Converting RGB to YCbCr and Returning Reduced Image

Author: Antoine Steiblen Commented by: Shane Sarnac

```
function [saved,z1] = Compress_RGB_to_YCbCr(img)

% Read image and convert to ycbcr
rgb = imread(img);
ycbcr = rgb2ycbcr(rgb);

% determine the size of the matrix
[rows, columns, depth] = size(ycbcr);
Vert_mat = rows/8;
check_rows = Vert_mat - floor(Vert_mat);
extra_row = 0;

if check_rows > 0;
    extra_row = 1;
end

step_row = floor(Vert_mat) + extra_row;

Hor_mat = columns/8;
check_columns = Hor_mat - floor(Hor_mat);

extra_col = 0;

if check_columns > 0;
    extra_col = 1;
end

step_col = floor(Hor_mat) + extra_col;

ycbcr = ycbcr(1:(end-extra_row),1:(end-extra_col),:);

c = 1;
saved = 0;

% Create 8x8 blocks and reduce each block
% The saved variable counts the number of bytes saved with each 8x8 block
for j = 1:8:floor(Hor_mat)*8
    for i = 1:8:floor(Vert_mat)*8
            a = 7;
            b = 7;
            [reduced(c).data,z1,z2,z3] =  Reducer(ycbcr(i:(i+a),j:(j+b),:));
            x = [z1;z2;z3];
            [rowIdx,colIdx] = find(x);
            v = accumarray(rowIdx,colIdx,[],@max)';
            saved = saved + sum([64 64 64] - v);
            c = c + 1;
```

```
    end

end

display(saved);

% Reconstuct ycbcr matrix (from 8x8 parts)
Mat = [];
counter = 1;
for c = 1:(floor(Hor_mat))

    Col = [];

    for r = 1:floor(Vert_mat)
        Col = [Col;reduced(1,(counter)).data];
        counter  = counter + 1;
    end

    Mat = [Mat,Col];
end

% Convert back to rgb and show image
Mat = ycbcr2rgb(Mat);
figure;
imshow(Mat)



end
```

```
Error using Compress_RGB_to_YCbCr (line 8)
Not enough input arguments.
```

Published with MATLAB® R2014b