# Converting RGB to YIQ and Returning Reduced Image

Authors: Shane Sarnac and Antoine Steiblen

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

% Convert from rgb to yiq
rgb = imread(img);
yiq = rgb2ntsc(rgb);

% Determine the size of the matrix.
[rows, columns, depth] = size(yiq);
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;

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

c = 1;
saved = 0;

% Break up into 8x8 blocks, reduce them, and count the number of bytes
% saved (saved)
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(yiq(i:(i+a),j:(j+b),:));
            x = [z1;z2;z3];
            [rowIdx,colIdx] = find(x);
            v = accumarray(rowIdx,colIdx,[],@max)';
            [v1, v2, v3] = size(v);
            if v2 < 3;
```

```
                v(3) = 0;
            end
            saved = saved + sum([64 64 64] - v);
            c = c + 1;
    end

end

display(saved);

% Reconstruct the matrix from reduced form.
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 = ntsc2rgb(Mat);
figure;
imshow(Mat)



end
```

```
Error using Compress_RGB_to_YIQ (line 7)
Not enough input arguments.
```

*Published with MATLAB® R2014b*