Table of Contents

Define parameters 1
get US data 1
get RF Data 2
get consecutive US Data 2
get maximum decorrelation for setting dynamic rance 3
display decorr in middle x-y slice for each data set 3
display decorr in middle x-y slice for each data set 4
ROC curve from R 5
ROC curve from R 6
display decorr in middle x-y slice for each data set 7
RF Time 10

Define parameters

```
warning('off','all')
try
    if ispc
        basePath = strcat('C:\Users\',getenv('username'),'\Box
\SiemensSC2000IQData');
    elseif ismac
        basePath = strcat('/Users/',getenv('USER'),'/box');
    end
catch
    basePath = matlabroot;
end
fullDirectory =  uigetdir(basePath);
dataDirectory = dir(fullDirectory);
fileName = 'bufApl0Out_0x0_0x0.data.dm.pmcr';
%user defined
azimuthAngle = 2*pi*80/360;
rangeAngle = 2*pi*80/360;
rmin = 0;
cartScalingFactor = 2; % dx/dr
sigma = 1; %mm
frameRate = 22; % hz
```

get US data

```
if ispc
    fullPath =strcat(fullDirectory,'\',
{dataDirectory(6).name},'\',fileName);
elseif ismac
    fullPath =strcat(fullDirectory,'/',
{dataDirectory(6).name},'/',fileName);
end

DmBase = read_lbdump(fullPath{1});
```

```
%computed from existing parameters/infofile
rmax = (size(DmBase.data,1)/DmBase.Info.NumSamplesPerMm);
thetamin= -azimuthAngle/2;
thetamax = azimuthAngle/2;
phimin=-rangeAngle/2;
phimax=rangeAngle/2;

interFrameTime = 1/frameRate;
```

get RF Data

```
RFDataTable = readtable('RFData_2019_1_31_16_52_332.txt');
for index = 1:length(RFDataTable.Day_Month_Year)
    dateInter = strsplit(RFDataTable.Day_Month_Year{index},'-');
    RFDay(index) = str2num(dateInter{1});
    RFMonth(index) = str2num(dateInter{2});
    RFYear(index) = str2num(dateInter{3});
end
RFData.RFDataTime =
 datetime(RFYear',RFMonth',RFDay',RFDataTable.Hours,RFDataTable.Minutes,RFDataTabl
RFStartTime = RFData.RFDataTime(2);
USStartTime = DmBase.startTime;
diffTime = RFStartTime - USStartTime;

for index = 1:length(RFData.RFDataTime)
    RFData.RFDataTime(index) = RFData.RFDataTime(index); %- diffTime;
    RFData.RFDataTime(index) = RFData.RFDataTime(index) -hours(5);
end
RFData.t1 = RFDataTable.t1;
RFData.t2 = RFDataTable.t2;
RFData.t3 = RFDataTable.t3;
RFData.t4 = RFDataTable.t4;
RFData.currentPower = RFDataTable.currentPower;
RFData.timer = RFDataTable.timer;
RFData.deliveredPower = RFDataTable.deliveredPower;
RFData.efficiency = RFDataTable.efficiency;
RFData.targetTemp = RFDataTable.targetTemp;
RFData.modeIndicator = RFDataTable.modeIndicator;
RFData.tempButtons = RFDataTable.tempButtons;
RFData.impedance = RFDataTable.impedance;
```

get consecutive US Data

if ispc fullPath =strcat(fullDirectory,'\',{dataDirectory(n).name},'\',fileName); elseif ismac fullPath =strcat(fullDirectory,'/',{dataDirectory(n).name},'/',fileName); end Dm = read_lbdump(fullPath{1}); usData(1) = USDataClass(Dm.data,Dm.startTime, Dm.Info,rmin,rmax,thetamin,thetamax,phimin,phimax,cartScalingFactor,sigma,interFrameTime); %usData.computeGaussianMask; usData(1).computeScanConvParams; usData(1).scanConvert3D_Frust;

```
tic;
for n = 3:size(dataDirectory)
```

```
    %tic
    if ispc
        fullPath =strcat(fullDirectory,'\',
{dataDirectory(n).name},'\',fileName);
    elseif ismac
        fullPath =strcat(fullDirectory,'/',
{dataDirectory(n).name},'/',fileName);
    end
    dataDirectory(n).name
    Dm = read_lbdump(fullPath{1});
    usData(n-2) = USDataClass(Dm.data,Dm.startTime,
 Dm.Info,rmin,rmax,thetamin,thetamax,phimin,phimax,cartScalingFactor,sigma,interFr

    %usData(n-3).scanConvert3DVolume_sliceMethod;
    %usData(n-3).compute3DDecorr_sliceMethod;
    %tic;
    %tic
    usData(n-2).scanConv_Frust();
    %toc
    %usData(n-3).scanConvert3D_Frust;
    %toc;
    %tic
    usData(n-2).compute3DDecorr();
    %toc
    %tocArr(n-3) = toc
end
toc;
```

