IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2018-2019

BEng Honours Degree in Computing Part III
BEng Honours Degree in Electronic and Information Engineering Part III
MEng Honours Degree in Electronic and Information Engineering Part III
MEng Honours Degree in Mathematics and Computer Science Part IV
BEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degrees in Computing Part III
MSc in Advanced Computing
MSc in Computing Science (Specialist)
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER C316

COMPUTER VISION

Tuesday 11th December 2018, 10:00
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators not required

1 Image filtering

a Given a 5x5 image as shown below, we would like to evaluate the image gradients by convolving the image with 3x3 Sobel filters h_x and h_y, which results in 3x3 output (without considering the boundaries).

1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5

image

$* h =$

filter

output

i) Write down the 3x3 horizontal Sobel filter h_x and vertical Sobel filter h_y.

ii) Calculate the 3x3 output when applying the Sobel filters h_x and h_y respectively.

b In the context of image filtering and interest point detection, we often perform Gaussian filtering before applying the Sobel filters.

i) Explain two reasons why Gaussian filtering is performed.

ii) For an image of size NxN and a Gaussian filter of size KxK, calculate the computational complexity using the big O notation when separable filtering is used.

c The reason that Gaussian filtering can be performed as separable filtering is that convolution satisfies the associativity property, i.e. $f * (h_1 * h_2) = (f * h_1) * h_2$. Prove the associativity property for 1D discrete convolution.

Recall that 1D discrete convolution is defined as, $f * h = \sum_{j=-\infty}^{+\infty} f[j]h[i-j]$. If we have $h[i] = h_1 * h_2 = \sum_{k=-\infty}^{+\infty} h_1[k]h_2[i-k] = \sum_{k=-\infty}^{+\infty} h_1[i-k]h_2[k]$, we can plug it into $f * h$ and obtain $f * (h_1 * h_2)$ to start the proof.

The three parts carry, respectively, 40%, 30%, 30% of the marks.

2 Feature descriptors

a In computer vision, the Laplacian of Gaussian (LoG) operation is often used for detection of interest points.

i) For a 2D function $f(x, y)$, write down the mathematical formulation for the Laplacian of the function.

ii) For a 5x5 image as shown below, calculate the 3x3 output when applying a Laplacian filter (without considering the boundaries).

1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5
1	2	3	4	5

image

$$* \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} =$$

filter

output

iii) Explain which two steps are performed in the LoG operation.

b Scale-invariant feature transform (SIFT) is a feature detection algorithm to detect and describe local features in images. It uses the difference of Gaussians (DoG) for interest point detection, which approximates the scale-normalised LoG. Let us define the Gaussian filter as,

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

i) Prove the following equation, also known as the heat diffusion equation.

$$\frac{\delta G}{\delta \sigma} = \sigma \nabla^2 G$$

where ∇^2 is the Laplacian operator.

ii) Let us define the difference of Gaussians as $DoG(x, y, \sigma) = G(k\sigma) - G(\sigma)$. Given that we can approximate the derivative $\frac{\delta G}{\delta \sigma}$ using the finite difference,

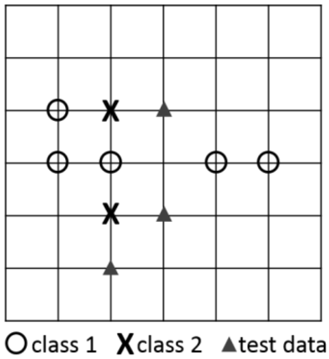
$$\frac{\delta G}{\delta \sigma} \approx \frac{G(k\sigma) - G(\sigma)}{k\sigma - \sigma}$$

prove that $DoG(x, y, \sigma)$ is a good approximation to the scale-normalised LoG, i.e. $\sigma^2 \nabla^2 G$.

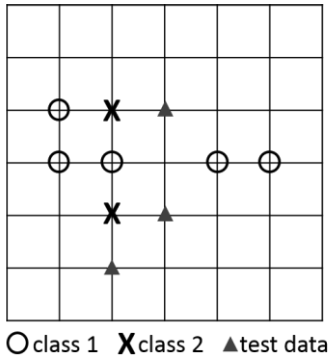
The two parts carry, respectively, 40%, 60% of the marks.

