Exam December 4th, 2017

*Duration: 3h. Lecture notes are authorized. Answers should be **justified** briefly.*

1 Filtering (5 points)

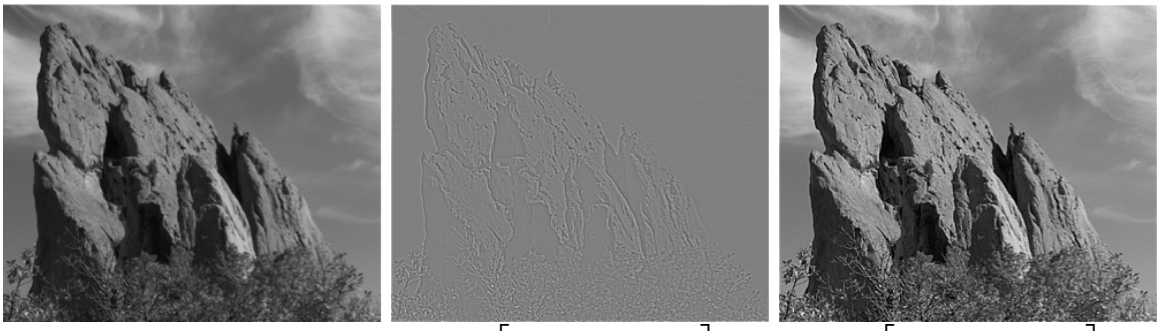
1.1 Smoothing filters

Input image Filter1 Filter2

In the above figure a Gaussian filter and a bilateral filter have been used to filter the input image.

1. Explain how Gaussian and bilateral filters differ.
2. In the figure identify which of the Gaussian and bilateral filters have been used to produce filter1 and filter2 (justify your answer).
3. Assuming that convolutions are performed with masks of size nxn, where n is an odd number, compare the number of operations required per pixel by brute force approaches with both filters.

1.2 Sharpen filter



Input image

$$1/8 \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \qquad 1/8 \begin{bmatrix} -1 & -1 & -1 \\ -1 & 16 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

The above figure shows results of convolutions of the input image with 2 filters whose masks are given below the images (note that an offset has been added to pixels in the middle image to ease visualisation).

1. What is the filter used with the middle image and what it does ?
2. What is the difference between the filters of the right and middle images ?
3. Deduce the principle of the sharpening operation performed here.

2 K-means

2.1 Color Compression (5 points)

We would like to compress a color image file by reducing the size of the pixel color information. This information is typically encoded with 8 bits per channel and over 3 channels (e.g. RGB). We would like to use K-means for that purpose.

1. Explain shortly the K-means method and what it optimizes.
2. We would like to encode pixel color information over 4bits (for the full set of colors):
   (a) Depict (pseudo-code) the corresponding k-means algorithm;
   (b) Once we have the reduced set of colors identified by the k-means algorithm, how do we transform the original image so that it presents only colors from the reduced set ?
   (c) how could we store the corresponding transformed image into a file ?
3. What is the interest of using the K-means reduced set of colors instead of a reduced set of uniformly distributed colors ?
4. What is the influence of the initial choice for the cluster centers ?

2.2 Point Segmentation (5 points)

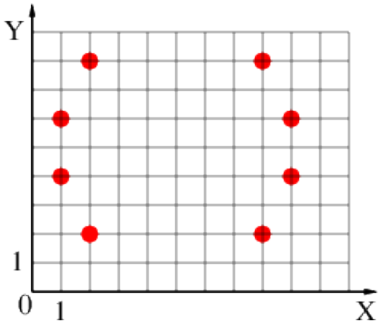


Figure 1: 8 input points.

The K-means algorithm is applied to the set of red points shown in Figure 1 with the Euclidean distance. The algorithm is initialized with $K = 2$ and with the centroids $(5, 2)$ and $(5, 8)$ in the plane where the origin is the lower left corner and where the grid size in Figure 1 is 1.

1. Detail the evolution of the algorithm and the final clusters obtained.
2. Is this solution a global minimum of the cost function optimized by K-means ? why ?
3. If we fix the grid size to 2 on the horizontal axis, *i.e.* graduations on the X axis are separated by a distance of 2, and 1 on the vertical axis, does it change the result ? If we fix the grid size to 1 on the horizontal axis and 2 on the vertical axis ?
4. If we choose the initial centroid locations randomly in the point set, does it have a chance to yield a better solution ?
5. In the case of image segmentation explain how color and spatial information can contribute to a K-means approach and their resulting effects on the segmentation.

3 Image Formation (5 points)

1. Why increasing the aperture size increases the blur in the captured images ?
2. Why bigger camera sensors are likely to provide better quality images ?
3. In a perspective projection why do objects further away appear smaller in the image ?
4. Given an object (perspectively) projected in an image how should I modify the focal length of the projection so that the size of the object in the image is divided by 2 ?
5. Assume that a cube of edge dimension e is observed with a perspective camera at distance d (from the projection center to the cube center) and a focal length f. Assume further that the cube is aligned with the image plane so that its projection forms 2 nested squares.
   (a) What is the exact length difference between the 2 square edges in the image ?
   (b) How should we modify the projection so that the smaller square gets bigger in the image while keeping the bigger square dimension constant, *i.e.* reducing the perspective effect ?