get maximum decorrelation for setting dynamic rance

```
maxDecorr = 0;
for n = 1:length(usData);
    if(max(usData(n).decorr(:))>maxDecorr)
        maxDecorr = max(usData(n).decorr(:));
    end
end
maxDecorr
```

display decorr in middle x-y slice for each data set

```
for n = 1:length(usData);
    figure(1)
    imagesc(usData(1).y_range,usData(1).z_range,
 usData(n).decorr(:,:,floor(end/2)-10,1),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
```

```
    colormap('gray');
    figure(2)
    imagesc(usData(1).y_range,usData(1).z_range,
 usData(n).decorr(:,:,floor(end/2),1),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
    colormap('gray');
    figure(3)
    imagesc(usData(1).y_range,usData(1).z_range,
 usData(n).decorr(:,:,floor(end/2)+10,1),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
    colormap('gray');
    pause(1);

end
```

display decorr in middle x-y slice for each data set

```
for n = 1:length(usData);
    figure(1)
    imagesc(usData(1).y_range,usData(1).z_range,
 squeeze(usData(n).decorr(:,floor(end/2)-10,:,1)),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
    colormap('gray');
    figure(2)
    imagesc(usData(1).y_range,usData(1).z_range,
 squeeze(usData(n).decorr(:,floor(end/2),:,1)),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
    colormap('gray');
    figure(3)
    imagesc(usData(1).y_range,usData(1).z_range,
 squeeze(usData(n).decorr(:,floor(end/2)+10,:,1)),[0 maxDecorr]);
    title('decorrelation map');
    xlabel('Azimuth(mm)');
    ylabel('Range (mm)');
    colorbar;
    colormap('gray');
    pause(1);

end
```

```
for frame = 1:length(usData)
    for n = 1:size(usData(1).decorr,3)
        imagesc(usData(frame).y_range,usData(frame).z_range,
 squeeze(usData(8).decorr(:,:,n,1)),[0 maxDecorr]);
        pause(.1)
    end
end

for frame = 1:length(usData)
    for n = 1:size(usData(1).decorr,2)
        imagesc(usData(frame).y_range,usData(frame).z_range,
 squeeze(usData(8).decorr(:,n,:,1)),[0 maxDecorr]);
        pause(.1)
    end
end

for frame = 1:length(usData)
    for n = 1:size(usData(1).decorr,1)
        imagesc(usData(frame).y_range,usData(frame).x_range,
 imrotate(squeeze(usData(8).decorr(n,:,:,1)),90),[0 maxDecorr]);
        axis image
        pause(.1)
    end
end
```

ROC curve from R

```
decorrArr = usData(5).decorr(:,floor(end/2),:,1);
%imagesc(usData(8).decorr(x:,:,floor(end/2),1),[0 maxDecorr]);
thisPic = imread('thispic.png');
orignalImageSize = size(thisPic)
thisPicDown = imresize(thisPic,[144 184]);
imagesc(imrotate(flip(thisPicDown),180))
%imagesc(squeeze((abs(decorrArr))));
axis image
labelObj = imfreehand;
labelArr = labelObj.createMask;
labelArr = labelArr(:);
decorrArr = decorrArr(:);
save('testDataExport.mat','labelArr','decorrArr')
shellString = strcat(pwd,'/','3DEchoDecorrelationScripts/DataScripts/
Matlab/TestScripts/ROCCurve.r ',{' '},'testDataExport.mat');
[test1234 testout] = system(shellString{1});
load('ROCOutput.mat')
imshow('thispic.png')
[maxVal maxInd] = max(1-ROC_fpr + ROC_tpr);
ROC_cutoff(maxInd)
plot(ROC_fpr,ROC_tpr,ROC_fpr(maxInd),ROC_tpr(maxInd),'r*')
 cutOff = ROC_cutoff(maxInd);
 thisSlice = squeeze(usData(8).decorr(:,floor(end/2),:,1));
 thisSlice(find(thisSlice < cutOff)) = 0;
 thisSlice(find(thisSlice >= cutOff)) = 1;

 edgeVals = edge(thisSlice);
```

```
 edgeVals = imresize(edgeVals,orignalImageSize(1:2));
 thisSlice = imresize(thisSlice,orignalImageSize(1:2));
 edgeVals(find(edgeVals <1)) = 0;
 thisSlice(find(thisSlice <1)) = 0;
 edgeVals = floor(edgeVals);
 thisSlice = floor(thisSlice);
 imshow(imrotate(flip(thisPic),180));
 alphamask(thisSlice,[0 0 1],.2)
% alphamask(edgeVals,[0 0 0],1)
```