3 Image classification

a The figures below show a set of training samples from class 1 (circles) and class 2 (crosses), as well as two test samples (triangles).



(a) K = 1



(b) K = 3

i) Use the Euclidean distance as the distance metric and apply the K nearest neighbours (KNN) algorithm with K=1 to classify the test samples. Draw circles or crosses on subfigure (a).

ii) Use the Euclidean distance and apply KNN with K=3 to classify the test samples. Draw circles or crosses on subfigure (b).

iii) Which of the following best describes the computational cost of KNN?
A) High training cost and low test cost.
B) No training cost and high test cost.

b Support vector machine (SVM) is a supervised learning algorithm, which aims to maximise the margin between the support vectors. Training the SVM model can be formulated as the following optimization problem,

$$\min_{\boldsymbol{w},b} E(\boldsymbol{w},b) = \|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N} \max(0,\ 1 - \ y_i(\boldsymbol{w}\cdot\boldsymbol{x_i} + b))$$

where $E(\boldsymbol{w},b)$ denotes the loss function, $(\boldsymbol{x_i}, y_i)$ denotes the i-th training sample, $\boldsymbol{w}$ and b denote the parameters of the classifier, C denotes the regularisation hyper-parameter.

i) We would like to use the gradient descent algorithm to optimise the loss function $E(\boldsymbol{w},b)$. The second part of the loss function is called the hinge loss. Let us define the hinge loss $h(\boldsymbol{w},b) = \max(0,\ 1 - \ y_i(\boldsymbol{w}\cdot\boldsymbol{x_i} + b))$. Calculate the subgradient of the hinge loss $h(\boldsymbol{w},b)$ with respect to $\boldsymbol{w}$ and b.

ii) The original SVM algorithm only applies to binary classification problems. For a multi-class problem with m classes, if we use one-versus-one strategy for classification, how many binary SVM classifiers do we need to build? Given a test sample, how do we determine the classification result?

iii) For a multi-class problem with m classes, if we use one-versus-all strategy for classification, how many binary SVM classifiers do we need to build?

The two parts carry, respectively, 40%, 60% of the marks.

4 Neural networks

a In neural networks, the activation function defines the output of a neuron given an input.

i) The sigmoid function is a kind of activation function. It is defined as

$$f(x) = \frac{1}{1+e^{-x}}$$

Calculate the derivative of $f(x)$ with respect to x.

ii) Describe a problem with the sigmoid function when we train the neural network using the gradient descent algorithm.

iii) Rectified linear unit (ReLU) was proposed to address this problem. It is defined as

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

Calculate the subgradient of $f(x)$ with respect to x.

iv) Describe a problem with ReLU when x is negative.

b A convolutional neural network (CNN) consists of a number of convolutional layers. A convolutional layer applies the convolution operation to the input and pass the output to the next layer.

i) The input layer is a NxN grayscale image. If it is fully connected to the second layer, which has M neurons, what is the total number of connections between the input layer and the second layer (without considering the bias term)?

ii) If the second layer is a 3x3 convolutional layer, what is the number of weights for the second layer (without considering the bias term)?

iii) Which of the following is true about the convolutional layer?
A) It is translation-invariant.
B) It is rotation-invariant.

iv) For classification tasks, CNNs often use the softmax function in the last layer to convert values into the range of a probability. The softmax function is defined as,

$$p_i = \frac{e^{c_i}}{\sum_{k=1}^{K} e^{c_k}} \text{ for } i = 1,2,\cdots,K.$$

Calculate the gradient $\frac{\partial p_i}{\partial c_j}$ and represent it using p_i and p_j.

v) What is the most commonly used loss function for classification tasks using CNNs, which describes the distance between the one-hot encoding of the ground truth label $y = [y_1, y_2, \cdots, y_K]$ and the predicted probability $p = [p_1, p_2, \cdots, p_K]$. Write down its name and definition.

The two parts carry, respectively, 30%, 70% of the marks.