
```

%Reads Raw Data from Ryans Struct and stores it in frames in a cell
    array
%structure. This is saved as AnimationFrames.mat and is read in by the
%AnimateData.m function

%clean it up
close all; clear all; clc

%load raw data
load('RawDataUP.Mat')

% create a frame of NaNs to pad IP tests that ended early
padScreen = ones(5,6)*NaN;

% Process data to make relative and remove outliers

% ambient
Amb.IP(Amb.IP < -198) = NaN;
Amb.IP(Amb.IP == 0) = NaN;
Amb.Ref(isoutlier(Amb.Ref(:,2:end), 'mean')) = NaN;

AmbMat = cell(length(Amb.Ref),2);

for i = 1:length(Amb.Ref)

    if i > length(Amb.IP)
        AmbMat{i,1} = padScreen;
    else
        AmbMat{i,1} = vec2mat(Amb.IP(i,2:end),6); %IP
    end

    AmbMat{i,2} = vec2mat(Amb.Ref(i,2:end),6); %Ref

end

% black out
Black.IP(Black.IP < -198) = NaN;
Black.IP(Black.IP == 0) = NaN;
Black.Ref(isoutlier(Black.Ref(:,2:end), 'mean')) = NaN;

BlackMat = cell(length(Black.Ref),2);

for i = 1:length(Black.Ref)

    if i > length(Black.IP)
        BlackMat{i,1} = padScreen;
    else
        BlackMat{i,1} = vec2mat(Black.IP(i,2:end),6); %IP
    end

    BlackMat{i,2} = vec2mat(Black.Ref(i,2:end),6); %Ref

```

```

end

% filtered
Filt.IP(Filt.IP < -198) = NaN;
Filt.IP(Filt.IP == 0) = NaN;
Filt.Ref(isoutlier(Filt.Ref(:,2:end), 'mean')) = NaN;

FiltMat = cell(length(Filt.Ref),2);

for i = 1:length(Filt.Ref)

    if i > length(Filt.IP)
        FiltMat{i,1} = padScreen;
    else
        FiltMat{i,1} = vec2mat(Filt.IP(i,2:end),6); %IP
    end

    FiltMat{i,2} = vec2mat(Filt.Ref(i,2:end),6); %Ref

end

% emissivity
Emis.IP(Emis.IP < -198) = NaN;
Emis.IP(Emis.IP == 0) = NaN;
Emis.Ref(isoutlier(Emis.Ref(:,2:end), 'mean')) = NaN;

EmisMat = cell(length(Emis.Ref),2);

for i = 1:length(Emis.Ref)

    if i > length(Emis.IP)
        EmisMat{i,1} = padScreen;
    else
        EmisMat{i,1} = vec2mat(Emis.IP(i,2:end),6); %IP
    end

    EmisMat{i,2} = vec2mat(Emis.Ref(i,2:end),6); %Ref

end

% grease
Grease.IP(Grease.IP < -198) = NaN;
Grease.IP(Grease.IP == 0) = NaN;
Grease.Ref(isoutlier(Grease.Ref(:,2:end), 'mean')) = NaN;

GreaseMat = cell(length(Grease.Ref),2);

for i = 1:length(Grease.Ref)

    if i > length(Grease.IP)
        GreaseMat{i,1} = padScreen;
    else
        GreaseMat{i,1} = vec2mat(Grease.IP(i,2:end),6); %IP
    end

```

```

        GreaseMat{i,2} = vec2mat(Grease.Ref(i,2:end),6); %Ref
    end

    %%%%%%%%%%%%%%
    % chamber cold
    C.IP(C.IP < -198) = NaN;
    C.IP(C.IP == 0) = NaN;
    C.Ref(isoutlier(C.Ref(:,2:end),'mean')) = NaN;

    CMat = cell(length(C.Ref),2);

    for i = 1:length(C.Ref)

        if i > length(C.IP)
            CMat{i,1} = padScreen;
        else
            CMat{i,1} = vec2mat(C.IP(i,2:end),6); %IP
            CMat{i,1}(:,4:6) = ones(5,3)*NaN;
        end

        CMat{i,2} = vec2mat(C.Ref(i,2:end),6); %Ref
    end

    % chamber hot
    H.IP(H.IP < -198) = NaN;
    H.IP(H.IP == 0) = NaN;
    H.Ref(isoutlier(H.Ref(:,2:end),'mean')) = NaN;

    HMat = cell(length(H.Ref),2);

    for i = 1:length(H.Ref)

        if i > length(H.IP)
            HMat{i,1} = padScreen;
        else
            HMat{i,1} = vec2mat(H.IP(i,2:end),6); %IP
        end

        HMat{i,2} = vec2mat(H.Ref(i,2:end),6); %Ref
    end

    %%%%%%%%%%%%%%
    % ambient model raw
    RawModel.IP(RawModel.IP < -198) = NaN;
    RawModel.IP(RawModel.IP == 0) = NaN;
    RawModel.Ref(isoutlier(RawModel.Ref(:,2:end),'mean')) = NaN;

```

```
RawModelMat = cell(length(RawModel.Ref),2);

for i = 1:length(RawModel.Ref)

    if i > length(RawModel.IP)
        RawModelMat{i,1} = padScreen;
    else
        RawModelMat{i,1} = vec2mat(RawModel.IP(i,2:end),6); %IP
    end

    RawModelMat{i,2} = vec2mat(RawModel.Ref(i,2:end),6); %Ref

end

% ambient model predict
PredModel.Ref(isoutlier(PredModel.Ref(:,2:end),'mean')) = NaN;

PredModelMat = cell(min(size((PredModel.Ref))),1);

for i = 1:min(size((PredModel.Ref)))

    PredModelMat{i,1} = vec2mat(PredModel.Ref(i,2:end),6); %Ref

end

save('AnimationFrames1.mat','AmbMat','BlackMat','EmisMat','FiltMat','GreaseMat');
```

Published with MATLAB® R2018a