
```

function temp_library = evalBox(S, P, T, row, col)

%specify box parameters
del_l_t = 30;
del_u_t = 50;
del_f = 30/2;
fanout = 3;

%declare temp_library
temp_library = [];

%set bounds
if(row+del_f > size(P,1)) % bottom - rows
    br_bound = size(P,1);
else
    br_bound = row+del_f;
end
if(col+del_u_t > size(P,2)) % right - cols
    rc_bound = size(P,2);
else
    rc_bound = col+del_u_t;
end
if(row-del_f < 1) % top - rows
    tr_bound = 1;
else
    tr_bound = row-del_f;
end
if(col+del_l_t > size(P,2)) % left - cols
    return; % if past matrix bound, we dont even need to worry about this
else
    lc_bound = col+del_l_t;
end

box_peaks = find(P(tr_bound:br_bound, lc_bound:rc_bound));
numPeaksBound = min(fanout, size(box_peaks,1));

for index = 1:numPeaksBound
    box_row = mod(box_peaks(index), ((br_bound-tr_bound)+1));
    if(box_row == 0) %% if edge case, set to edge (e.g. 21 mod 21 = 0 should have
        box_row = ((br_bound-tr_bound)+1);
    end
    box_col = floor(box_peaks(index)/((br_bound-tr_bound)+1)) + 1;
    %if(row+box_row-del_f > size(S,1) || col+del_l_t+box_col > size(S,2) ||...
    %    row+box_row-del_f < 0 || col+del_l_t+box_col < 0)
    %    do nothing, box is outside of S matrix bounds
    %else
        freq2 = S(tr_bound+box_row-1, col+del_l_t+box_col-1);
        temp_library = [temp_library; S(row,col), freq2, T(1,col), T(1, box_col)-T
    %end
end

```

end

*Error using evalBox (line 13)
Not enough input arguments.*

Published with MATLAB® R2014a