ROC curve from R

```
currPic =0;
labelArr = [];
decorrArr = [];
labelArr_full = [];
decorrArr_full = [];
for currDepth = floor(linspace(1,size(usData(2).decorr,1),14))
    currPic = currPic+1;
    decorrArr = usData(2).decorr(currDepth,:,:,1);
    %imagesc(usData(8).decorr(x:,:,floor(end/2),1),[0 maxDecorr]);
    thisPic = imread(strcat('t1_slice',num2str(currPic),'.png'));
    orignalImageSize = size(thisPic)
    thisPicDown = imresize(thisPic,[205 205]);
    figure(1)
    imagesc(squeeze((abs(decorrArr))));
    axis image
    figure(2)
    imagesc((thisPicDown))
    axis image

    axis image
    labelObj = imfreehand;
    labelArr = labelObj.createMask;
    labelArr = labelArr(:);
    labelArr_full = [labelArr_full; labelArr];
    decorrArr = decorrArr(:);
    decorrArr_full = [decorrArr_full; decorrArr];
    save('testDataExport.mat','labelArr','decorrArr')
    shellString = strcat(pwd,'/','3DEchoDecorrelationScripts/
DataScripts/Matlab/TestScripts/ROCCurve.r ',{'
 '},'testDataExport.mat');
    [test1234 testout] = system(shellString{1});
    load('ROCOutput.mat')
    imshow('rocPlot2.png')
    [maxVal maxInd] = max(1-ROC_fpr + ROC_tpr);
    ROC_cutoff(maxInd)
    %plot(ROC_fpr,ROC_tpr,ROC_fpr(maxInd),ROC_tpr(maxInd),'r*')
    cutOff = ROC_cutoff(maxInd);
    thisSlice = squeeze(usData(8).decorr(currDepth,:,:,1));
    thisSlice(find(thisSlice < cutOff)) = 0;
    thisSlice(find(thisSlice >= cutOff)) = 1;
```

```
    edgeVals = edge(thisSlice);
    edgeVals = imresize(edgeVals,orignalImageSize(1:2));
    thisSlice = imresize(thisSlice,orignalImageSize(1:2));
    edgeVals(find(edgeVals <1)) = 0;
    thisSlice(find(thisSlice <1)) = 0;
    edgeVals = floor(edgeVals);
    thisSlice = floor(thisSlice);
    imshow(thisPic);
    alphamask(thisSlice,[0 0 1],.2)
    alphamask(edgeVals,[0 0 0],1)
end
labelArr = labelArr_full;
decorrArr = decorrArr_full;
save('testDataExport.mat','labelArr','decorrArr')
    shellString = strcat(pwd,'/','3DEchoDecorrelationScripts/
DataScripts/Matlab/TestScripts/ROCCurve.r ',{'
 '},'testDataExport.mat');
    [test1234 testout] = system(shellString{1});
    load('ROCOutput.mat')
    imshow('rocPlot2.png')
    [maxVal maxInd] = max(1-ROC_fpr + ROC_tpr);
    ROC_cutoff(maxInd)
    plot(ROC_fpr,ROC_tpr,ROC_fpr(maxInd),ROC_tpr(maxInd),'r*')
    cutOff = ROC_cutoff(maxInd);
    thisSlice = squeeze(usData(8).decorr(currDepth,:,:,1));
    thisSlice(find(thisSlice < cutOff)) = 0;
    thisSlice(find(thisSlice >= cutOff)) = 1;

    edgeVals = edge(thisSlice);
    edgeVals = imresize(edgeVals,orignalImageSize(1:2));
    thisSlice = imresize(thisSlice,orignalImageSize(1:2));
    edgeVals(find(edgeVals <1)) = 0;
    thisSlice(find(thisSlice <1)) = 0;
    edgeVals = floor(edgeVals);
    thisSlice = floor(thisSlice);
    imshow(thisPic);
    alphamask(thisSlice,[0 0 1],.2)
    alphamask(edgeVals,[0 0 0],1)

plot(ROC_fpr,ROC_tpr,ROC_fpr(maxInd),ROC_tpr(maxInd),'r*',linspace(0,1,length(ROC_
title('ROC Curve')
xlabel('False Positive Rate')
ylabel('True Positive Rate')
```

