

```

%function [rt3,rt6,rt9,rt12] = response_time
function response_time
rt3 = zeros([600 1]);
rt6 = zeros([600 1]);
rt9 = zeros([600 1]);
rt12 = zeros([600 1]);

chanX = 1;
chanY = 2 ;

eventchan = 5;
visual_chan = 8;

[file, loc] = uigetfile('*.smr', 'Pick an SMR file');           % pointer to one of
the *.smr files to be analyze
if isequal(file,0)|isequal(loc,0)
    error('File open cancelled')
else
    disp(['File ', loc, file, ' found'])
end

cd(loc);                % generate list of files to be analyze. all and only
the *.smr files must be in one directory.
files = dir('*.smr');
for j = 1:length(files)
    count3 = 1;
    count6 = 1;
    count9 = 1;
    count12 = 1;
    fid = fopen(files(j).name);
    if fid < 1
        error('File not found');
    end

    [dataX, header] = SONGGetChannel(fid, chanX);
    [dataX] = transpose(SONADCToDouble(dataX, header));

    [dataY, header] = SONGGetChannel(fid, chanY);
    [dataY] = transpose(SONADCToDouble(dataY, header));

    samplerate = 1/header.sampleinterval;

    [trig, trig_header] = SONGGetEventChannel(fid, eventchan);
    if isempty(trig)
        error('ERROR ::: No event channel available'); % add which file
    end

    [visual_trig, trig_header] = SONGGetEventChannel(fid, visual_chan);
    if isempty(visual_trig)
        error('ERROR ::: No event channel available'); % add which file
    end

    for i = 1:length(trig)
        if (dataX(floor((trig(i))*samplerate))>10)
            event3(count3) = trig(i);
            count3 = count3 + 1;
        end
    end
end

```

```

elseif (dataX(floor((trig(i))*samplerate))<(-10))
    event9(count9) = trig(i);
    count9 = count9 + 1;
elseif (dataY(floor((trig(i)+0.5)*samplerate))>10)
    event12(count12) = trig(i);
    count12 = count12 + 1;
elseif (dataY(floor((trig(i)+0.5)*samplerate))<(-10))
    event6(count6) = trig(i);
    count6 = count6 + 1;
end
end

for i = 1:length(event3)
    temp = abs(visual_trig - event3(i));
    min1 = min(temp);
    clear temp;
    rt3(j*20+i) = min1;
end

for i = 1:length(event6)
    temp = abs(visual_trig - event6(i));
    min1 = min(temp);
    clear temp;
    rt6(j*20+i) = min1;
end

for i = 1:length(event9)
    temp = abs(visual_trig - event9(i));
    min1 = min(temp);
    clear temp;
    rt9(j*20+i) = min1;
end

for i = 1:length(event12)
    temp = abs(visual_trig - event12(i));
    min1 = min(temp);
    clear temp;
    rt12(j*20+i) = min1;
end
fclose(fid);
end

sample3 = find(rt3 < 1);
rt3 = rt3(sample3);

sample6 = find(rt6 < 1);
rt6 = rt6(sample6);

sample9 = find(rt9 < 1);
rt9 = rt9(sample9);

sample12 = find(rt12 < 1);
rt12 = rt12(sample12);

sample3 = find(rt3 > 0.2);
rt3 = rt3(sample3);

```

```

sample6 = find(rt6 > 0.2);
rt6 = rt6(sample6);

sample9 = find(rt9 > 0.2);
rt9 = rt9(sample9);

sample12 = find(rt12 > 0.2);
rt12 = rt12(sample12);

rt = [rt3(find(rt3 ~= 0));rt6(find(rt6 ~= 0));rt9(find(rt9 ~= 0));rt12(find(rt12
~= 0))];
group(1:length(find(rt3 ~= 0))) = '3';
group(length(find(rt3 ~= 0))+1 : length(find(rt6 ~= 0)) + length(find(rt3 ~=
0))) = '6';
group(length(find(rt6 ~= 0))+length(find(rt3 ~= 0))+1 : length(find(rt9 ~= 0)) +
length(find(rt3 ~= 0))+length(find(rt6 ~= 0))) = '9';
group(length(find(rt6 ~= 0))+length(find(rt9 ~= 0))+length(find(rt3 ~= 0))+1 :
length(find(rt12 ~= 0))+length(find(rt3 ~= 0))+length(find(rt6 ~=
0))+length(find(rt9 ~= 0))) = 'T';
f = figure;
boxplot(rt',group');
ylabel('Response Time (s)');
xlabel('Direction of Movement');
saveas(f,'ResponseTime','fig');
saveas(f,'ResponseTime','emf');

close all
clear

```