School of Computer Science and Communication, KTH
Lecturer: Mårten Björkman

EXAM

Image Analysis and Computer Vision, DD2423

Friday, 15^{th} of January 2015, 14.00–19.00

Allowed helping material: Calculator, the mathematics handbook Beta (or similar).

Language: The answers can be given either in English or Swedish.

General: The examination consists of Part A and Part B. For the passing grade E, you have to answer correctly at least 80% of Part A. If your score is less than 80%, the rest of the exam will not be corrected. Part B of the exam consists of **six** exercises that can give at most 50 points.

The results will be announced within three weeks.

Part A

Provide short answers to the questions! Each answer is worth maximum one point.

1. Using a drawing describe how the world gets projected to an image with a pin-hole camera model. Why isn't it possible to manufacture a real pin-hole cameras? *Answer: Light passes through a tiny hole and gets projected onto a plane on the other side of the hole. Since it has to be infinitesimally small, no light would pass through, if we were to product a real one.*

2. Why is it valuable to detect image discontinuities (edges), if you want to understand the 3D world through the study of images? *Answer: Discontinuities in the world get projected to discontinuities in images. Thus to learn about the the world, we better study image discontinuities.*

3. What benefits to you get by representing image and world points using homogeneous coordinates, instead of Cartesian coordinates? *Answer: Sequences of transformations and projections can be represented as products of matrices. Then you may represent things such as points in infinity.*

4. What kind of information exists in the phase and magnitude of the Fourier transform of an image? *Answer: The magnitude contains information on the what wave forms the image can be decomposed into, while the phase tells us where the wave forms are located.*

5. If you want to reduce the dimensionality of data using PCA, how can you tell how many dimensions you should use, for the errors introduced not to be too large? *Answer: To perform a PCA you compute the eigenvalues and eigenvectors of a covariance matrix. The sum of all eigenvalues for those dimensions that are ignored, tells us how much noise error is introduced in the process.*

6. What kind of imperfections does the intrinsic camera parameters try to compensate for? *Answer: They try to compensate for the fact that the lens system is not placed exactly on-top of the camera chip, with the centers in line.*

7. Why do you usually use bilinear interpolation for image transformations? *Answer: Because it prevents aliasing that would appear if you only use nearest neighbour pixel look-ups.*

8. A frequency space decomposition of a signal assumes that the signal is periodic, but that is not true for most images. Why can you still use Fourier transforms for images? *Answer: Because it is assumed that the images forever repeats itself for coordinates outside the image frame, which makes it all periodic, where an image is just one period.*

9. According to the Sampling Theorem, when can you assume to get aliasing in image sampling? *Answer: When the frequency content (or the bandwidth) of the image is larger than half the sampling frequency.*

10. What properties do Gaussian filters possess that make them suitable for scale-space representation? Mention at least two relevant such properties. *Answer: Linear shift-invariant, semi-group property, non-enhancement of local extrema*

11. How do you compute a second moment matrix that is used for corner detection? *Answer: First you compute the gradients* (f_x, f_y) *and then you square these and sum up in local windows*

$$\begin{pmatrix} \sum f_x^2 & \sum f_x f_y \\ \sum f_x f_y & \sum f_x^2 \end{pmatrix}$$

12. What kind of information does the SIFT descriptor contain? Think of how it is computed. *Answer: The SIFT descriptor contains a 2D array of small histograms of gradients, where each histogram represents different parts of an image window around the feature.*

13. Explain briefly how RANSAC works, if you for example like to detect a line. *Answer: You pick a minimum set of points, fit these to a model and then test the model against all other points. You do this multiple times and finally keep the model that most points can agree on.*

14. Why are template based methods rarely used for object recognition? *Answer: Because the appearance of an object tends to change a lot depending on from which orientation you look at at.*

15. In what sense is it usually harder to compute optical flow than binocular disparities? *Answer: Since the scale of the world is approximately known as well as the length of the baseline, you can predict the range of possible disparities. For motion however you have no such clues and the optical flow can be arbitrarily large.*

Part B

Exercise 1 (4+3=7 points)

1. Assume you have a 2D square in 3D Euclidean space, with corners $x_1 = (0,0,0)^\top$, $x_2 = (1,0,0)^\top$, $x_3 = (1,1,0)^\top$ and $x_4 = (0,1,0)^\top$. This square is projected onto two different cameras, camera A and camera B, that have the projection matrices

$$P_A = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 1 & 2 & 3 \\ 1 & 0 & 2 & 1 \end{pmatrix}, \; P_B = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 1 & 2 & 3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Apply the two projections and draw the square in the 2D image plane for each of the two cameras. How can you tell from the projections of the square, which camera is affine and which is perspective? How can you draw the same conclusion by directly studying the projection matrices?

