

Voxel filtering manual

Two filtering steps are performed in sequence:

- 3D spatial median filter
- Temporal filter over time of interest

Procedures:

1. Plug in the directory of the reconstructed MAT files (**Not Tiff**)

```
%% Section 1: Read in Mat Files
```

```
Mat_datasetdir='E:\yangyang\FLFM_reconstruction_My\Data_Seisure\2023_Sebfish';
img_folder='Substack (47-97)\';
```

2. Plug in the ROI 3D Mask for your reconstructed data

- ❖ The mask file here in Tiff since I created it in FIJI but any binary mask data that's of the same size as your data and can be recognized by MATLAB is fine.

```
%% Section 2: Read in Mask (mask of ROI)
```

```
% Note that it could be a subsection of interest in 3D where signals are
% interesting to you
```

```
Mask_datasetdir='E:\yangyang\FLFM_reconstruction_My\Data_Seisure\2023_Sebfish';
fileName='ROI_Mask.tif';
```

3. Test run the filtering first by setting a subset of time point in **Event_Tpts**. Here, the time points should be more than 60 since it is the minimum number of data points required for MATLAB filtering function to run.

```
%% Section 3: Read in Voxel signals
```

```
Event_Tpts=1:100;
% Event_Tpts=1:length(imglist);
```

4. Start median spatial filtering with a filter cube of size $5 \times 5 \times 5$. The size can be changed in the red box area. More information refers to [3-D median filtering - MATLAB medfilt3 \(mathworks.com\)](https://www.mathworks.com/help/matlab/ref/medfilt3.html).

```
%% Section 4 : 3D Spatial filter
```

```
Voxel.Spfftraces=zeros(length(ROI_ind),length(Event_Tpts)); % #Voxels * #Tpts
fprintf('start medfilt3 spatial filter...\n');
for t=1:length(Event_Tpts)
    img=zeros(single(size(ROIMask)));
    img(ROI_ind)=Voxel.traces(:,t);
    img =medfilt3(img,[5 5 5]) ; % median 3D filter with cubic of 5 Voxels
```

5. Start lowpass filter. Here I used Butterworth filter. You could use any filter you want. There are a lot of filter function in MATLAB. To make sure the processing size does not exceed the GPU memory, the number of voxels to be filtered each time is set to be 5000 in **Voxel_step**. You could change to a different number depending on your GPU memory. To set the Butterworth filter, first set the sampling rate then set the top frequency you would like to cut-off. The example given here is: **30HZ** of sampling rate and low-pass signal below **3HZ**.

```
%% Section 5: Temporal filtering butterworth
d = designfilt("lowpassiir", 'SampleRate', 20, ...
    'HalfPowerFrequency', 3, 'DesignMethod', "butter", FilterOrder=20);

tic;
Voxel_step=5000;
Voxel.filttraces=[];
% for i=1:Voxel_step:1000
```

6. Set saving folder. The filtered data will be in Tiff

```
%% Section 6: write out

% Setting Nomalization
Numcolor = 65536;
Imgmax=max(Voxel.filttraces(:));
Imgmin = 0;

Tiff_datasetdir='E:\yangyang\FLFM_reconstruction_My\Data_Seisure\2023_Sebfush
```