
Part 4: Multi-Resolution Spline

```
images = ["../images/CARTOON.jpg", "../images/flowergray.jpg", "../images/kitty.jpg", ...
          "../images/polarcities.jpg", "../images/text.jpg" ];

% define the image pair
n = 3;
image1Index = [4,3,1];
image2Index = [5,4,2];
filter = [0.25,0.25; 0.25,0.25];

for i = 1 : n
    image1 = double(imread(images(image1Index(i)))));
    image2 = double(imread(images(image2Index(i)))));

    % Blend image
    blendedImage = zeros(size(image2));

    % Get mask
    bi_mask = ones(256,256);
    bi_mask(:,1:128) = 0;

    LP1 = laplacianPyramid(image1, filter);
    LP2 = laplacianPyramid(image2, filter);

    GR = gaussianPyramid(bi_mask, filter);
    LS = {};

    for j = 1: length(LP1)
        LS{end +1} = GR{j}.*LP1{j} + (1-GR{j}).*LP2{j};
        blendedImage = blendedImage + LS{end};
    end

    figure;
    imshow(blendedImage,[]);
end

function lp = laplacianPyramid(img, filter)
    [height, width] = size(img);
    side = log2(height);
    lp = {};

    for i = 1:side
        filteredImg = imfilter(img, filter, 'replicate', 'same');
        [h, w] = size(filteredImg);
        filteredImg = filteredImg(1:2:h, 1:2:w);

        % Get binary image
        biImage = imresize(filteredImg, 2, 'bilinear');

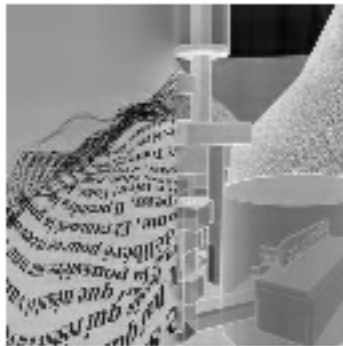
        % Resize
```

```
        lp{end+1} = imresize(biImage - img, [height,
width], 'bilinear');
        img = filteredImg;
    end
end

function gp = gaussianPyramid(img, filter)
    [height, ~] = size(img);
    side = log2(height);
    gp = {};

    for i = 1:side
        filteredImg = imfilter(img, filter, 'replicate', 'same');
        [height, width] = size(filteredImg);
        filteredImg = filteredImg(1:2:height, 1:2:width);

        % Resize
        gp{end+1} = imresize(filteredImg, 2^i, 'bilinear');
        img = filteredImg;
    end
end
```





Published with MATLAB® R2019b