Answer: With points in homogeneous coordinates, the projections in the first camera are

$$x_1^A = (0,3,1)^\top, \; x_2^A = (1,6,2)^\top \simeq (0.5,3,1)^\top, \; x_3^A = (3,7,2)^\top \simeq (1.5,3.5,1)^\top, \; x_4^A = (2,4,1)^\top$$

and for the second camera

$$x_1^B = (0,3,1)^\top, \; x_2^B = (1,6,1)^\top, \; x_3^B = (3,7,1)^\top, \; x_4^B = (2,4,1)^\top$$

Unlike camera A that is perspective, camera B is affine, since parallel lines in the world have been projected to parallel lines in the image. Furthermore, the matrix of camera B has a last row for which all elements are zero, except the last one. (1p know how to go to and from homogeneous coordinates, 1.5p get the perspective right, 1p answer one affine question, 0.5p answer also the second affine question)

2. In an image you observe a polygon with corners at $x_1 = (0,1)^\top$,$x_2 = (4,0)^\top$,$x_3 = (4,2)^\top$ and $x_4 = (0,3)^\top$. You believe the polygon is actually a square in 3D and want to transform the polygon into a square. Find an image transformation T that transforms the corners into a new set of corners with coordinates $y_1 = (0,0)^\top$,$y_2 = (1,0)^\top$,$y_3 = (1,1)^\top$ and $y_4 = (0,1)^\top$.

Answer: Given the shape of the polygon, it's clear that the transformation will be an affine one, which means that only 3 points needs to be considered. With the first, second and last corners in homogeneous coordinates and stacked into matrices we get

$$X = \begin{pmatrix} 0 & 4 & 0 \\ 1 & 0 & 3 \\ 1 & 1 & 1 \end{pmatrix} \text{ and } Y = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

What we search for is a 3×3 matrix T that satisfies $Y = TX$. We can compute the inverse of X.

$$X^{-1} = \frac{1}{8} \begin{pmatrix} -3 & -4 & 12 \\ 2 & 0 & 0 \\ 1 & 4 & -4 \end{pmatrix}$$

and conclude that

$$T = YX^{-1} = \frac{1}{8}\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}\begin{pmatrix} -3 & -4 & 12 \\ 2 & 0 & 0 \\ 1 & 4 & -4 \end{pmatrix} = \frac{1}{8}\begin{pmatrix} 2 & 0 & 0 \\ 1 & 4 & -4 \\ 0 & 0 & 8 \end{pmatrix}$$

We can test the transformation by observing that x_3 will in fact be transformed into y_3, as could be predicted. (1p properly set up problem, 1p get it almost right, 1p get it completely right).

Exercise 2 (3+2+3=8 points)

1. Computing the average of neighbouring points in an image can be seen as applying a box filter to the image. What is the Fourier Transform of the box filter $f(x)$ defined below?

$$f(x) = \begin{cases} 1, & -\frac{1}{2} \leq x \leq \frac{1}{2} \\ 0, & otherwise \end{cases}$$

Why does averaging often lead to undesirable effects in the resulting images, unlike for example what happens when you apply Gaussian filters?

Answer:

$$\hat{f}(\omega) = \int_{-1/2}^{1/2} e^{-i\omega x} = \left[\frac{1}{-i\omega}e^{-i\omega x}\right]_{-1/2}^{1/2} = \frac{1}{-i\omega}(e^{-i\omega/2} - e^{i\omega/2}) = \frac{2}{\omega}\frac{e^{i\omega/2} - e^{-i\omega/2}}{2i} = \frac{\sin(\omega/2)}{\omega/2}$$

This is a sinc function that has the undesirable property of being equal to zero for some particular frequencies, which means that some patterns in the original image can completely disappear, which is not natural. It's also strange that the the Fourier Transform can have negative values. (1p know what Fourier Transform is, 1p do it right, 1p know why undesirable)

2. How do you compute a convolution in the general case? What is the convolution of two box filters, that is $f(x) * f(x)$ given the definition of $f(x)$ above?

