%% set file path: manually select folder. Load data.

fp = uigetdir;

cd(fp)

list=dir('\*.mat');

list={list.name};

%% threshold cells

for f=1:size(list, 2)

fn=list{1,f}

load([fn])

cod2 = round(pts(:,4:5)); % take x and y coordinates of the localizations (column 4 = X position, column 5 = Y position in the pts function)

cod2(cod2(:,1)<1, :)=[];

cod2(cod2(:,2)<1, :)=[];

hst2 = accumarray(cod2(:, [2, 1]), 1, [512, 512]);

hst2=single(hst2);

bw = imbinarize(hst2, 1);

bw2 = bwmorph(bw, 'dilate', 2);

bw3 = imfill(bw2, 'holes');

bw4 = bwmorph( bw3, 'erode', 2);

figure, imagesc(bw4), axis xy

%% select 1 cell = the object with the largest surface area, and only take localizations inside this area into account

cc = bwconncomp(bw4);

props = regionprops(cc, 'area');

idx = find([props.Area] == max([props.Area]));

BW = ismember(labelmatrix(cc), idx);

B = cell2mat(bwboundaries(BW));

in = inpolygon(pts(:,4),pts(:,5),B(:,2),B(:,1));

%% plot all localizations inside the selected area as green circles

plot(B(:,2),B(:,1), '-r')

hold on

selpts = pts(in, [4,5]);

figure, plot(selpts(:,1), selpts(:,2), 'og'), axis xy

%% knnsearch: find k-nearest neighbours

[~, D] = knnsearch(selpts, selpts, 'K', 2);

%% plot diff limited image 2048 x 2048

cm=1/25:1/25:1;cm=cm';

pe=[0 0 0];

pe=cat(1, pe, [cm cm/2 zeros(25, 1)]);

pe=cat(1, pe, [ones(25, 1) 0.5+cm/2 zeros(25, 1)]);

pe=cat(1, pe, [ones(25, 1) ones(25, 1) cm]);

colormap(pe)

mx = 512/0.25;

cod = floor(pts(:,4:5)/0.25)+1;

cod(cod(:,1)<1, :)=[];

cod(cod(:,2)<1, :)=[];

hst = accumarray(cod(:, [2, 1]), 1, [mx, mx]);

hst=single(hst);

hsts=imgaussfilt(hst,1);

imagesc(hsts, [0 7]); colormap(pe);

axis xy equal tight %ij xy

line([1812.49,2000],[50,50],'Color','w','LineWidth',2) % put scalebar of 5µm

%% DBScan analysis with minimum 17 neighbors (MinPts) and search radius (ε) 0.55

DB = dbscan(selpts, 0.55, 17);

max(DB); % max number of clusters

selpts = selpts.\*0.107; % convert pixel to µm

%% calculate for all the clusters: the number of the cluster, number of localizations (NrPts), area, density, eccentricity and center of a plotted ellipse

prop\_cl = table('Size', [max(DB), 6], 'VariableTypes',{'single','single','single','single','single','single'}, 'VariableNames', {'ID', 'nrpts', 'area', 'density', 'ecc', 'center'});

prop\_cl.ID = [1:max(DB)]';

flg\_plot = 0;

h = waitbar(0, 'Calculating properties of the clusters...');

for i=1:max(DB)

cl\_each = selpts(find(DB == i), 1:2);

boundary(cl\_each(:,1), cl\_each(:,2)); %find edges of the cluster

bx = boundary(cl\_each(:,1), cl\_each(:,2));

prop\_cl.area(i) = polyarea(cl\_each(bx,1), cl\_each(bx,2)); %calculate area of cluster. in µm²

prop\_cl.nrpts(i) = length(cl\_each); %calculate density of the cluster. nrpts per um²

prop\_cl.density(i) = length(cl\_each)/prop\_cl.area(i);

try %calculate eccentricity using the MinVolEllipse function (fit with an ellipse)

[A, c] = MinVolEllipse (cl\_each', .01);

prop\_cl.center(i, 1:2) = c';

[~, Q, ~] = svd(A); r1 = 1/sqrt(Q(1,1)); r2 = 1/sqrt(Q(2,2));

o = max(r1, r2); p = min(r1, r2); prop\_cl.ecc(i) = sqrt (1-p^2/o^2);

if flg\_plot

plot(cl\_each(:,1), cl\_each(:,2), '.k');

hold on, plot(cl\_each(bx,1), cl\_each(bx,2), 'or'), title(i);

Ellipse\_plot(A, c)

hold off

pause

end

catch

disp([fn ' not analyzed'])

end

waitbar(i/max(DB), h)

end

close(h)

%% calculate cell-specific parameters

cell.area = max([props.Area])\*0.0114; %cell area in um²

cell.density = max(DB)/cell.area; %number of clusters per um2

cell.ratio = sum(prop\_cl.area)/cell.area; %ratio of area occupied by clusters

%% only retain clusters with ≥ 50 localizations and calculate again cell-specific parameters

prop\_50 = prop\_cl;

prop\_50(prop\_50.nrpts<50,:)=[];

cell50.area = max([props.Area])\*0.0114;

cell50.density = size(prop\_50, 1)/cell.area; %number of clusters per um2

cell50.ratio = sum(prop\_50.area)/cell.area; %ratio of area occupied by clusters

%% calculate mean and median cluster specific parameters per cell (both for all clusters and clusters ≥ localizations)

MM = table('Size', [2, 6], 'VariableTypes',{'single','single','single','single','single','single'}, 'VariableNames', {'area\_mean', 'area\_median', 'density\_mean', 'density\_median', 'ecc\_mean', 'ecc\_median'}, 'RowNames', {'all loc', '>50'});

MM (1,:) = {mean(prop\_cl.area), median(prop\_cl.area), mean(prop\_cl.density), median(prop\_cl.density), mean(prop\_cl.ecc), median(prop\_cl.ecc)};

MM (2,:) = {mean(prop\_50.area), median(prop\_50.area), mean(prop\_50.density), median(prop\_50.density), mean(prop\_50.ecc), median(prop\_50.ecc)};

%% save

file = [fp fn];

filename = file(27:strfind([fp fn], '.mat')-1); % adjust the number 27 depending on the name of the file in your pc

outfile = [filename '\_DBscan.mat'];

save(outfile, 'DB', 'selpts', 'file', 'cell', 'prop\_cl', 'cell50', 'prop\_50','MM');

filename\_prop\_50 = [filename '\_prop\_50.txt'];

filename\_prop\_cl = [filename '\_prop\_cl.txt'];

filename\_MM = [filename '\_means\_medians.txt'];

writetable(prop\_50, filename\_prop\_50);

writetable(prop\_cl, filename\_prop\_cl);

writetable(MM, filename\_MM, 'WriteRowNames',true);

% Close figures and clean variables

close all

clearvars -except f fn fp list pts

end