Import EPG data and Create heatmaps

Importing the data

Required files:

An initial.TextGrid file where the audio signal is segmented and annotated by allophone type (/ku/ vs. /ki/). The .TextGrid file then needs to be converted into a Table and saved with *Save as tab-separated file* in Praat (Boersma & Weenink, 2022).

A .csv file with the EPG frames in hexadecimal numbers.

A .txt file with the ordered list of stimuli.

Lookup table (mapping of hexadecimal numbers to EPG palate electrodes)

```
sortVec = [39 23 15 7 60 36 52 12 55 63 47 31 4 20 28 44 30 22 14 6 45 61 37 53 ...
    54 38 62 46 5 13 21 29 26 18 10 2 40 56 32 48 50 34 58 42 0 8 16 24 ...
    27 19 11 3 41 57 33 49 51 35 59 43 1 9 17 25] + 1;
```

Open table file saved from the .TextGrid

```
[filename, pathname] = uigetfile({'*.*'},'Open table file saved from .TextGrid');
if ~ischar(filename)
    return;
end

myGrid = readtable(fullfile(pathname, filename), 'FileType','text');
```

Open .csv file with EPG frames in hexadecimal numbers

```
[filename, pathname] = uigetfile({'*.csv'},'Open the .csv file with EPG frames');
if ~ischar(filename)
    return;
end
myFile = fullfile(pathname, filename);
ops = detectImportOptions(myFile);
ops.VariableTypes = {'double', 'char'};
epgData = readtable(myFile, ops);
epgData.Properties.VariableNames = {'Var1', 'Var2'};
```

Convert hexadecimal frames to a binary vector

```
binaryData = hexToBinaryVector(epgData.Var2);
```

Open the .txt file with stimuli order

```
[filename, pathname] = uigetfile({'*.txt'},'Open stimuli order file');
if ~ischar(filename)
    return;
end

myStimFile = readtable(fullfile(pathname, filename), 'ReadVariableNames', ...
    true,'NumHeaderLines',0);
```

Get corresponding frames for boundaries segmented in Praat

```
%Define frame rate
frameRate = 200;

%Get initial EPG frame
tmin2Frame = round(myGrid.tmin*frameRate);
%Get final frame
tmax2Frame = round(myGrid.tmax*frameRate);

%Initialize results table
resultats = cell(size(myGrid,1),1);

%Loop through .TextGrid intervals
for idx = 1:size(resultats, 1)

%Initialize data table
    currentDataStore = zeros([8, 8, tmax2Frame(idx)-tmin2Frame(idx)+1]);
    frameCount = 1;
%Loop through EPG frames of current interval
    for jdx = tmin2Frame(idx):tmax2Frame(idx)
        currentFrame = fliplr(binaryData(jdx,:));
        sortedEPG = currentFrame(sortVec);
        frameData = permute(reshape(sortedEPG, [8, 8]), [2, 1]);
        currentDataStore(:,:,frameCount) = frameData;
        frameCount = frameCount + 1;
    end
    resultats(idx) = {currentDataStore};
end
```

Data visualization

Create a heatmap of mean tongue/palate contact for the two allophones

```
%Uses previously imported Table, divides allophones based on tier
%annotation
myGroups = unique(myGrid.text);
for idx = 1:length(myGroups)
    currentGroup = myGroups(idx);
    currentInds = strcmp(currentGroup, myGrid.text);
    newData = resultats(currentInds);
    concatData = cat(3, newData{:});
    sumFrame = sum(concatData,3);
    imshow(imresize(rescale(sumFrame, 0, 1), 30))
    colormap jet
    title(currentGroup, 'Interpreter', 'Latex')
    exportgraphics(gca, [char(currentGroup) '_heatmap.jpg']);
    close
end
```

Create a heatmap of mean tongue/palate contact for each stimulus

```matlab
%Uses previously imported stimuli list
unicStims = unique(myStimFile.prompt);
for idx = 1:length(unicStims)
    currentStim = unicStims{idx};
    inds = find(contains(myStimFile.prompt, currentStim));
    frame1 = mean(resultats{inds(1)}, 3);
    subplot(3,6,idx)
    imshow(imresize(rescale(frame1, 0, 1), 30))
    colormap jet
    title(currentStim, 'Interpreter', 'Latex', 'FontSize',8)
end
exportgraphics(gcf, 'stims_heatmap.jpg');
```