Solutions Sample Questions : Section 5 & 6.

Dr. Tony Scanlan

Machine Vision & Image Processing

Answer Q1:

(i) After multiplication with Eigenvectors the data is transformed to a new coordinate space and the covariance of the data is removed.

(ii) If the data is multiplied by a subset of Eigenvectors, then the dimensionality of the data is reduced.

(iii)

a. False: Eigenvectors are not obtained from the centered data matrix.
b. True: Eigenvectors are obtained from the sample covariance matrix.

Answer Q2:

An eigenvector v is a vector that when multiplied by a matrix A, is equal to a scalar λ times the eigenvector.

$$\boldsymbol{Av} = \lambda \boldsymbol{v} \qquad \textbf{(1)}$$

The above equation will only have a solution if the characteristic equation (below) is satisfied.

$$|\boldsymbol{A} - \lambda \boldsymbol{I}| = 0 \qquad \textbf{(2)}$$

Where I is the identity matrix and the straight brackets refer to the determinant of a matrix ($\det A \; = |\boldsymbol{A}| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$), we see that for a 2 x 2 matrix the determinant is simply the difference of the products along the diagonals.

In the example in the question where $\boldsymbol{A} = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix}$, we can write the characteristic equation as

$$|A - \lambda I| = \left|\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right| = 0$$

$$\left|\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix}\right| = 0$$

$$\left|\begin{pmatrix} -\lambda & 1 \\ -2 & -3-\lambda \end{pmatrix}\right| = -\lambda(-3-\lambda) - 1(-2) = \lambda^2 + 3\lambda + 2 = 0$$

If we solve the quadratic equation $\lambda^2 + 3\lambda + 2 = 0$, we get $\lambda_1 = -1, \lambda_2 = -2$. These values are the eigenvalues, now we want to find the associated eigenvectors. We can substitute the eigenvalues individually into the equation (1) to find the eigenvector $\boldsymbol{v_1}$.

$$\boldsymbol{Av_1} = \lambda_1 \boldsymbol{v_1}$$

$$\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \boldsymbol{v_1} = -\boldsymbol{v_1}$$

$$\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} v_{11} \\ v_{12} \end{pmatrix} = \begin{pmatrix} -v_{11} \\ -v_{12} \end{pmatrix}$$

$$\begin{pmatrix} v_{12} \\ -2v_{11} - 3v_{12} \end{pmatrix} = \begin{pmatrix} -v_{11} \\ -v_{12} \end{pmatrix}$$

This gives us two simultaneous equations $v_{12} = -v_{11}$ and $-2v_{11} - 3v_{12} = -v_{12}$. We note that both equations give us the same results $v_{12} = -v_{11}$ i.e. the elements of the first eigenvectors have equal magnitude but opposite sign. We can choose *any values* *(Note1)* that satisfy this e.g $\boldsymbol{v_1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

We now want to find the second eigenvector, as before we substitute the eigenvalue into the equation (1)

$$\boldsymbol{Av_2} = \lambda_2 \boldsymbol{v_2}$$

$$\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \boldsymbol{v_2} = -2\boldsymbol{v_2}$$

$$\begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix} = \begin{pmatrix} -2v_{21} \\ -2v_{22} \end{pmatrix}$$

$$\begin{pmatrix} v_{22} \\ -2v_{21} - 3v_{22} \end{pmatrix} = \begin{pmatrix} -2v_{21} \\ -2v_{22} \end{pmatrix}$$

Once again we have two simultaneous equations $v_{22} = -2v_{21}$ and $-2v_{21} - 3v_{22} = -2v_{22}$. Again we note that both equations give the same results $v_{22} = -2v_{21}$, i.e. the first element of the eigenvector v_{21} has half the magnitude but the opposite sign to the second element v_{22} . We again can choose any values that satisfy this i.e. $\boldsymbol{v_2} = \begin{bmatrix} 1 \\ -2 \end{bmatrix}$

We can verify that these eigenvectors are correct by multiplying them by **A** and λ.

