Computational Photography Assignment 3

Single Michael
08-917-445

Task 1

Task 2

- Our given image I is an m by n
- Image I is monochromatic, i.e. there is only one color-channel.

Algorithm 1 Moving Average box filter

Input: Grayscale *Image* I resolution m by n
Output: *Image* $\hat{I}$

Procedures: $getDimensions(Image)$

1: $[h, w] = getDimensions(I)$
2: $r = \lceil \frac{w-1}{2} \rceil$
3: **Foreach** $Pixel\, p \in Image\, I$ **do**
4: $\quad contribution = 0$
5: $\quad$ **Foreach** $Pixel\, p_n \in r - Neighborhood\ \mathcal{N}_r(p)$ **do**
6: $\quad\quad contribution = contribution + I(p + p_n)$
7: $\quad$ **end for**
8: $\quad \hat{I}(p) = \frac{contribution}{m \cdot n}$
9: **end for**

Remarks:

- By pixels in the algorithn 1 we are referring to the coordinates of the pixel in the image.
- $I(p)$ denotes accessing the pixel-(color)-values in the images at the position of the pixel p in the image I.
- $\mathcal{N}_r(p)$ denotes the neighborhood with radius r around a given pixel p. In the context of pixel-coordinates, think of it as a box-grid, centered at the pixel coordinates of p. This grid has a radius of r. This means there are r neighbors (pixel-coordinates in the grid) below, on top, on the left and on the right of p.
- Our algorithm can easily be extended for color Images by simply applying the same algorithm to each color-channel seperatly.
- The assumption of being provided by a m by n can easily be extended for the case when $n \neq m$. This only will affect the computation of the radius r in algorithm 1.
  Computing $\lceil 0.5 \cdot \left(\lceil \frac{m-1}{2} \rceil + \lceil \frac{n-1}{2} \rceil \right) \rceil$ would be a valid option in order to compute r.
- If w (i.e. n) is odd, then $\lceil \frac{w-1}{2} \rceil$ is equal to $\frac{w-1}{2}$.

Aysmptotic Complexity

Task 3

Task 4

Task 5

Task 6