水平集分割
主要修改的部分
添加的辅助文件
运行说明
效果展示

水平集分割

主要修改的部分

- **evolution_cv.m**

  水平集演化核心部分

  演化方程：

$$\frac{\partial \phi}{\partial t}=\mu[\triangle \phi-div(\frac{\nabla \phi}{|\nabla \phi|})]+\lambda \delta(\phi)div(g\frac{\nabla \phi}{|\nabla \phi|})+\nu g\delta(\phi) \tag{1}$$

```
function phi = evolution_cv(I, phi0, g, gx, gy, mu, nu, lambda, delta_t, epsilon, numIter)
    I = BoundMirrorExpand(I); % 镜像边缘延拓
    phi = BoundMirrorExpand(phi0);
    [phix, phiy] = gradient(phi);
    phixy = sqrt(phix.^2 + phiy.^2 + 1e-10);
    phix = phix ./ phixy;
    phiy = phiy ./ phixy;

    for k = 1 : numIter
        phi = BoundMirrorEnsure(phi);
        delta_h = Delta(phi,epsilon);
        Curv = curvature(phi);

        % updating the phi function
        distRictTerm = mu * (4 * del2(phi) - Curv);
        lengthTrem = lambda * delta_h .* (phix .* gx + phiy .* gy + g .* Curv);
        areaTerm = nu * g .* delta_h;
        new_term = distRictTerm + lengthTrem + areaTerm;
        phi = phi + delta_t * new_term;
    end
    phi = BoundMirrorShrink(phi); % 去掉延拓的边缘
end
```

- **Delta.m**

  演化方程的$\delta_\varepsilon(x)$函数：

$$\delta_\varepsilon(x)=\begin{cases}0 & |x|>\varepsilon\\ \frac{1}{2\varepsilon}[1+cos\frac{\pi x}{\varepsilon}] & |x|\leqslant\varepsilon\end{cases} \tag{2}$$

```
    function Delta_h = Delta(phi, epsilon)
        Delta_h = (1 / (2 * epsilon)) * (1 + cos(pi * phi / epsilon));
        b = (phi <= epsilon) & (phi >= -epsilon);
        Delta_h = Delta_h .* b;
    end
```

- **calc_g.m**

  演化方程中的g函数：

$$g=\frac{1}{1+|\nabla G_\sigma * I|^2} \tag{3}$$

```
function g = calc_g(I,sigma)
    I = BoundMirrorExpand(I);
    G = fspecial('gaussian', 15, sigma);
    X = conv2(I,G,'same');
    [Ix, Iy] = gradient(X);
    g = 1 ./ (1 + Ix.^2 + Iy.^2);
end
```

