对于输入的图像，首先调整其尺寸为512×512。然后，使用Canny边缘检测算法识别stress fiber的边缘。具体地，应用高斯滤波平滑图像，去除噪声。然后，计算图像的梯度，得到可能的边缘。接着，对得到的边缘应用非极大值抑制（Non-Maximum Suppression，NMS）来消除边误检，并通过双阈值的方法筛选边缘信息，其中，双阈值基于统计特征得到。具体地，通过Sobel算子计算水平和垂直方向的梯度和，根据这两个梯度计算梯度幅值用于表示每个像素处的梯度强度，即：

之后，由梯度幅值G进一步计算梯度的均值M和标准差S，并根据这两个值来确定双阈值中的阈值上限，而阈值下限通过缩减比𝒯控制，公式定义如下：

同时，我们将调整后的图像转换到HSV空间，并根据设定蓝色区域的HSV范围在图像中确定蓝色区域的中心点。然后，计算中心点到每一个stress fiber边缘的距离以及每一个stress fiber边缘与水平线的夹角（0°<A<180°）。对于一张图像中存在多个蓝色区域的图像，需要计算每一个蓝色区域的中心到stress fiber边缘的距离，然后对距离取平均值作为一个stress fiber到中心点的距离。在这一过程中，去除了面积较小的蓝色区域以及长度较短的stress fiber边缘。

最后，通过空间离散度量化混乱程度。具体地，将（，）定义为二维空间中的一个点。然后，将二维空间划分为n个子空间，并计算各个子空间的点的个数占总点数的比例，将其作为落入该子空间的概率p。接着通过计算信息熵etp并除以一个和维度相关的值来消除维度数对信息熵的影响，得到的值就是空间离散度*dsp*，公式定义如下：

etp 衡量点分布在空间子区域中的不确定性，p代表点的数量占总点数的比例。n为子区域的总数，通过被最大熵除，将ETP归一化为0~1的标度，便于不同区域间的比较。

We resize the input image to 512×512 pixels. Then, the standard Canny edge detection algorithm [1] is employed to outline the edges of stress fibers. Technically speaking, a Gaussian filter is applied to smoothing the image whilst reducing noise. Afterwards, image gradients are computed so that potential edges can be identified.

The detected edges are then processed using a Non-Maximum Suppression (NMS) technique [2] to eliminate false detections, followed by edge screening via a dual-threshold approach of Canny edge detection [1]. The dual thresholds are determined, based on statistical characteristics of image pixels. To be specific, the standard Sobel operator [3] is used to calculate horizontal and vertical gradients ( and ). The gradient magnitude G at each pixel, representing gradient intensity, is computed as:

$$G = \sqrt{G\_{x}^{2} + G\_{y}^{2}}$$

The mean (M) and standard deviation (S) of the gradient magnitudes are then calculated. The upper threshold () and the lower threshold () are determined as follows, where 𝒯 denotes the reduction ratio:

$$T\_{\max} = M + 2 \times S$$

$$T\_{\min} = \frac{T\_{\max}}{\mathcal{T}}$$

Next, the resized image is converted to the counterpart in the HSV color space. The center point of the blue region is identified based on a predefined HSV range. For each stress fiber edge, two parameters are computed:

1. The distance () from the center of the blue region to the edge.

2. The angle () between the edge and the horizontal axis (0° < Aⁱ < 180°).

For the images containing multiple blue regions, the average distance () from each stress fiber edge to all the blue centers is calculated. During this process, small blue regions and short stress fiber edges are filtered out.

Finally, spatial dispersion (\*dsp\*) is used to measure the degree of disorder. Each pair (, ) is treated as a point in the 2D space. The space is divided into \*n\* subspaces, and the probability \*p\* of points falling into each subspace is computed as the ratio of points in that specific subspace against the total number of the points. The entropy (\*etp\*) and normalized spatial dispersion (\*dsp\*) are derived as follows:

where etp is used to measure the uncertainty of the point distribution within spatial sub-regions, where p represents the proportion of the points in each sub-region relative to the total number of the points. n denotes the total number of the sub-regions. By dividing ETP by the maximum entropy (log₂n), the result is normalized to a scale of [0,1], facilitating the comparisons across different regions.

$$etp = \sum \left( -p \times \log(p) \right)$$

$$dsp = \frac{etp}{- \log\left(\frac{1}{n}\right)}$$

1. Maranga, Johnas Omanwa, Justine John Nnko, and Shengwu \*\*ong. "Learned active contours via transformer-based deep convolutional neural network using canny edge detection algorithm." *Signal, Image and Video Processing* 19.3 (2025): 222.
2. Hosang, Jan, Rodrigo Benenson, and Bernt Schiele. "Learning non-maximum suppression." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
3. Kanopoulos, Nick, Nagesh Vasanthavada, and Robert L. Baker. "Design of an image edge detection filter using the Sobel operator." IEEE Journal of solid-state circuits 23.2 (1988): 358-367.

Where is the link of the source code?