Answer:

$$f(x) * f(x) = \int f(\alpha) f(x-\alpha) d\alpha = \min(\frac{1}{2}, x+\frac{1}{2}) - \max(-\frac{1}{2}, x-\frac{1}{2}) = \begin{cases} 1-|x|, & -1 \leq x \leq 1 \\ 0, & otherwise \end{cases}$$

Drawing a figure makes it easier to see what the result of the integration should be. (1p know what convolution is, 1p do it right)

3. A box filter has the benefit of being easy to compute, since it's just a summation of neighbouring pixels, but there are other filters that are almost as simple. Compute the Fourier Transform of the triangular shaped filter $g(x)$ defined below.

$$g(x) = \begin{cases} 1-|x|, & -1 \leq x \leq 1 \\ 0, & otherwise \end{cases}$$

If a box filter has undesirable properties, does a triangular shaped filter also have problems? If any, what would those problems be?

Answer: Given the answer from the previous question and knowing that a convolution corresponds to a multiplication in the Fourier domain the conclusion can be drawn that

$$g(x) = f(x) * f(x) \Rightarrow \hat{g}(\omega) = \hat{f}(\omega)\hat{f}(\omega) = \frac{\sin^2(\omega/2)}{(\omega/2)^2}$$

Thus the triangular filter has exactly the same problem as the box filter. The problem can be solved by directly applying a Fourier Transform to $g(x)$, but that is much harder. (1p know to use multiplication, 1p do it right, 1p know why still undesirable.)

Exercise 3 (4+3+2=9 points)

1. Assume the histogram of a grey-level image is given by $p(z) = 2z^4 - 3z^2 + 8/5,\ z \in [0,1]$. Determine a transformation $z' = T(z)$, such that the histogram of the resulting image is $p(z') = 1,\ z' \in [0,1]$. For which grey-level values z does the transformation increase the contrasts and for which do they decrease?

   Answer: The transformation is given by

$$T(z) = \int_0^z p(\alpha)d\alpha = \left[\frac{2}{5}\alpha^5 - \alpha^3 + \frac{8}{5}\alpha\right]_0^z = (2z^5 - 5z^3 + 8z)/5$$

   Obviously, $T(0) = 0$ and $T(1) = 1$, which means that no additional adjustments have to be made. If $T'(z) = p(z) > 1$ you get a stretching, which increases the contrasts.

$$p(z) = 2z^4 - 3z^2 + \frac{8}{5} = 1 \rightarrow 2z^4 - 3z^2 + \frac{3}{5} = 0 \rightarrow z^2 = \frac{3}{4} \pm \sqrt{\frac{3^2}{4^2} - \frac{3}{10}} = \frac{3}{4} \pm \sqrt{\frac{21}{80}}$$

   Among all solutions only $z_1 = \sqrt{3/4 - \sqrt{21/80}} \approx 0.487$ is in $[0,1]$. By observing that $p(0) > 1$, while $p(1) < 1$, it can be concluded that contrasts increase for $z < z_1$ and decrease for $z > z_1$. (1p know to compute transformation, 1p do it right, 1p know condition for stretching, 1p do it right)

2. Propose an isotropic sharpening filter based on differential operators that leaves uniform image areas unchanged. What does isotropic mean in this case? In what sense is sharping different from contrast enhancement, such as the histogram equalization above?

   Answer: You may compute the Laplacian of the image, weight is with a constant k and subtract it from the image, that is $g(x,y) = f(x,y) - k\Delta f(x,y) = f(x,y) - k(f_{xx}(x,y) + f_{yy}(x,y)$. Nothing will happen to uniform areas, since for these $\Delta f(x,y)$ is zero. Isotropic here means that it is rotationally symmetric. Sharpening only affects the grey-level values around edges, unlike contrast enhancement that affects all pixels. (1p good suggestion for filter, 1p understand isotropic, 1p know difference)

3. On the 1D image

$$f(x) = [\ 2\ 5\ 4\ 3\ 6\ 10\ 12\ 11\ 14\ 10\ ]$$

   apply three different filters of size 3; a) a mean filter, b) a binomial filter and c) a median filter. Assume that pixels outside the left and right borders are equal to zero. Apply the filters individually, not in sequence.

   Answer: a)

$$[\ 7\ 11\ 12\ 13\ 19\ 28\ 33\ 37\ 35\ 24\ ]/3$$

   b)

$$[\ 9\ 16\ 16\ 16\ 25\ 38\ 45\ 48\ 49\ 34\ ]/4$$

   c)

$$[\ 2\ 4\ 4\ 4\ 6\ 10\ 11\ 12\ 11\ 10\ ]$$

   (0.5p per filter allowing minor mistakes, 0.5p all correct)

Exercise 4 (2+4+2=8 points)

1. Assume you have detected an image region, as indicated by the 1s in the figure below. Compute the zeroth and first order image moments of the region, given a suitable choice of coordinate axes.

