% created in January 2021, by Ho Nhu Y Nguyen

This dataset contains the data used in the article: “Three-dimensional view of out-of-plane artifacts in photoacoustic imaging using a laser-integrated linear-transducer-array probe”.

https://doi.org/10.1016/j.pacs.2020.100176

- Data files for the Method section: .rfe files with a prefix “oneAbsorber”
- Data files for section 4.1: .rfe files with a prefix “in-plane-absorber”
- Data files for section 4.2: .rfe files with a prefix “out-of-plane-absorber”
- Data files for section 4.3: .rfe files with a prefix “in-vivo”
- Data files for section 4.5: xz calibration.mat
- Data files for section 4.5.1: .rfe files with a prefix “two-absorbers”

1. Figures in this paper

This part explains how to plot those Matlab figures used in the paper.

1.1. Figure 1

- Run noLaser_DMAS_3D_axial_displacement.m
- Select all rfe files a prefix “oneAbsorber”
- Run the part for figure 2 in three_dimensional_reconstruction_figures.m

1.2. Figure 2

- Comment this part in noLaser_DMAS_3D_axial_displacement.m

```
for iFile = 1:nFiles-1
    temp1 = squeeze(tempRekon3D(iFile,:,:,:));
    for jFile = iFile+1:nFiles
        temp2 = squeeze(tempRekon3D(jFile,:,:,:));
        tempRekon = temp1.*abs(temp2);
        tempRekon = sign(tempRekon).*sqrt(abs(tempRekon));
        rekon3D = rekon3D + tempRekon;
    end
```

- Uncomment this part in noLaser_DMAS_3D_axial_displacement.m

```
for iFile = 1:nFiles
    tempRekon = squeeze(tempRekon3D(iFile,:,:,:));
    rekon3D = rekon3D + tempRekon;
```

- Run noLaser_DMAS_3D_axial_displacement.m
- Select all rfe files a prefix “oneAbsorber”
- Run the part for figure 2 in three_dimensional_reconstruction_figures.m
- Comment and uncomment back when finished.

1.3. Figure 3

- Comment this part in noLaser_DMAS_3D_axial_displacement.m

```
for iFile = 1:nFiles-1
    temp1 = squeeze(tempRekon3D(iFile,:,:,:));
    for jFile = iFile+1:nFiles
        temp2 = squeeze(tempRekon3D(jFile,:,:,:));
        tempRekon = temp1.*abs(temp2);
        tempRekon = sign(tempRekon).*sqrt(abs(tempRekon));
        rekon3D = rekon3D + tempRekon;
    end
```

- Uncomment this part in noLaser_DMAS_3D_axial_displacement.m

```
for iFile = 1:nFiles
    tempRekon = squeeze(tempRekon3D(iFile,:,:,:));
    rekon3D = rekon3D + tempRekon;
```

- Run noLaser_DMAS_3D_axial_displacement.m
- Select all rfe files a prefix “oneAbsorber”

- When finished, run this command in Matlab `rekon3DDAS = rekon3DZ;`
- Repeat the above steps but with steps 1 and 2 inverse
- When finished, run this command in Matlab `rekon3DDMAS = rekon3DZ;`
- Run the part for figure 3 in three_dimensional_reconstruction_figures.m
- Comment and uncomment back the parts in recon_3D.m when finished.

1.4. Figure 6

- In recon_3D_x_z_displacement.m, set:
```
system = 2;
dispStepX = 0.525;
dispStepZ = 0.525;
```
- Run recon_3D_x_z_displacement.m, select all rfe files with a prefix “in-plane-absorber”
- Run the part for figure 6 in three_dimensional_reconstruction_figures.m

1.5. Figure 7

- Repeat the first 2 steps in Figure 6
- In calculate_resolutions.m, set
```
xCrop = 1:100;
yCrop = 24:44;
zCrop = 140:210;
```
- Run the part for figure 7 in three_dimensional_reconstruction_figures.m

1.6. Figure 8

- In recon_3D_x_z_displacement.m, set:
```
system = 2;
dispStepX = 0.525;
dispStepZ = 0.525;
```
- Run recon_3D_x_z_displacement.m, select all rfe files with a prefix “out-of-plane-absorber”
- Run the part for figure 8 in three_dimensional_reconstruction_figures.m
- To calculate the resolutions, repeat steps in Figures 7 with
```
xCrop = 45:105;
yCrop = 24:44;
zCrop = 61:121;
```

1.7. Figure 9

- Run show_image_multi_wavelength.m
- Select file in-vivo-23.00-36_20190606_110617.rfe
- Run the part for figure 9 in three_dimensional_reconstruction_figures.m

1.8. Figure 10

- Comment everything below `rekon3DCal(rekon3D<threshold) = 0;` in axial_sensitivity_compensation.m and in elevation_sensitivity_compensation.m
- Run recon_3D_x_z_displacement.m, select all rfe files with a prefix “in-vivo”
- Run the script paper_figures.m with proper code for this figure.
- Run the part for figure 10 in three_dimensional_reconstruction_figures.m
- Change the view angle for different figures
- When finished, undo step 1.

1.9. Figure 11

- Run the part for figure 11 in three_dimensional_reconstruction_figures.m
- Select “out-of-plane-absorber _20192131_132122.rfe”
- Select all rfe files with a prefix “out-of-plane-absorber”

- Select all rfe files with a prefix “out-of-plane-absorber”
- Select “in-vivo-23.00-36_20190606_110617.rfe”
- Select all rfe files with a prefix “in-vivo”
- Select all rfe files with a prefix “in-vivo”

1.10. Figure 12

- Run the part for figure 12 in three_dimensional_reconstruction_figures.m
- That part for figure 12 loads processed calibration data “xz calibration.mat”.

1.11. Figure 13

- Comment everything below `rekon3DCal(rekon3D<threshold) = 0;` in axial_sensitivity_compensation.m and in elevation_sensitivity_compensation.m
- Run recon_3D_x_z_displacement.m, select all rfe files with a prefix “two-absorbers”
- Run the part for figure 12 in three_dimensional_reconstruction_figures.m
- Undo step 1.
- In recon_3D.m, set:

```
rekon3DCal = axial_sensitivity_compensation(x,z,rekon3D,...
     tempRekon3D,20,probePos);
rekon3DCal = elevation_sensitivity_compensation(x,z,rekon3D,...
     tempRekon3D,30,probePos);
```

- Repeat steps 2 and 3

1.12. Figure 14

- Similar as Figure 13
- In recon_3D.m, set:

```
threshold = max(rekon3DZ(:))/6;
rekon3DCal = axial_sensitivity_compensation(x,z,rekon3D,...
     tempRekon3D,threshold,probePos);
rekon3DCal = elevation_sensitivity_compensation(x,z,rekon3D,...
     tempRekon3D,threshold,probePos);
```

2. Videos in this paper

Videos are created using “three_dimensional_reconstruction_videos.m”. These are 3D images in different view angles.