$\lambda_1 = -1, \boldsymbol{v_1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

$$\boldsymbol{Av_1} = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$\lambda_1 \boldsymbol{v_1} = -1 \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Note1: We note that these are not the only eigenvectors we could have chosen that would have satisfied our equation. Typically, however we will choose eigenvectors that have a magnitude equal to 1 ($\boldsymbol{v_1} = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix}$). This is because only the direction information from the eigenvector is important, the eigenvalues denote the relative importance of the eigenvectors (as we have seen in the PCA & Eigenfaces lessons).

Note2: We can obtain eigenvalues and eigenvectors using python. See Colab note book here

Answer Q3:

In this case $A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$. We will use the same procedure as before finding the eigenvalues using the characteristic equation $|A - \lambda I|$ and then obtain the eigenvectors/eigenvalues.

$$|A - \lambda I| = \left| \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

$$\left| \begin{pmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{pmatrix} \right| = (2-\lambda)^2 - 1 = \lambda^2 - 4\lambda + 3$$

Roots of this equation are the eigenvectors we get $\lambda_1 = 3, \lambda_2 = 1$.

Substituting for $\lambda_1 = 3$ into the equation $\boldsymbol{Av} = \lambda \boldsymbol{v}$**,**

$$\boldsymbol{A}\boldsymbol{v}_{\mathbf{1}} = \lambda_1 \boldsymbol{v}_{\mathbf{1}}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \boldsymbol{v}_{\mathbf{1}} = 3\boldsymbol{v}_{\mathbf{1}}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} v_{11} \\ v_{12} \end{pmatrix} = \begin{pmatrix} 3v_{11} \\ 3v_{12} \end{pmatrix}$$

$$\begin{pmatrix} 2v_{11} + v_{12} \\ v_{11} + 2v_{12} \end{pmatrix} = \begin{pmatrix} 3v_{11} \\ 3v_{12} \end{pmatrix}$$

We again have simultaneous equations from the above matrix, which give the same result $v_{12} = v_{11}$. Thus the eigenvector $\boldsymbol{v}_{\mathbf{1}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ or we can normalise this to $\boldsymbol{v}_{\mathbf{1}} = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}$

Substituting for $\lambda_2 = 1$ into the equation $\boldsymbol{A}\boldsymbol{v} = \lambda \boldsymbol{v}$**,**

$$\boldsymbol{A}\boldsymbol{v}_{\mathbf{2}} = \lambda_2 \boldsymbol{v}_{\mathbf{2}}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \boldsymbol{v}_{\mathbf{2}} = \boldsymbol{v}_{\mathbf{2}}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix} = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$

$$\begin{pmatrix} 2v_{21} + v_{22} \\ v_{11} + 2v_{22} \end{pmatrix} = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$

We again have simultaneous equations from the above matrix, which give the same result $v_{22} = -v_{21}$. Thus the eigenvector $\boldsymbol{v}_{\mathbf{1}} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ or we can normalise this to $\boldsymbol{v}_{\mathbf{1}} = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}$

Answer Q4:

(I) We know that principle components analysis is based on finding the eigenvalues and eigenvectors of the sample covariance matrix $\boldsymbol{\Sigma}$. The sample covariance matrix is calculated as the product of the mean corrected data matrix and it's transpose $\boldsymbol{\Sigma} = \frac{1}{n-1}\boldsymbol{X}_c\boldsymbol{X}_c^T$

To find $\boldsymbol{X_c}$ we must first find the mean of the data along each dimension and subtract it from the data $\boldsymbol{X}$.

$$\mu_1 = \frac{1}{4}\sum_{i=1}^{4} x_{1i} = \frac{1}{4}(4 + 2 + 5 + 1) = 3$$

$$\mu_2 = \frac{1}{4}\sum_{i=1}^{4} x_{2i} = \frac{1}{4}(1 + 3 + 4 + 0) = 2$$

$$\boldsymbol{X_c} = \boldsymbol{X} - \boldsymbol{\mu}$$

$$\boldsymbol{X_c} = \begin{bmatrix} 4 & 2 & 5 & 1 \\ 1 & 3 & 4 & 0 \end{bmatrix} - \begin{bmatrix} 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 2 & -2 \\ -1 & 1 & 2 & -2 \end{bmatrix}$$