添加的辅助文件

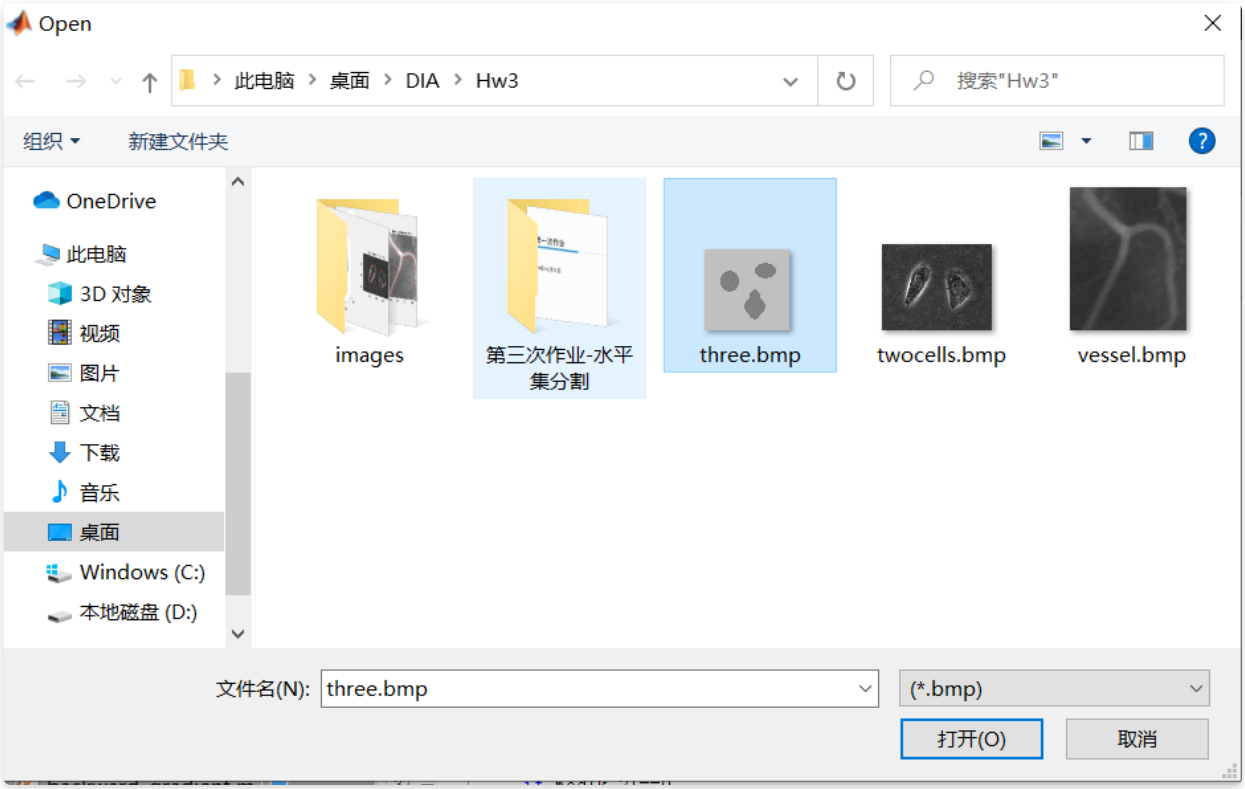
- **initial.m**

  对于三张图，不同的初始化结果带来的效果会有一些差异，因此加入了一个函数，根据待处理的图片选取合适的参数进行初始化。

```
function [phi_0, kk] = initial(f,U)
    [nrow, ncol] = size(U);
    if f(1) == 'v'
        c0 = 3;
        initialLSF = -c0 * ones(size(U));
        initialLSF([1:15, nrow - 15 : nrow], 30 : ncol - 30) = c0;
        initialLSF(15:nrow - 15, [1:5, ncol - 5:ncol]) = c0;
        phi_0 = initialLSF;
        kk = 300;
    else
        c0 = 3;
        initialLSF = c0 * ones(size(U));
        roi = 6;
        initialLSF(roi : nrow - roi, roi : ncol - roi) = -c0;
        phi_0 = initialLSF;
        kk = 160;
    end
end
```

