CS-E4850 Computer Vision

Exam 17th of December 2021, Lecturer: Juho Kannala

There are plenty of questions. Possibly many more than can be solved in the given time but answer as many as you can in the available time. The number of points awarded from different parts is shown in parenthesis at the end of each question. The maximum score from the whole exam is 42 points.

The exam must be taken completely alone. Showing or discussing it with anyone is forbidden!

1 Image filtering (6 p)

(a) Filter image J with the gaussian filter G using zero padding. (0.5 p)

$$J = \begin{array}{|c|c|c|}\hline 2 & 4 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 3 & 2 \\ \hline \end{array} \qquad G = \frac{1}{16}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

(b) Filter the above image J with a 3×3 median filter using zero padding. (0.5 p)

(c) Is it more efficient to filter an image with two 1D filters as opposed to a 2D filter? Why? How does the computational complexity relate to the size of the filter kernel (with K×K pixels) in both cases? (1 p)

(d) Is the following convolution kernel separable? If so, separate it. (0.5 p)

$$H = \begin{array}{|c|c|c|c|}\hline 4 & 2 & 6 & 8 \\ \hline 0 & 0 & 0 & 0 \\ \hline 2 & 1 & 3 & 4 \\ \hline 8 & 4 & 12 & 16 \\ \hline \end{array}$$

The bilateral filter consists of a domain kernel $d(i,j,k,l)$ and a range kernel $r(i,j,k,l)$. I is the original image. The coordinates (i,j) represent the pixel to be filtered and (k,l) the neighbouring pixels of the window centered in (i,j). σ_d and σ_r are smoothing parameters and $I(i,j)$ and $I(k,l)$ are the intensity of pixels (i,j) and (k,l) respectively.

$$d(i,j,k,l) = \exp\left(-\frac{(i-k)^2+(j-l)^2}{2\sigma_d^2}\right), \quad r(i,j,k,l) = \exp\left(-\frac{\|I(i,j)-I(k,l)\|^2}{2\sigma_r^2}\right)$$

$$I = \begin{array}{|c|c|c|c|c|}\hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \boxed{255} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

(e) Briefly explain the advantages of a bilateral filter compared to a Gaussian filter. (0.5 p)

(f) Construct 3×3 range- and domain kernels at the center pixel (i_c, j_c) of I marked with a box. Let $\sigma_d = \sigma_r = 1$. (1 p)

(g) The bilateral weight function is the multiplication of the range- and domain kernels. Construct the 3×3 bilateral weight function $w(i_c, j_c, k, l)$ and briefly discuss what limitation of a bilateral filter your result indicates. (1 p)

Your camera produces 1D images and your task is to detect edges in the images. Consider the example 1D image L below:

$$L = \begin{array}{|c|c|c|c|c|c|c|c|} \hline 133 & 132 & 121 & 110 & 100 & 80 & 61 & 60 \\ \hline \end{array}$$

(h) Propose a suitable kernel for edge detection in 1D images. (0.5 p)

(i) Using your kernel indicate where an edge would be detected in image L. (0.5 p)

2 Image formation (5 p)

Consider a camera with a camera projection matrix P and a 3D-point X in homogenous coordinates:

$$P = \begin{bmatrix} 5 & -14 & 2 & 17 \\ -10 & -5 & -10 & 50 \\ 10 & 2 & -11 & 20 \end{bmatrix} \qquad X = \begin{bmatrix} 4 \\ 2 \\ 2 \\ 1 \end{bmatrix}$$

(a) What are the 3D Cartesian coordinates of the point X? (0.5 p)

(b) Compute the Cartesian image coordinates of the projection of X. (0.5 p)

(c) We project point Z and get the following result $PZ = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^T$. What is the interpretation of the projection of the point Z? (1 p)

(d) Compute the Cartesian coordinates of the camera center. (1 p)

(e) Show that the two cameras $P_1 = K_1[R\,|T]$ and $P_2 = K_2[R\,|T]$ have the same camera center. (0.5 p)

Now we switch to an ideal pinhole camera with the following intrinsic parameters:

- 10 mm focal length
- Each pixel is 0.02 mm × 0.02 mm
- Pixel coordinates start at (0,0) in the upper left corner of the image.
- The image principal point is at pixel (500,500)
- No distortion

The world reference system is the same as the camera's canonical reference system (camera is at the world origin and pointed towards the positive z-axis).

