Version RC/4

EGT3
ENGINEERING TRIPOS PART IIB

Thursday 2 May 2019 2 to 3.40

Module 4F12

COMPUTER VISION

*Answer not more than **three** questions.*

All questions carry the same number of marks.

*The **approximate** percentage of marks allocated to each part of a question is indicated in the right margin.*

*Write your candidate number **not** your name on the cover sheet.*

STATIONERY REQUIREMENTS
Single-sided script paper

SPECIAL REQUIREMENTS TO BE SUPPLIED FOR THIS EXAM
CUED approved calculator allowed
Engineering Data Book

10 minutes reading time is allowed for this paper at the start of the exam.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed to do so.

1 (a) A grey scale image, $I(x, y)$, is first low-pass filtered (smoothed) by convolving it with two 1-D Gaussian filters before image gradients are computed as part of the feature detection process:

$$S(x, y) = \sum_{u=-n}^{n} \sum_{v=-n}^{n} g_\sigma(u) g_\sigma(v) I(x - u, y - v)$$

(i) Explain why smoothing is necessary before differentiation. [10%]

(ii) Show that the smoothing operation above is equivalent to convolving the image with a 2-D Gaussian. What are the advantages of implementing the convolution with two 1-D convolutions? [10%]

(b) Differentiation of the smoothed image, $S(x, y)$, can also be implemented with discrete convolutions.

(i) By first considering the Taylor series expansion of $S(x, y)$, show that approximations for the second-order derivatives, $\partial^2 S/\partial x^2$ and $\partial^2 S/\partial y^2$, can also be computed by convolving $S(x, y)$ with discrete 1-D convolutions. Identify the filter coefficients needed for each derivative. [20%]

(ii) Hence derive the 2-D filter needed to compute the Laplacian of the smoothed image, $\nabla^2 S(x, y)$. [10%]

(c) Consider an algorithm to localise image features for matching in 2-D images.

(i) The Laplacian of the smoothed image, $\nabla^2 S(x, y)$, can be used to localise *blob-like* features in both image position and scale. Explain how this can be implemented efficiently using an *image pyramid*. [15%]

(ii) The neighbourhood of each image feature is first geometrically normalised to a 16×16 patch of pixels by sampling pixels at an appropriate scale and orientation. How are the feature scale and orientation estimated? [15%]

(iii) The Scale-Invariant Feature Transform (SIFT) is then used to describe the 16×16 patches of pixels in order to match them in different images and over different viewpoints. Describe how this descriptor encodes the feature shape and how it achieves its invariance to lighting, image and viewpoint changes. What are its limitations? [20%]

2 The relationship between a 3-D world point $\mathbf{X} = (X, Y, Z)^T$ and its corresponding pixel at image co-ordinates (u, v) under perspective projection can be written using *homogeneous* co-ordinates by a *projection* matrix:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(a) (i) What is the algebraic and geometric significance of the variable s? [10%]

(ii) Express the projection matrix as the product of three transformations which encode the camera position and orientation, perspective projection onto a plane with focal length f, and the *internal* camera calibration parameters (principal point, pixels per unit length). Identify clearly the elements of each transformation matrix and the total number of parameters. [20%]

(iii) Show how to compute the camera position and orientation from a known projection matrix. [10%]

(b) Consider the restricted case of viewing 3-D world points which lie on the X-Y plane under perspective projection.

(i) Give an expression for the modified transformation as a projection matrix. How many degrees of freedom does this transformation now have? [10%]

(ii) Describe, using sketches, how a square on the world plane might appear in the image. You should be careful to account for each degree of freedom. [10%]

(iii) By first considering the *vanishing* points of lines in the plane, derive an algebraic expression for the plane's *horizon*. [20%]

(c) An Augmented Reality (AR) application running on a mobile phone camera must add computer-generated information onto the acquired image as the camera moves. Give details of how this is achieved if the viewing conditions (and hence the projection matrices) are initially unknown and changing. [20%]

3 Consider a stereo vision system where a 3-D point has image projections (u, v) and (u', v') in the left and right camera images. The 3-D point has co-ordinates $\mathbf{X}$ and $\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}$ in the left and right camera co-ordinate systems respectively. The internal calibration parameter matrices of the left and right cameras, $\mathbf{K}$ and $\mathbf{K}'$, are known.

