```
function [L,N] = myslic(img, k, m, n_turn)
```

```
% img:输入图像
% k:希望分成多少个超像素
% m:取代 Nc 的常数值，一般在[1,40]
% n_turn:迭代几次
%
% L:返回的标签图，值为 1：N
% N：最终的超像素个数
```

所用数据

```
[rows, cols, chan] = size(img);
img = rgb2lab(img);
S = sqrt(rows*cols/k);
S = ceil(S);
row_step = floor(rows/S);
col_step = floor(cols/S);
C = zeros(k,6);        % 1:3 mean Lab value; 4:5 x,y; 6 Num of Pixels
L = -ones(rows, cols);
d = inf(rows, cols);
```

```
输入参数的数目不足。

出错 myslic (line 10)

[rows， cols， chan] = size(img);
```

初始化中心点(没有按论文所说找 3*3 里梯度最小地方，作用不大，第一次迭代更新就把这个效果抵消很多)

```
kk = 1;
for ii = 1:row_step
    for jj = 1:col_step
        rowc = round(S*(ii-0.5));
        colc = round(S*(jj-0.5));
        C(kk,1:3) = img(rowc,colc,:);
        C(kk,4) = rowc;
        C(kk,5) = colc;
        C(kk,6) = 0;
        kk = kk + 1;
    end
end
k = kk - 1;
```

开始迭代（一般不超过 10 次）

```
for n = 1:n_turn
    % assignment
    for kk = 1:k
        %搜索区域
        rmin = max(C(kk,4)-S, 1);
        rmax = C(kk,4)+S;
        if(rows-C(kk,4) < 2*S)
            rmax = rows;
        end
        cmin = max(C(kk,5)-S, 1);
        cmax = C(kk,5)+S;
        if(cols-C(kk,5) < 2*S)
            cmax = cols;
        end

        for ii = rmin:rmax
            for jj =cmin:cmax
                dl = C(kk,1) - img(ii,jj,1);
                da = C(kk,2) - img(ii,jj,2);
                db = C(kk,3) - img(ii,jj,3);
                dx = C(kk,4) - ii;
                dy = C(kk,5) - jj;
                dc2 = dl^2 + da^2 + db^2;
                ds2 = dx^2 + dy^2;
                D = sqrt(dc2 + ds2 * m^2 / S^2);
                if(D < d(ii,jj))
                    d(ii,jj) = D;
                    L(ii,jj) = kk;
                end
            end
        end
    end
    %update
    C(:) = 0;
```

```
        for ii = 1:rows
            for jj = 1:cols
                C(L(ii,jj),1:5) = C(L(ii,jj),1:5) + [img(ii,jj,1) img(ii,jj,2) img(ii,jj,3) ii jj];
                C(L(ii,jj),6) = C(L(ii,jj),6) + 1;
            end
        end
    for kk = 1:k
        C(kk,1:5) = round(C(kk,1:5)/C(kk,6));
    end
end

%直接定义的迭代几轮，就不设残差阈值和计算残差了
```

清除孤立像素

```
for ii = 2:rows-1
    for jj = 2:cols-1
        this = L(ii,jj);
        same_num = (this==L(ii-1,jj-1)) + (this==L(ii-1,jj)) + (this==L(ii,jj-1)) + (this==L(ii+1 ,jj-1)) + (this==L(ii-1,jj+1)) + (this==L(ii,jj+1)) + (this==L(ii+1,jj)) + (this==L(ii+1,jj+1));
        if(same_num < 3.5)
            if(L(ii,jj) ~= L(ii-1,jj))
                L(ii,jj) = L(ii-1,jj);
            else
                L(ii,jj) = L(ii,jj-1);
            end
            C(this,6) = C(this,6) - 1;
        end
    end
end
```

重新整理 L 和 C，得到 N

```
N = 0;
to_zero = 0;%用来记 kk 前有几个超像素被清没了，L 里标记减对应的数
for kk = 1:k
    if(C(kk,6) ~= 0)
        N = N + 1;
    else
        to_zero = to_zero + 1;
    end
    C(kk,6) = to_zero;%功能变了，以前存包含个数，现在存之前空超像素个数
end
for ii = 1:rows
    for jj = 1:cols
        L(ii,jj) = L(ii,jj) - C(L(ii,jj),6);
    end
end
```

```
end
```

Published with MATLAB® R2018b