0	0	0	1	1
0	0	1	1	1
0	0	1	1	0
1	1	1	0	0
0	1	1	0	0

Answer: Assuming a horizontal x-axis and vertical y-axis (going downwards) starting from 0, image moments are given by

$$m_{ij} = \sum_{x,y} f(x,y) x^i y^j$$

$$m_{00} = 1+2+4+3+2 = 12$$

$$m_{10} = 1\cdot 0+2\cdot 1+4\cdot 2+3\cdot 3+2\cdot 4 = 0+2+8+9+8 = 27$$

$$m_{01} = 2\cdot 0+3\cdot 1+2\cdot 2+3\cdot 3+2\cdot 4 = 0+3+4+9+8 = 24$$

(1p understand how to compute moments, 1p doing it right)

2. Obviously, the region is close to elliptic in shape and can be represented by a covariance matrix. Using image moments of higher order and the choice of coordinate axes you defined before, compute the covariance matrix. Also explain how you can use the matrix to compute the dominating orientation of the ellipse (without actually doing it on the matrix you just computed).

Answer: To compute the covariance matrix, we need the second order image moments too.

$$m_{20} = 1\cdot 0^2+2\cdot 1^2+4\cdot 2^2+3\cdot 3^2+2\cdot 4^2 = 0+2+16+27+32 = 77$$

$$m_{11} = (3+4)\cdot 0+(2+3+4)\cdot 1+(2+3)\cdot 2+(0+1+2)\cdot 3+(1+2)\cdot 4 = 0+9+10+9+12 = 40$$

$$m_{02} = 2\cdot 0^2+3\cdot 1^2+2\cdot 2^2+3\cdot 3^2+2\cdot 4^2 = 0+3+8+27+32 = 70$$

Now we can compute the matrix

$$C = \frac{1}{m_{00}^2}\begin{pmatrix} C_{xx} & C_{xy} \\ C_{xy} & C_{yy} \end{pmatrix}$$

with

$$C_{xx} = m_{20}\cdot m_{00} - m_{20}^2 = 77\cdot 12 - 27^2 = 195$$

$$C_{xy} = m_{11}\cdot m_{00} - m_{10}\cdot m_{01} = 40\cdot 12 - 24\cdot 27 = -168$$

$$C_{yy} = m_{02}\cdot m_{00} - m_{02}^2 = 70\cdot 12 - 24^2 = 264$$

and get

$$C = \frac{1}{144}\begin{pmatrix} 195 & -168 \\ -168 & 264 \end{pmatrix}.$$

(1p understand 2nd order moments are needed, 1p computing moments, 1p get matrix right, 1p explain hot get orientation)

3. Using the morphological structural element

0	1	0
1	1	1
0	1	0

compute an *opening* operation on the image region above. Assume that pixels outside the window are always set to 0, during all steps of the operation.

Answer: An opening operation is an erosion followed by a dilation. After erosion only one pixel survives,

0	0	0	0	0
0	0	0	1	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

which after dilation results in

0	0	0	1	0
0	0	1	1	1
0	0	0	1	0
0	0	0	0	0
0	0	0	0	0

(1p understand what opening means, 1p doing it right)

Exercise 5 (2+3+4=9 points)

1. What are the similarities and differences between K-means clustering and fitting of Gaussian Mixture models with points representing pixels in an image?

   Answer: Both methods represent a distribution of some points using a known number of clusters that are much fewer than the number of points. Points are iterative compared to clusters, where the similarities are used to update the centers and shapes of the clusters. However, they differ in the sense that for K-means a pixel only influences the closest cluster, whereas pixels influence all clusters with GMMs. Clusters are also spherical with K-means, while they are elliptic with GMMs. (1p one correct similarity that is not too trivial, 1p one correct difference that is not to trivial)

2. Methods using active contours (or snakes) are based on energy minimization. How do such methods represent a contour and what terms does the energy formulation typically consist of? Shortly describe at least two such energy terms. How do you go about to minimize the energy in practice, in order to get the final contour?