(a) For a calibrated stereo system the corresponding points, (u, v) and (u', v') satisfy the *epipolar constraint* which is expressed with the *fundamental matrix* below:

$$\begin{bmatrix} u' & v' & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0$$

(i) Derive the epipolar constraint (as given above) and clearly show how the fundamental matrix can be factorised into matrices representing the camera motion (translation $\mathbf{T}$ and rotation $\mathbf{R}$) and the internal camera calibration parameters. [25%]

(ii) Give algebraic expressions for the *epipolar line* corresponding to a point in the left image with pixel coordinates (u, v) and for the *epipole* in the right image. [15%]

(b) Consider an uncalibrated pair of cameras with unknown positions and orientations.

(i) Describe a method for finding possible matches (point correspondences) between the left and right images. [10%]

(ii) How many point correspondences are required to estimate the fundamental matrix? [10%]

(iii) How is the transformation estimated when a large number of matches is available? [10%]

(c) The recovered fundamental matrix is to be used to compute the stereo camera geometry and the 3-D co-ordinates of visible points in the scene.

(i) Show how to recover the projection matrices for the left and right camera viewpoints from the fundamental matrix if the internal camera parameters, $\mathbf{K}$ and $\mathbf{K}'$, are known. Include details of how any ambiguity is avoided. [20%]

(ii) How are the 3-D positions of visible points in the scene recovered? [10%]

4 A computer vision expert has designed a system for signature recognition. The system takes in a *pair* of images of signatures $\mathbf{z}_1$ and $\mathbf{z}_2$ and outputs the probability that the signatures match. The system comprises three stages:

First, the two input images $\mathbf{z}_1$ and $\mathbf{z}_2$ are separately passed through a convolutional neural network (CNN) with parameters ϕ to yield two D-dimensional hidden vectors $\mathbf{h}_1 = \text{CNN}_\phi(\mathbf{z}_1)$ and $\mathbf{h}_2 = \text{CNN}_\phi(\mathbf{z}_2)$.

Second, scalar activation variables are produced by computing the squared difference between each dimension of the two hidden vectors, weighting the squared difference by a parameter w_d, and summing the results together with a bias parameter w_0,

$$a = w_0 + \sum_{d=1}^{D} w_d (h_{1,d} - h_{2,d})^2.$$

Third, the activation variables are passed through a logistic function to yield

$$p(y = 1|\mathbf{z}_1, \mathbf{z}_2, \phi, \mathbf{w}) = \sigma(a) = \frac{1}{1 + \exp(-a)}$$

where $y = 1$ indicates that the images contain matching signatures and $\mathbf{w} = [w_0, \ldots, w_D]$ is a vector containing the bias and the weights.

(a) Explain the purpose of each stage of the system. Your answer should describe why the expert is employing a CNN in the first stage, what the role of the parameters $\mathbf{w}$ is in the second stage, and why a logistic function is used in the third stage. [30%]

(b) The system is to be trained using N labelled pairs of training images $\{y^{(n)}, \mathbf{z}_1^{(n)}, \mathbf{z}_2^{(n)}\}_{n=1}^N$. Write down the log-likelihood of the parameters $\mathcal{L}(\phi, \mathbf{w}) = \log p(\{y^{(n)}\}_{n=1}^N | \{\mathbf{z}_1^{(n)}, \mathbf{z}_2^{(n)}\}_{n=1}^N, \phi, \mathbf{w})$ in terms of the logistic functions $\sigma(a^{(n)})$. [20%]

(c) The CNN parameters have been pre-trained, but the parameters $\mathbf{w}$ must be learned. Compute the derivative of the log-likelihood with respect to an element of the weight vector $\mathbf{w}$, i.e. $\frac{\mathrm{d}\mathcal{L}(\phi,\mathbf{w})}{\mathrm{d}w_d}$. [30%]

(d) Explain how the derivatives computed above can be used to train the parameters $\mathbf{w}$. Your answer should indicate what steps should be taken to handle large numbers of training examples N. [20%]

END OF PAPER

THIS PAGE IS BLANK