=====================================================================================

MIT License

Copyright (c) 2024 [REJU GEORGE THOMAS]

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all

copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

SOFTWARE.

=============================================================================

# MATLAB Skin Thickness

% Add Bio-Formats toolbox to MATLAB path

addpath('C:\Program Files\MATLAB\R2024b\bfmatlab\bfmatlab'); % Update with your actual path

% Ensure Bio-Formats is correctly configured

bfCheckJavaPath();

bfCheckJavaMemory();

% List of ND2 file paths

nd2FilePaths = {

*Add full path to filename*

}; % Add more file paths as needed

% Initialize variables for merged Z-stack and Z-slice index tracking

mergedZStack = [];

mergedZIndices = [];

currentZOffset = 0; % Track sequential Z indices across files

% Loop through each ND2 file

for fileIdx = 1:length(nd2FilePaths)

nd2FilePath = nd2FilePaths{fileIdx};

% Initialize the Bio-Formats reader

reader = bfGetReader(nd2FilePath);

% Get metadata

omeMeta = reader.getMetadataStore();

zSize = omeMeta.getPixelsSizeZ(0).getValue(); % Number of Z slices

cSize = omeMeta.getChannelCount(0); % Number of channels

width = omeMeta.getPixelsSizeX(0).getValue(); % Image width

height = omeMeta.getPixelsSizeY(0).getValue(); % Image height

% Preallocate an array for the FITC (Green) channel Z-stack

fitcStack = zeros(height, width, zSize, 'double');

% Loop through Z-planes and extract each one

for z = 1:zSize

% Read only the FITC channel (assuming second channel is FITC)

if cSize > 1

fitcStack(:, :, z) = bfGetPlane(reader, (z - 1) \* cSize + 2);

else

warning('ND2 file does not have a second channel for FITC fluorescence.');

end

end

% Close the reader

reader.close();

% Normalize Z-stack for visualization (scale to [0, 1])

fitcStackNorm = (fitcStack - min(fitcStack(:))) / (max(fitcStack(:)) - min(fitcStack(:)));

% Restore full Z-stack without filtering planes

validPlanes = fitcStackNorm;

% Adjust Z-plane indices to be sequential

numValidPlanes = size(validPlanes, 3);

if numValidPlanes > 0

adjustedZIndices = currentZOffset + (1:numValidPlanes);

mergedZStack = cat(3, mergedZStack, validPlanes);

mergedZIndices = cat(1, mergedZIndices, flip(adjustedZIndices')); % Invert Z-stack order

currentZOffset = currentZOffset + numValidPlanes; % Increment offset

end

end

% Create the X, Y grid for the surface plot

[X, Y] = meshgrid(linspace(1, size(mergedZStack, 2), size(mergedZStack, 2)), ...

linspace(1, size(mergedZStack, 1), size(mergedZStack, 1)));

% Plot the 3D surface for each slice layer

figure;

hold on;

for z = 1:size(mergedZStack, 3)

Z1 = double(mergedZIndices(z)) \* ones(size(mergedZStack(:, :, z))); % Inverted Z index as height

Z2 = mergedZStack(:, :, z); % Intensity values

surf(X, Y, Z1, Z2, 'EdgeColor', 'none'); % Map intensity to color

end

colormap jet; % Use jet colormap for intensity

colorbar;

title('Layered 3D Surface Plot: Bottom-to-Top Z-Stack (FITC Channel Only)');

% Set axis labels with increased font size and bold text

xlabel('X-axis (Pixels)', 'FontSize', 14, 'FontWeight', 'bold');

ylabel('Y-axis (Pixels)', 'FontSize', 14, 'FontWeight', 'bold');

zlabel('Z-slice index (bottom to top)', 'FontSize', 14, 'FontWeight', 'bold');

% Adjust viewing angle for better 3D visualization

view(3); % Set 3D view

set(gca, 'ZDir', 'reverse'); % Flip the Z-axis to display bottom Z-stack at the top

% Enhance appearance

grid on;

shading interp; % Smooth shading

hold off;