(f) Calculate the intrinsic- and extrinsic matrix. (1 p)

(g) A point **X** has coordinates (50, 150, 800) centimeters in the world reference system. Compute the projection of the point into image coordinates. (0.5 p)

3 Triangulation (5 p)

Two cameras are looking at the same scene. The projection matrices of the two cameras are $\mathbf{P}_1$ and $\mathbf{P}_2$. They see the same 3D point $\mathbf{X} = (X, Y, Z)^\top$. The observed coordinates for the projections of point $\mathbf{X}$ are $\mathbf{x}_1$ and $\mathbf{x}_2$ in the two images, respectively. The numerical values are as follows:

$$\mathbf{P}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad \mathbf{x}_1 = \begin{bmatrix} 2 \\ 4 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 0 \\ -1.5 \end{bmatrix}.$$

(a) Present a derivation for the linear triangulation method and explain how $\mathbf{X}$ can be solved using that approach in the general case (i.e. no need to compute with numbers in this subtask). (1 p)

(b) Compute the 3D coordinates of the point $\mathbf{X}$ using the given numeric values for the camera projection matrices and image points. It is sufficient to just give the result. (Hint: You can calculate this with a computer or using pen and paper. In the latter case it may be easiest to write the projection equations in homogeneous coordinates by explicitly writing out the unknown scale factors, and to solve X, Y, Z and the scale factors directly from those equations.) (1 p)

(c) A third camera $\mathbf{P}_3$ is added to the scene. Describe how the linear triangulation method above can be extended to use the information from all the three cameras. (1 p)

(d) If there is noise (i.e. measurement errors) in the observed image coordinates of point $\mathbf{X}$, the linear triangulation method above is not the optimal choice but a nonlinear approach can be used instead. What error function is typically minimized in the nonlinear approach? (1 p)

(e) How does the nonlinear triangulation approach differ from the bundle adjustment procedure which is commonly used in structure-from-motion problems (i.e. how is the bundle adjustment problem different)? (1 p)

4 Local feature detection and description (4 p)

Below we have computed the gradients of an image at each pixel:

$I_x =$

3	2	1	-1	-1
4	3	2	0	-1
4	3	4	2	1
1	1	3	2	2

$I_y =$

2	3	1	1	-1
2	3	2	-1	-1
2	4	4	1	2
-1	0	3	2	3

(a) Compute the second moment matrix M for the coloured 3×3 window W. Assume that the weighting function w is a constant $w(x, y) = 1$

$$M = \begin{bmatrix} \sum_{x,y} w(x,y) I_x^2 & \sum_{x,y} w(x,y) I_x I_y \\ \sum_{x,y} w(x,y) I_x I_y & \sum_{x,y} w(x,y) I_y^2 \end{bmatrix}$$

(0.5 p)

(b) Compute the value of the corner response function when $\alpha = 0.04$:

$$R = det(M) - \alpha\, trace(M)^2$$

(0.5 p)

(c) How would you characterise the "cornerness"of window W and why? (1 p)

Let's assume that we detected SIFT regions from two images (i.e. circular regions with assigned orientations) of the same textured plane.

(d) What is the minimum number of SIFT region correspondence pairs needed for computing a similarity transformation between the pair of images? (1 p)

(e) How do we compute a histogram of gradient orientations when generating a SIFT descriptor? (1 p)

5 Geometric 2D transformations (4 p)

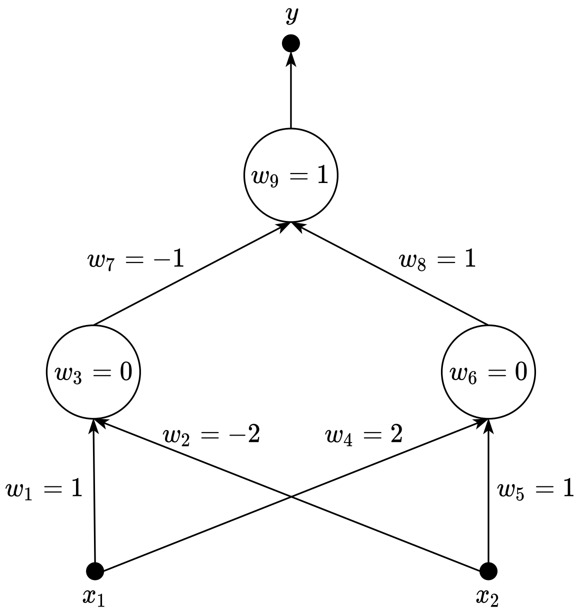
(a) A rectangle with corners $A = (-1, 1)$, $B = (1, 1)$, $C = (1, -1)$, $D = (-1, -1)$ is transformed by a transformation so that the new corners are $A' = (1, 3)$, $B' = (3, 3)$, $C' = (-2, 1)$, $D' = (-6, 1)$, respectively. An affine transformation does not explain the observations perfectly, but there is reason to believe that the transformation is affine and there is noise in the observations. Write down the equations to solve the transformation using the least squares method. **Note:** You don't actually have to numerically solve the transformation, just present a derivation of the solution. (1 p)