   Answer: A contour is represented by a 2D function $X(s),\ s \in [0,1]$, in the image plane. An energy formulation typically consists of at least two terms that work in opposite directions, such one term that penalizes long contours and another one that prefers contours along edges. When the energy is minimized these two different goals are balanced against each other. In practice, you divide the curve into points along the curve and iteratively try to move the points along the normal of the curve to gradually minimize the energy until convergence is reached. (0.5p know how a curve is represented, 1.5p two examples of terms, 1p describe how it is done)

3. With the density used for Mean Shift segmentation defined as

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} k(|x - x_i|^2),$$

   where $k(|x|^2)$ is some continuous density function, derive the update function that is used to maximize the density. Also explain how the result of using this update function can be used to get a segmentation of an image.

   Answer: The update function is given by

$$\nabla f(x) = \frac{2}{N}\sum_{i=1}^{N}(x - x_i)k'(|x - x_i|^2) = 0 \rightarrow x^{t+1} = \frac{\sum_{i=1}^{N} x_i\, k'(|x^t - x_i|^2)}{\sum_{i=1}^{N} k'(|x^t - x_i|^2)}$$

   Segmentation is done by starting at each pixel in the image, iterating until convergence and then see what convergence points the pixels will arrive to. Pixels that converge to the same point belong to the same segment in the final segmentation. (1p understand one has to derivate, 1p get update function right, 1p understand starting point is a pixel, 1p understand same convergence point means same segment)

Exercise 6 (3+3+3=9 points)

1. Assume you have a translating (but not rotating) robot moving around in a world and that you track two points, A and B, with a camera that has the focal length $f = 1$. At time t the points are located at image points $\mathbf{p}_A^t = (x_A^t, y_A^t, f)^\top$ and $\mathbf{p}_B^t = (x_B^t, y_B^t, f)^\top$ in homogeneous coordinates respectively. Show that using image positions from two points in time, $t = 1$ and $t = 2$, you can compute the translational direction $\mathbf{t} = (T_x, T_y, T_z)^\top$ in 3D space.

   Answer: There are different ways of showing this, some more complex than others. Point A has moved along a line between $\mathbf{p}_A^1$ *and* $\mathbf{p}_A^2$*, a line with equation* $\mathbf{l}_A = \mathbf{p}_A^1 \times \mathbf{p}_A^2$*. For point B there is a similar line* $\mathbf{l}_B = \mathbf{p}_B^1 \times \mathbf{p}_B^2$*. These lines intersect in the image at the vanishing point* $\mathbf{p} = \mathbf{l}_A \times \mathbf{l}_B$ *that encodes the direction of the translation* $\mathbf{t}$ *in 3D space. With the direction normalized to have a length equal to 1, the direction is thus given by* $\mathbf{t} = \mathbf{p}/|\mathbf{p}|$. (1p understand that intersection is needed, 1p computing intersection, 1p get it right)

2. Using a stereo system consisting of two cameras, with focal lengths $f = 1000$ pixels separated by a baseline $b = 10$ *cm*, you measure the binocular disparity d of an observed 3D point to determine its depth Z. If you want to have an accuracy in depth of 1 *mm* on the depth $Z = 1$ *m*, how accurately do you need to be able to measure the disparity?

   Answer: First find how an error in disparity leads to an error in depth.

$$Z = \frac{bf}{d} \Rightarrow \frac{\partial Z}{\partial d} = -\frac{bf}{d^2} = -\frac{Z^2}{bf} = -\frac{1^2}{0.1 \cdot 1000}\frac{m}{pixels} = -0.01\frac{m}{pix}$$

   Now the required accuracy in disparity can be written in terms of the referred accuracy in depth.

$$\Delta d = \frac{\Delta Z}{|\partial Z/\partial d|} = \frac{0.001}{0.01} pix = 0.1\ pix$$

   Thus you need to measure disparities with a precision of 1/10 pixels, which will be very hard. (1p understand relation between disparity and depth, 1p set up a formulation relating accuracy in depth and disparity, 1p get it right)

3. For two parallel cameras with a horizontal baseline and vertical y-axes the essential matrix has a particularly simple form. What is the relative rotation and translation between the cameras and what is the corresponding essential matrix?

   Answer: The rotation is $R = I$ *and the translation* $\mathbf{t} = (b, 0, 0)^\top$*. Thus the essential matrix becomes*

$$E = R[\,\mathbf{t}\,]_\times \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -b \\ 0 & b & 0 \end{pmatrix}.$$

   (1p able to express rotation and translation, 1p know what an essential matrix is, 1p get it right)