**Contour Detection - Solution**

**Method Description：**

For Part 1(Warm-up), to reduce the image artifacts, I tried padding the image on its left, right, up, down directions 3 pixels, which fits the size of the filter. The convolution will have larger overlap area with the images, which will reduce the artifacts of the image boundaries. Then I used the gaussian filter from scipy.ndimage to achieve the gaussian filter, start from 1, I tried sigma = 2, sigma = 3. From 1 to 2, the performance increased, and from 2 to 3, the performance decreased. Eventually after multiple experiments I picked sigma = 2.5 as the best parameter. For the non-maximum suppression, I first calculate the angle (in radians) between the positive x-axis and the point (x, y) in the xy-plane with np.arctan2(). Then I continue to determine the direction of the edge at each pixel based on its gradient direction angle. This is done by comparing the gradient angle to a set of predefined threshold values, which correspond to four different edge directions: horizontal, vertical, and two diagonal directions, in other words, I round the gradient direction angle to 45\*n, n is the integer. However, even though I tried multiple parameters, the eventual result still have 0.001 distance to the final results.

**Precision Recall Plot:**

**图表, 折线图

描述已自动生成**

**Results Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method** | **Overall F-max score** | **Average max F-score** | **AP** | **Runtime(seconds)** |
| Initial implementation | **0.404** | **0.460** | **0.315** | **0.006** |
| Warm-up [remove boundary artifacts] | **0.424** | **0.483** | **0.359** | **0.006** |
| **Smoothing** | **0.520** | **0.522** | **0.333** | **0.007** |
| **Non-maximum Suppression** | **0.551** | **0.580** | **0.511** | **0.147** |
| **Val set numbers of best model [From gradescope]** | **0.551** | **0.580** | **0.511** | **0.147** |

**Visualizations:**

**黑白色的照片

中度可信度描述已自动生成**

**考拉在树上

描述已自动生成**

**在黑暗中

低可信度描述已自动生成**

My code works relatively well for the contours that was formed by large piece geometry pieces. However, when the image is shattered, or having condensed geometries, or especially with condense, line-like objects(like grass), my code does not behave so well. Also, my contour detected is not smooth enough and I also did not completely remove the artifacts. I think it might be the problem of implementation of NMS. Or the sigma of Gaussian filter.