp. 1035–1042, Sep. 2001.

[6] D. Marin, A. Aquino, M. E. Gegúndez-Arias, and J. M. Bravo, "A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features," *IEEE Trans. Med. Imag.*, vol. 30, no. 1, pp. 146–158, Jan. 2011.

[7] E. Türetken, F. Benmansour, B. Andres, P. Głowacki, H. Pfister, and P. Fua, "Reconstructing curvilinear networks using path classifiers and integer programming," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 12, pp. 2515–2530, Dec. 2016.

[8] D. Chai, W. Förstner, and F. Lafarge, "Recovering line-networks in images by junction-point processes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1894–1901.

[9] J. De *et al.*, "A graph-theoretical approach for tracing filamentary structures in neuronal and retinal images," *IEEE Trans. Med. Imag.*, vol. 35, no. 1, pp. 257–272, Jan. 2016.

[10] J. Gall and V. Lempitsky, "Class-specific Hough forests for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 1022–1029.

[11] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2188–2202, Nov. 2011.

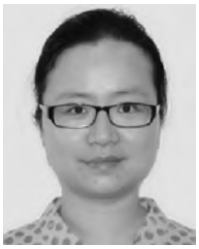
[12] Y. Shang *et al.*, "Vascular active contour for vessel tree segmentation," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 4, pp. 1023–1032, Apr. 2011.

[13] U. T. V. Nguyen, A. Bhuiyan, L. A. F. Park, and K. Ramamohanarao, "An effective retinal blood vessel segmentation method using multi-scale line detection," *Pattern Recognit.*, vol. 46, no. 3, pp. 703–715, 2013.

[14] C. Steger, "An unbiased detector of curvilinear structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 2, pp. 113–125, Feb. 1998.

[15] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 1998, pp. 130–137.
[16] C. Steger, "Unbiased extraction of lines with parabolic and Gaussian profiles," *Comput. Vis. Image Understand.*, vol. 117, no. 2, pp. 97–112, 2013.
[17] M. W. Law and A. C. Chung, "Three dimensional curvilinear structure detection using optimally oriented flux," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 368–382.
[18] M. W. K. Law and A. C. S. Chung, "An oriented flux symmetry based active contour model for three dimensional vessel segmentation," in *Proc. Eur. Conf. Comput. Vis. Conf. Comput. Vis.*, 2010, pp. 720–734.
[19] E. Turetken, C. Becker, P. Glowacki, F. Benmansour, and P. Fua, "Detecting irregular curvilinear structures in gray scale and color imagery using multi-directional oriented flux," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2014, pp. 1553–1560.
[20] C.-K. Tang and G. Medioni, "Curvature-augmented tensor voting for shape inference from noisy 3D data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 858–864, Jun. 2002.
[21] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 3, pp. 266–280, Mar. 2000.
[22] L. Gu and L. Cheng, "Learning to boost filamentary structure segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 639–647.
[23] G. Lin, X. Zhang, Z. He, H. Li, and C. Li, "Segment 2D and 3D filaments by learning structured and contextual features," *IEEE Trans. Med. Imag.*, vol. 36, no. 2, pp. 596–606, Feb. 2017.
[24] M. Berks, Z. Chen, S. Astley, and C. Taylor, "Detecting and classifying linear structures in mammograms using random forests," in *Information Processing in Medical Imaging*, vol. 22. Berlin, Germany: Springer, 2011, pp. 510–524.
[25] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler, "A higher-order CRF model for road network extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1698–1705.
[26] C. Becker, R. Rigamonti, V. Lepetit, and P. Fua, "Supervised feature learning for curvilinear structure segmentation," in *Medical Image Computing and Computer-Assisted Intervention*. Berlin, Germany: Springer, 2013.
[27] A. Sironi, V. Lepetit, and P. Fua, "Multiscale centerline detection by learning a scale-space distance transform," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2697–2704.
[28] E. Turetken, F. Benmansour, and P. Fua, "Automated reconstruction of tree structures using path classifiers and mixed integer programming," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 566–573.
[29] R. Rigamonti and V. Lepetit, "Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2012, pp. 189–197.
[30] R. Rigamonti, A. Sironi, V. Lepetit, and P. Fua, "Learning separable filters," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2754–2761.
[31] M. Schneider, S. Hirsch, B. Weber, G. Székely, and B. H. Menze, "Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters," *Med. Image Anal.*, vol. 19, no. 1, pp. 220–249, 2015.
[32] A. Sironi, E. Türetken, V. Lepetit, and P. Fua, "Multiscale centerline detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1327–1341, Jul. 2016.
[33] A. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Trans. Med. Imag.*, vol. 19, no. 3, pp. 203–210, Mar. 2000.
[34] X. You, Q. Peng, Y. Yuan, Y.-M. Cheung, and J. Lei, "Segmentation of retinal blood vessels using the radial projection and semi-supervised approach," *Pattern Recognit.*, vol. 44, nos. 10–11, pp. 2314–2324, 2011.
[35] G. Azzopardi, N. Strisciuglio, M. Vento, and N. Petkov, "Trainable COSFIRE filters for vessel delineation with application to retinal images," *Med. Image Anal.*, vol. 19, no. 1, pp. 46–57, Jan. 2015.
[36] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, no. 2, pp. 111–122, 1981.
[37] B. Leibe, "Combined object categorization and segmentation with an implicit shape model," in *Proc. Workshop Statist. Learn. Comput. Vis. (ECCV)*, Prague, Czech Republic, 2004, pp. 17–32.
[38] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

HANJIN ZHANG received the M.S. degree in control science and engineering from the University of Shanghai for Science and Technology in 2012. He is currently pursuing the Ph.D. degree with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University. His research interests include image analysis, object detection, and recognition.



YANG YANG received the Ph.D. degree in computer science from Shanghai Jiao Tong University, China, in 2009. She was a Visiting Scholar with the University of California at Riverside from 2012 to 2013. She is currently an Associate Professor with the Department of Computer Science, Shanghai Jiao Tong University. Her research interests focus on machine learning and bioinformatics.

HONGBIN SHEN received the Ph.D. degree from Shanghai Jiao Tong University, China, in 2007. He was a Post-Doctoral Research Fellow with the Harvard Medical School from 2007 to 2008, and a Visiting Professor with the University of Michigan in 2012. He is currently a Professor with the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University. He has published over 100 journal papers and constructed 35 bioinformatics servers in these areas. His research interests include pattern recognition and bioinformatics. He is an ESI Highly Cited Researcher in 2014 and 2015. He is an Associate Editor of *BMC Bioinformatics*. He serves as an editorial member of several international journals.