(b) Present a derivation of the direct linear transformation (DLT) algorithm for fitting a homography to the above set of four point correspondences. It is sufficient to explain how the solution can be computed but numerical solution is not required. (1 p)

(c) Describe the main stages of a RANSAC algorithm, which could be used to fit either affine transformation or a homography to a set of point correspondences. What is the minimal size of the random sample of correspondences in these two cases (i.e. affine and homography). Which geometric transformation model is used in panoramic image stitching and why? (2 p)

6 Neural networks and object detection (7 p)

The small neural net in the figure below uses ReLU as the nonlinearity at the output of each neuron. The values specified in the hollow circles are biases, and the values along the edges are gains.



(a) Are all the layers in the network above fully connected? (1 p)

(b) What is the output y from the net above when the input is as follows? (1 p)

$$x_1 = 2 \quad \text{and} \quad x_2 = 5$$

(c) What is the gradient $\mathbf{g}$ of the output y of the network above with respect to the weight vector

$$\mathbf{w} = [w_1,\ w_2,\ w_3,\ w_4,\ w_5,\ w_6,\ w_7,\ w_8,\ w_9]^T$$

when the input has the values given in the previous problem? Just give the result if you are confident of your answer. (2 p)

(d) With image data convolutional neural networks are much more popular than fully connected neural networks. Why is this? (1 p)

(e) Especially deep convolutional neural networks have proven to be effective. What function do the earlier layers (a.k.a. the base network) of a deep convolutional neural network serve and why are they often re-used from pre-existing networks such as VGG16. (1 p)

(f) SSD object detector evaluates only a small set (e.g. 4) of default boxes of different aspect ratios at each location. How can it detect large and small objects if the boxes are of fixed size? (1 p)

7 Feature tracking (5 p)

Let $I(\mathbf{x})$ and $J(\mathbf{x})$ be two grayscale images of the same scene taken from slightly different viewpoints and possibly slightly different orientations. We'd like to track a point $\mathbf{x}_I$ in image I to it's coordinate $\mathbf{x}_J$ in image J. That is we'd like to know the two dimensional displacement $\mathbf{d}^*$ of point $\mathbf{x}_I$ such that:

$$\mathbf{x}_J = \mathbf{x}_I + \mathbf{d}^*$$

To approximate $\mathbf{d}^*$ we look at a window (small square) $W(\mathbf{x}_I)$ of odd side-length 2h+1 pixels centered around the point $\mathbf{x}_I$ in image I and search for $\mathbf{d}$ that minimizes the dissimilarity between the windows in both images:

$$\mathbf{d}^* = \arg\min_{\mathbf{d}} \epsilon(\mathbf{d})$$

where the dissimilarity $\epsilon(\mathbf{d})$ is defined as a sum over the whole image $\mathbf{x} = (x_1, x_2)$:

$$\epsilon(\mathbf{d}) = \sum_{\mathbf{x}} [J(\mathbf{x}+\mathbf{d}) - I(\mathbf{x})]^2 w(\mathbf{x}-\mathbf{x}_I)$$

$w(\mathbf{x})$ is the indicator function of a $W(\mathbf{x})$:

$$w(\mathbf{x}) = \begin{cases} 1 & \text{if } |x_1| \leq h \text{ and } |x_2| \leq h \\ 0 & \text{otherwise.} \end{cases}$$

We assume that the motion of the camera between the two images is so small that the magnitude of $\mathbf{d}^*$ is much smaller than the diameter of $W(\mathbf{x}_I)$ and use an iterative approach so that we can formulate the problem as follows: find a step displacement $\mathbf{s}_t$ that, when added to $\mathbf{d}_t$, yields a new displacement $\mathbf{d}_{t+1}$ at each iteration t such that $\epsilon(\mathbf{d}_t + \mathbf{s}_t)$ is minimized. We add $\mathbf{d}_t$ into $\mathbf{x}$ as follows $J_t(\mathbf{x}) = J(\mathbf{x}+\mathbf{d}_t)$ and approximate the image function $J_t(\mathbf{x}+\mathbf{s}_t)(= J(\mathbf{x}+\mathbf{d}_t+\mathbf{s}_t))$ with its first-order Taylor expansion:

$$J_t(\mathbf{x}+\mathbf{s}_t) \approx J_t(\mathbf{x}) + [\nabla J_t(\mathbf{x})]^T \mathbf{s}_t$$