display decorr in middle x-y slice for each data set

```
for n = 1:size(usData(8).rawData_cart,3);
    imagesc(log10(abs(usData(1).rawData_cart(:,:,n,1))),
[log10(min(abs(usData(1).rawData_cart(:)))),
 log10(max(abs(usData(1).rawData_cart(:))))]);
    colormap('gray')
```

```
    colorbar;
    pause(.001);

end

thresh = 15;
for frames = 1:length(usData)
    for n = 1:size(usData(frames).decorr,3)
    %for n = 50:50
        figure(1)
        thisVol = squeeze(abs(usData(frames).decorr(:,:,n,1))) %-
squeeze(abs(usData(1).decorr_slicemethod(:,:,n,1)));

  %imagesc(20*log10(abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))));
        imagesc(usData(5).z_range,usData(5).x_range,thisVol);

        colormap('gray')
        hold on
        convolvedImage = conv2((1/25)*ones(10,10),squeeze((thisVol)));

  %contour(usData(5).y_range,usData(5).x_range,convolvedImage(5:end-5,5:end-5),
[thresh thresh],'r');
        pause(.1);
        hold off
    end
end

thresh = 15;
for frames = 1:length(usData)
    for n = 1:size(usData(frames).decorr_slicemethod,3)
    %for n = 50:50
        figure(1)
        thisVol =
 squeeze(abs(usData(frames).decorr_slicemethod(:,:,n,1))) %-
squeeze(abs(usData(1).decorr_slicemethod(:,:,n,1)));

  %imagesc(20*log10(abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))));
        imagesc(usData(5).y_range,usData(5).x_range,thisVol);

        colormap('gray')
        hold on
        convolvedImage = conv2((1/25)*ones(10,10),squeeze((thisVol)));

 contour(usData(5).y_range,usData(5).x_range,convolvedImage(5:end-5,5:end-5),
[thresh thresh],'r');
        pause(.1);
        hold off
    end
end

n = 50
figure(1)
thresh = .4;
for frames = 3:length(usData)
```

```
    thisVol =
 squeeze(abs(usData(frames).decorr_slicemethod(:,:,n,1))) %-
squeeze(abs(usData(1).decorr_slicemethod(:,:,n,1)));

  %imagesc(20*log10(abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))));
    imagesc(usData(5).y_range,usData(5).x_range,thisVol);

    colormap('gray')
    hold on
    convolvedImage = conv2((1/25)*ones(10,10),squeeze((thisVol)));

 contour(usData(5).y_range,usData(5).x_range,convolvedImage(5:end-5,5:end-5),
[thresh thresh],'r');
    pause(1);
end

h = ginput(2)
mydist = pdist(h)
plot(h)

for frames = 3:length(usData)
    for n = 1:size(usData(frames).decorr_slicemethod,2)
    %for n = 50:50
        figure(1)
        thisVol =
 squeeze(abs(usData(frames).decorr_slicemethod(:,n,:,1))) -
squeeze(abs(usData(1).decorr_slicemethod(:,n,:,1)));

  %imagesc(20*log10(abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))));
        imagesc(thisVol,[0 .45]);

        %colormap('gray')
        hold on
        contour(conv2((1/25)*ones(10,10),squeeze((thisVol))),1,'r');
        pause(.1);
        hold off
    end
end

for frames = 3:length(usData)
    for n = 61
    %for n = 50:50
        figure(1)
        thisVol =
 squeeze(abs(usData(frames).decorr_slicemethod(:,n,:,1))) -
squeeze(abs(usData(1).decorr_slicemethod(:,n,:,1)));

  %imagesc(20*log10(abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))));
        imagesc(thisVol,[0 .45]);

        %colormap('gray')
        hold on
        contour(conv2((1/25)*ones(10,10),squeeze((thisVol))),1,'r');
        pause(1);
```

```
        hold off
    end
end
```

