```
% mx_lk_weibull - maximum likelihood test for an Apple using Weibull
% Distribution.
% Written by Katie Tsai and Matthew O'Connell

function mx = mx_lk_weibull(info,apple,x,low,high,removemin)
    % takes optional argument as to whether smaller connected objects
 are
    % removed.
    % For real images, removemin should be false.
    if nargin <= 5
        removemin = true;
    else
        removemin = false;
    end

    r = double(info(:,:,1)); g = double(info(:,:,2)); b =
 double(info(:,:,3));
    [rows,cols,map]=size(x);

    % Fit RGB histogram to Weibull Type XII distribution
    binwidth=30;
    graph_title = strcat(capitalize(apple), ' Apple');
    figure; hold on; grid on;
    r_hist = histfit(r(:),binwidth,'Weibull');
 set(r_hist(1),'facecolor','r','facealpha',.75);
 set(r_hist(2),'color','r');
    g_hist = histfit(g(:),binwidth,'Weibull');
 set(g_hist(1),'facecolor','g','facealpha',.75);
 set(g_hist(2),'color','g');
    b_hist = histfit(b(:),binwidth,'Weibull');
 set(b_hist(1),'facecolor','b','facealpha',.75);
 set(b_hist(2),'color','b');
    title(graph_title); xlim([0,256]); set(gca,'xtick',0:32:256);
 hold off;

    % Fit the Weibull distribution and calculate alpha, c, and k for
 sets of RGB data
    pd_r = fitdist(im2double(r(:)),'Weibull');
    pd_g = fitdist(im2double(g(:)),'Weibull');
    pd_b = fitdist(im2double(b(:)),'Weibull');

    % calculate R, G, B probability density for original image
    rx = x(:,:,1); gx = x(:,:,2); bx = x(:,:,3);
    pdf_rx=pdf(pd_r,0:255); prob_r=pdf_rx(rx+1);
    pdf_gx=pdf(pd_g,0:255); prob_g=pdf_gx(gx+1);
    pdf_bx=pdf(pd_b,0:255); prob_b=pdf_bx(bx+1);

    % calculate likelihood that a pixel is on histogram
    prod_pdf = (prob_r).*(prob_g).*(prob_b);
    prod_pdf=prod_pdf/(max(max(prod_pdf)));

    % Label connected objects in binarized image.
```

```
    pdf_filt = reshape(bwareafilt(imbinarize(prod_pdf),[low,high]),
[rows,cols]);
    [BW, num]=bwlabel(pdf_filt);

    %figure; hold on;
    %subplot 221; imshow(pdf_rx); title ('Filtered Product of Red
 PDF');
    %subplot 222; imshow(pdf_gx); title ('Filtered Product of Green
 PDF');
    %subplot 223; imshow(pdf_bx); title ('Filtered Product of Blue
 PDF');
    %subplot 224; imshow(x); title ('Original image of apples');  hold
 off;

    % Fill in connected objects in binarized image to better visualize
 apples.
    CC=bwconncomp(BW);
    numPixels = cellfun(@numel,CC.PixelIdxList);
    [mx,idx_mx] = max(numPixels);
    if(removemin == true)
        while (CC.NumObjects > 1)
            [mn,idx_mn] = min(numPixels);
            BW(CC.PixelIdxList{idx_mn}) = 0;
            CC=bwconncomp(BW);
            numPixels = cellfun(@numel,CC.PixelIdxList);
            [mx,idx_mx] = max(numPixels);
        end
    end

    % Apply filtered image with filled connected objects to original
 image.
    BW2 = imfill(BW, 'holes');
    %newX = x.*uint8(BW2);
    %blk = newX == 0; newX(blk) = 255;
    BW3=~BW2;
    r(BW3)=255; r=reshape(r,[rows,cols]);
    g(BW3)=255; g=reshape(g,[rows,cols]);
    b(BW3)=255; b=reshape(b,[rows,cols]);
    BW4=cat(3,r,g,b);

    figure; hold on;
    subplot 221; imshow(x); title ('1.  Original image');
    subplot 222; imshow(pdf_filt); title (['2.  Filtered image for '
 graph_title ' data']);
    subplot 223; imshow(BW2); title(['3.  Filling in the filtered
 image']);
    subplot 224; imshow(BW4); title('4.  Applying filter to original
 image'); hold off;
end
```

```
Not enough input arguments.

Error in mx_lk_weibull (line 15)
```

```
   r = double(info(:,:,1)); g = double(info(:,:,2)); b =
double(info(:,:,3));
```

Published with MATLAB® R2017a