Minimizing $\epsilon(\mathbf{d}_t + \mathbf{s}_t)$ leads to a linear system of equations $A\mathbf{s}_t = \mathbf{b}$ where

$$A = \sum_{\mathbf{x}} \nabla J_t(\mathbf{x})[\nabla J_t(\mathbf{x})]^T w(\mathbf{x}-\mathbf{x}_I) \quad \text{and} \quad \mathbf{b} = \sum_{\mathbf{x}} \nabla J_t(\mathbf{x})[I(\mathbf{x}) - J_t(\mathbf{x})] w(\mathbf{x}-\mathbf{x}_I)$$

The overall displacement is then the sum of all the steps:

$$\mathbf{d}^* = \sum_t \mathbf{s}_t$$

(a) Show that minimizing $\epsilon(\mathbf{d_t} + \mathbf{s}_t)$ leads to a linear system of equations $A\mathbf{s}_t = \mathbf{b}$. (1 p)

NOTE: the problems (b)-(e) below don't require that you have solved problem (a).

Assuming a window size of 3×3 ($h = 1$) and an initial guess of displacement $\mathbf{d}_0 = [0, 0]^T$. For a particular value of $\mathbf{x}_I$, the two components of $\nabla J_0(\mathbf{x})$ inside the window $W(\mathbf{x}_I)$ are:

$$\frac{\partial J_0}{\partial x_1} = \begin{array}{|c|c|c|} \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline 10 & 10 & 10 \\ \hline \end{array} \quad \text{and} \quad \frac{\partial J_0}{\partial x_2} = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

and the difference between the two images is:

$$I(\mathbf{x}) - J_0(\mathbf{x}) = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

(b) Compute A and **b**. (1 p)

(c) Does the feature at $\mathbf{x}_I$ suffer from the aperture problem? Briefly justify your answer. (1 p)

(d) Give the minimum-norm solution $\mathbf{s}_0$ to the linear system $A\mathbf{s}_0 = \mathbf{b}$ (1 p)

(e) Assume that further iterations of the Lucas-Kanade algorithm do not change the solution $\mathbf{s}_0$ much. Does your answer to the previous question imply that the image motion between I and J at $\mathbf{x}_I$ is approximately horizontal? Briefly justify your answer. (1 p)

8 Camera calibration (6 p)

Camera calibration means that given a sufficient amount of points with known 3D coordinates and their image projections we can estimate the camera parameters. Let us consider a simplified case where only the height of the 3D points varies. That is to say we don't know the position of the rigidly mounted camera or its camera parameters but we know the x-coordinates(the z and y coordinates don't change) of some calibration points and their image projections. We measure the following data:

Calibration Point	x-coordinate	Image Coordinates (u,v)
Point 1	50 mm	(100,250)
Point 2	100 mm	(140,340)

(a) Assume a projective camera model and write the matrix equation that describes the relationship between world coordinates (x) (the height) and image coordinates (u, v) (the pixel coordinates where the point is projected in the image). Give your answer using homogeneous coordinates and a projection matrix containing the unknown camera parameters. (1 p)

(b) How many degrees of freedom does this transformation have? (0.5 p)

(c) How many calibration points and their associated image coordinates are required to solve for all of the unknown parameters in the projective camera model? . (0.5 p)

(d) Assume you have access to more calibration points and their associated image coordinates than required according to your answer in (c). Are the additional points useful or redundant? How would you solve for the parameters in this case where there are more points than required? (1 p)

(e) Assume that the camera is now calibrated. Given a new point and only its associated u image coordinate solve for the height (x-coordinate) of the point. Present the equation(s) that are used to solve for the height. (1 p)

(f) We now also have access to the associated v image coordinate of the new point, but calculating the height of the point using the v-coordinate gives a slightly different height then when using the u image coordinate. Is this a problem and how would you calculate the height in this case? (1 p)

(g) If in each calibration image we only measured the u pixel coordinate of the point, could the camera still be calibrated? If so, how many calibration points are required? If not, briefly describe why not. (1 p)