

Report for HW2

About outputs

Output1 is from algorithm using LoGs in different scales.

Output2 is from algorithm using DoGs in different scales.

Output3 is from algorithm using a LoGs on images in different scales with edges detection.

Output4 is from algorithm using a LoGs on images in different scales without edges detection.

Some details in implementation:

1. Responses are normalized to 0 to 255. This makes threshold selecting a little bit easier.
2. Thresholds are set factitiously by myself according to the results. The outputs of a same kind are set the same threshold for comparison.
3. Initial size k_0 is set as 1 to avert too small blob, which are less meaningful.
4. Scaling rate k is computed by a expression with parameters so that biggest blob size can be controlled here.
5. The minimum of sizes of mask is 5 and the characteristic radius is about 2.5. These make blobs detected are really like blob while can detect small blobs.
6. In theory, normalization and characteristic radius can be computed by timing sigma. However, these approach leads to inaccuracy results. So I process normalization by keeping sum of absolute values in each filter the same and gain characteristic radius by counting negative values in middle line of a filter.
7. Non-maximum suppression in the same layer is processed by loops. I tried those 3 functions in documentation, but they were all slow.
8. Non-maximum suppression between different layers is processed by loops in a list form. It's fast and can achieve complex suppression condition.

Comparisons

459 circles



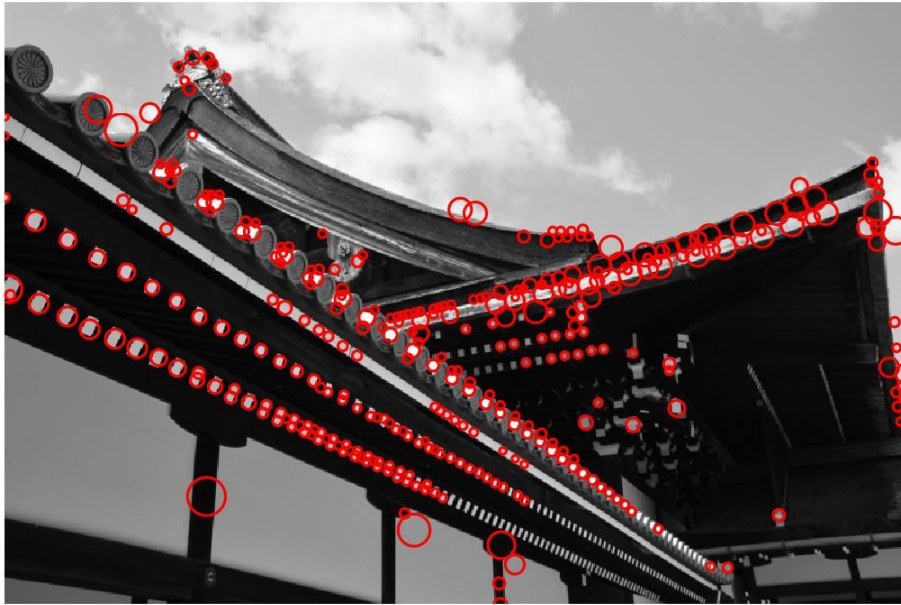
Result of LoG

424 circles



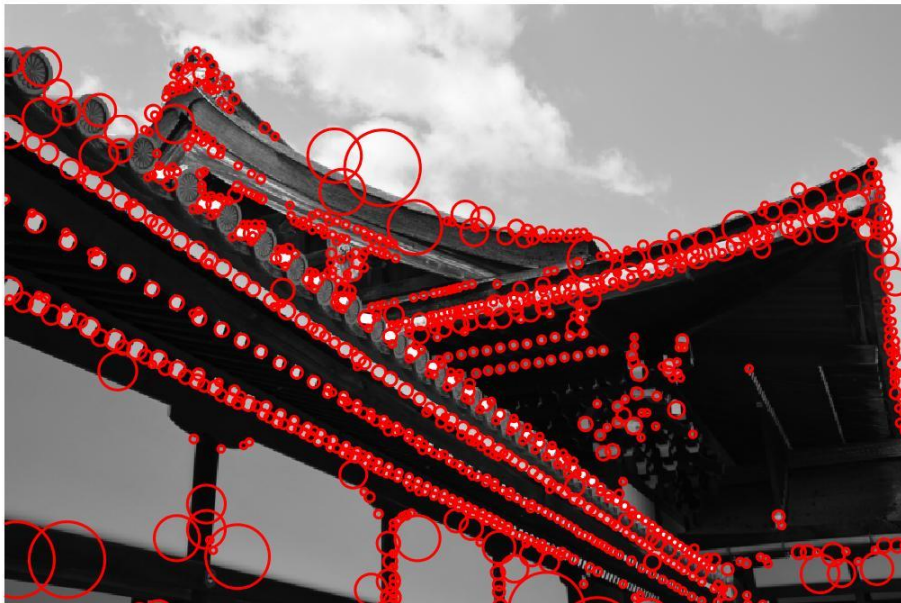
Result of Dog

344 circles



Result of downsample with edges detection

964 circles



Result of downsample without edges detection

imfilter	15	1.112 s	0.004 s	<div></div>
nonmaximum_suppression_matrix	15	0.591 s	0.591 s	<div></div>
nonmaximum_suppression_list	1	0.156 s	0.156 s	<div></div>
Main time of LoG				
imfilter	15	1.122 s	0.005 s	<div></div>
nonmaximum_suppression_matrix	15	0.625 s	0.625 s	<div></div>
nonmaximum_suppression_list	1	0.146 s	0.146 s	<div></div>

Main time of DoG

detect_edges	15	1.653 s	1.554 s	
imfilter	90	0.134 s	0.010 s	
nonmaximum_suppression_matrix	15	0.099 s	0.099 s	
nonmaximum_suppression_list	1	0.087 s	0.087 s	

Main time of downsample and edges detection

1. LoG vs DoG.

They are almost the same on both results and running time since I just use DoG filters to replace LoG filters.

2. Scaling filter vs downsample.

Scaling filter approach has better results, while downsample approach costs much little time. If edges detection is added to downsample approach, they have similar results and running times, but downsample approach can be set to a lower threshold, which makes it a little bit more robust when image was dark. But downsample approach leads to inaccuracy of center of large circles.

Other topics

1. For dark pictures with light parts, blobs in dark parts are hard to detect. There should be different threshold set for different parts of a picture.
2. The blobs detected for the same object in different pictures would vary.