RF Time

for dataSet = 1:length(usData) usDataTime(dataSet) = usData(dataSet).time; usData(dataSet).ROIBounds = [10,66,50,80,52,78,NaN,NaN]; usROIData_slicemethod(dataSet).data = (usData(dataSet).decorr_slicemethod(10:66,50:80,52:78,1:end)); usAvg_slicemethod(dataSet) = mean(mean(mean(mean(usROIData_slicemethod(dataSet).data)))) end

```
for dataSet = 1:length(usData)
    usDataTime(dataSet) = usData(dataSet).time;
    %usData(dataSet).ROIBounds = [10,66,50,80,52,78,NaN,NaN];
    usROIData(dataSet).data = (usData(dataSet).decorr);
    usAvg(dataSet) = mean((usROIData(dataSet).data(:)))
end
plot(usAvg);

plot(RFData.RFDataTime,RFData.t1/(max(RFData.t1(:))), ...
    RFData.RFDataTime,RFData.t2/
(max(RFData.t2(:))),RFData.RFDataTime,RFData.t3/
(max(RFData.impedance(:))),RFData.RFDataTime,RFData.t4/
(max(RFData.currentPower(:))))
title('Plot of obtained parameters from reverse engineered data
 stream')
xlabel('time');
ylabel('Parameter value, normalized')
legend('t1','t2','Impedance','Current power being delivered')

for frames = 1:length(usData)
    for n = 1:size(usData(frames).decorr_slicemethod,3)
    %for n = 50:50
        figure(1)


 imagesc(imrotate(squeeze(abs(usROIData_slicemethod(frames).data(:,:,n,1))),180));
        hold on

 contour(imrotate(conv2((1/25)*ones(20,20),squeeze((usROIData_slicemethod(frames).
        pause(.1);
        hold off
    end
end
for frames = 1:length(usData)
    for n = 1:size(usData(frames).decorr,3)
    %for n = 50:50
        figure(1)


 imagesc(imrotate(squeeze(abs(usROIData(frames).data(:,:,n,1))),180));
        hold on

 contour(imrotate(conv2((1/25)*ones(20,20),squeeze((usROIData(frames).data(:,:,n,1
```

```
        pause(.1);
        hold off
    end
end

for frames = 3:length(usData)
    for n = 1:size(usData(frames).decorr_slicemethod,3)
    %for n = 50:50
        figure(1)
        thisVol =
 squeeze(abs(usData(frames).decorr_slicemethod(:,:,n,1))) -
squeeze(abs(usData(1).decorr_slicemethod(:,:,n,1)));

 imagesc((abs(usData(frames).rawData_cart_slicemethod(:,:,n,1))),
[0 .1]);
        %imagesc(thisVol,[0 20]);

        %colormap('gray')
        hold on
        contour(conv2((1/25)*ones(10,10),squeeze((thisVol))),.4,'r');

  %imagesc(abs(conv2((1/25)*ones(10,10),squeeze((thisVol)))),1,'r','AlphaData', .1
        %imagesc(abs(thisVol),1,'r','AlphaData', .1);
        figure(2)
        imagesc(abs(squeeze(thisVol)))
        pause(.1);
        hold off
    end
end

for i = 3:length(usData)
    decorrSum(i-2) = sum(sum(sum(usData(i).decorr)));
end
plot(RFData.RFDataTime,RFData.t1/(max(RFData.t1)),
[usData(3:end).time],decorrSum/max(decorrSum))
title('average decorrelation and temperature')
legend('Temperature, normalized','Decorrelation, normalized')

for i = 1:length(usData)
    decorrAvg(i) =
 sum(sum(sum(usData(i).decorr_slicemethod(10:66,50:80,10:66))))/
prod(size(usData(i).decorr_slicemethod(10:66,50:80,10:66)));
end

for n = 2:length(usData)
    thisVol = squeeze(usData(n).decorr_slicemethod(:,:,:,1));
    kernelSize = 5;
    smoothkernel = 1/
(kernelSize^3)*ones(kernelSize,kernelSize,kernelSize);
    convedVol = convn(smoothkernel,thisVol);
    %[xMax yMax zMax] = size(convedVol);
    [X Y Z] =
 meshgrid(1:usData(n).x_range,1:usData(n).y_range,usData(n).z_range);
    isosurface(X,Y,Z,convedVol,0)
    hold on
```

```
    %isosurface(X,Y,Z,convedVol,9)
    %isosurface(X,Y,Z,convedVol,8)
    %isosurface(X,Y,Z,convedVol,7)
    %isosurface(X,Y,Z,convedVol,6)
    %isosurface(X,Y,Z,convedVol,5)
    isosurface(X,Y,Z,convedVol,8)
    %isonormals(X,Y,Z,
(squeeze(usData(2).decorr_slicemethod(:,:,:,1))),p1)
    %p.FaceColor = 'red';
    %p.EdgeColor = 'none';
    pause(.1);
    hold off
    view(-131,30)
    input('next')
    clf
end
```

Published with MATLAB® R2016a