运行说明

运行文件目录中的main函数，将会出现如下图所示界面

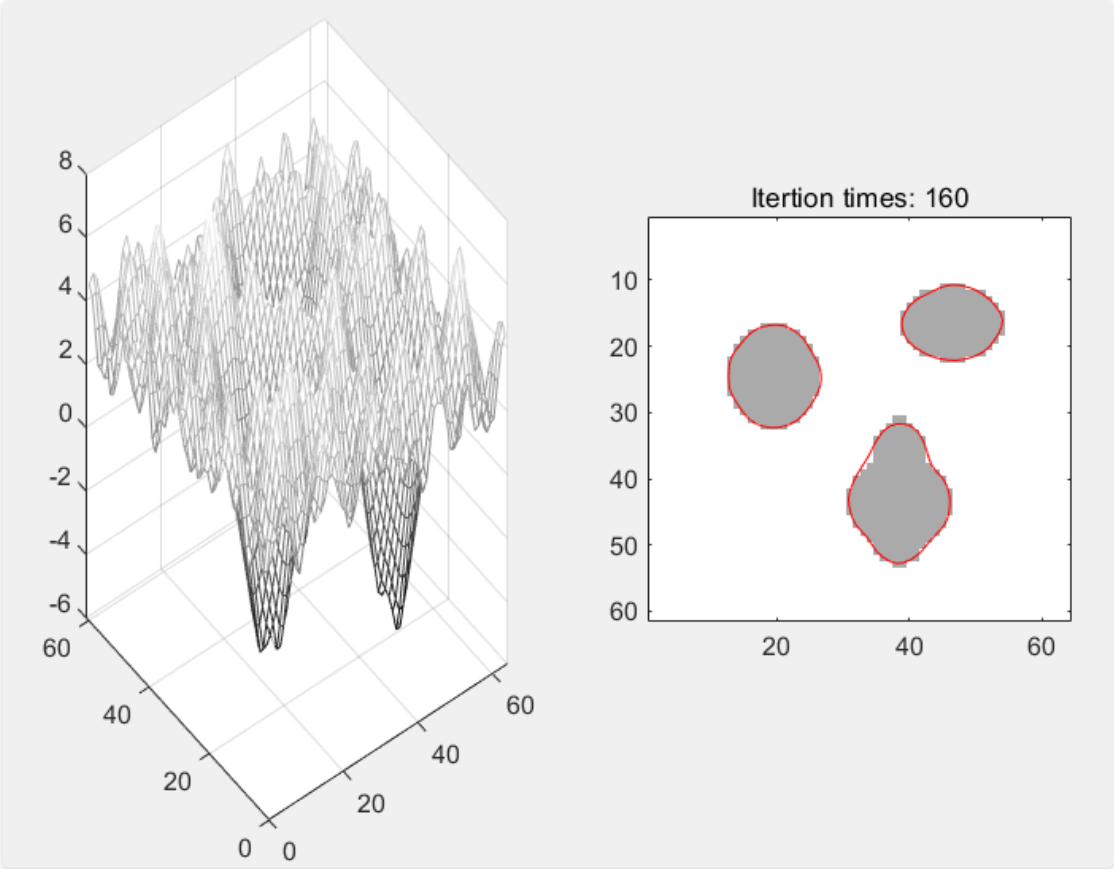


运行界面

随后选择需要进行处理的图片即可。

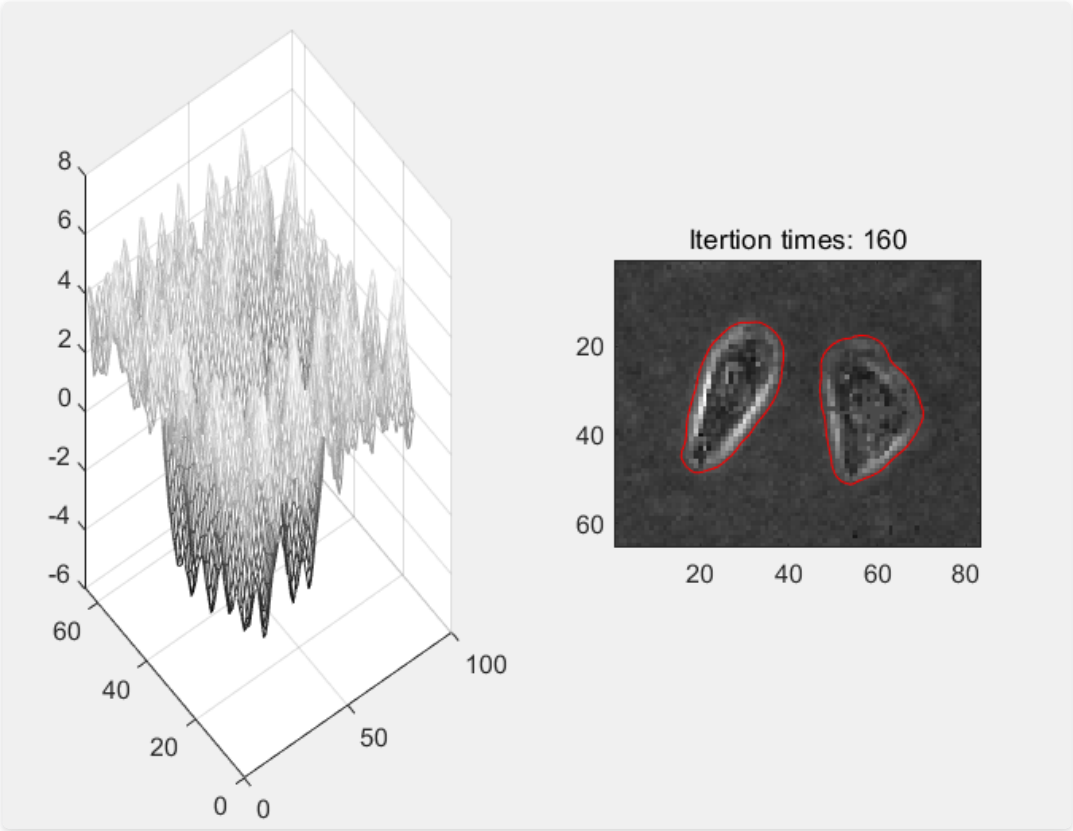
效果展示

- **three.bmp**



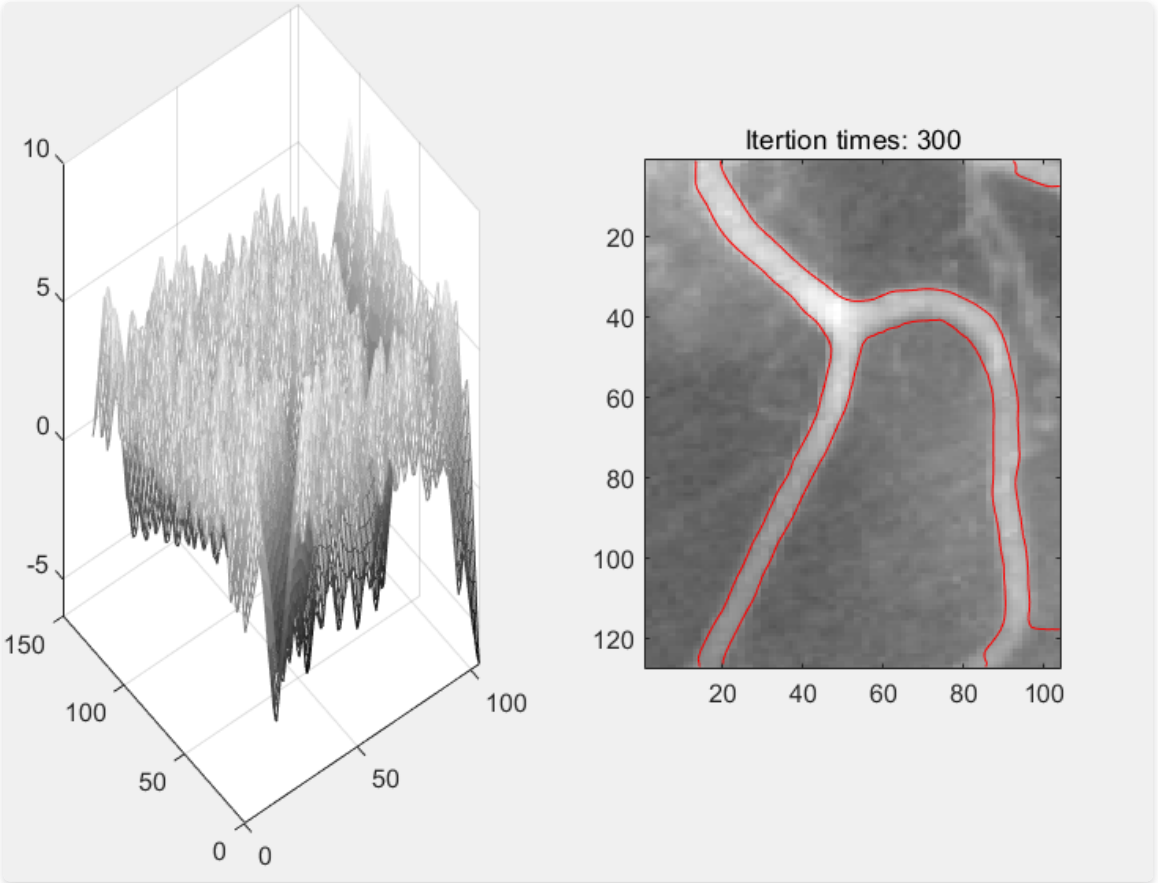
three.bmp处理结果

- **twocells.bmp**



twocells.bmp处理结果

- **vessel.bmp**



vessel.bmp处理结果