We can now calculated the sample covariance matrix.

$$\boldsymbol{\Sigma} = \frac{1}{n-1}\boldsymbol{X}_c\boldsymbol{X}_c^T$$

$$\boldsymbol{\Sigma} = \frac{1}{3}\begin{bmatrix} 1 & -1 & 2 & -2 \\ -1 & 1 & 2 & -2 \end{bmatrix} \times \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 2 & 2 \\ -2 & -2 \end{bmatrix} = \frac{1}{3}\begin{bmatrix} 10 & 6 \\ 6 & 10 \end{bmatrix} = \begin{bmatrix} 3.33 & 2 \\ 2 & 3.33 \end{bmatrix}$$

The eigenvectors of the sample covariance matrix are the principle components, we will obtain them using python.

For Eigenvalue λ_1 =5.33, the eigenvector $\boldsymbol{v_1} = \begin{bmatrix} 0.707 \\ 0.707 \end{bmatrix}$ and λ_1 =1.33 , $\boldsymbol{v_2} = \begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix}$

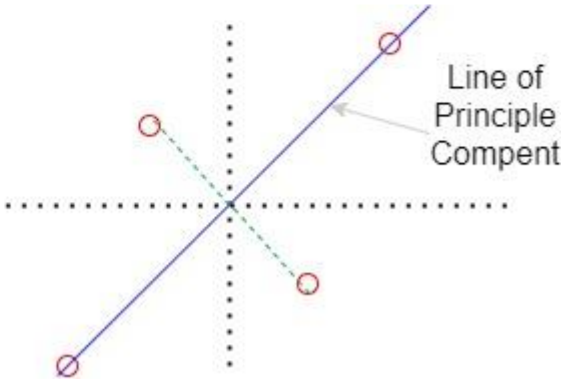
(ii) To reduce the dimensionality of the data we choose the eigenvector corresponding to the largest eigenvalue λ_1 =5.33.

(iii) In order to calculate the transformed data T, we must multiply the mean corrected data by $T = \boldsymbol{V}^T\boldsymbol{X_c}$

In this case this $\boldsymbol{V^T} = \boldsymbol{v_1}^{\boldsymbol{T}}$ as we only are multiplying by one eigenvector to reduce the dimensionality.

$$T = \boldsymbol{v_1}^{\boldsymbol{T}}\boldsymbol{X_c} = [0.707 \quad 0.707]\begin{bmatrix} 1 & -1 & 2 & -2 \\ -1 & 1 & 2 & -2 \end{bmatrix} = [0,0,2.82,-2.82]$$

A picture of the line corresponding to the eigenvector is shown in the figure below with the transformed data. It can be seen that the line intercepts two of the two outer points, the inner points are transformed to the origin.



Answer Q5:

The fisherface method is better at handling variation in Illumination & expression compared to Eigenfaces. The illumination causes large intra-class variation which can dominate inter-class differences. The LDA method uses class labels to find a transformation that preserves discrimination between classes.

Answer Q6:

(i) The statements are:

a. **True**: The bag of words algorithm does match similar documents by comparing word frequency.

b. **True:** The bag of features can be used for image classification, however it can't be used for object detection without modification.

c. **True:** An orderless representation is used, (despite the use of the orderless representation the bag of features is still good at image classification.)

(ii) We typically use the SIFT descriptor in the bag of features as it produces robust descriptors which can be used for clustering.

(iii) The first step in the bag of features approach is to <u>build a visual vocabulary</u>. To achieve this we first extract and represent the features from all images in a dataset. The features may be obtained with a feature detector or by random or uniform sampling. The features are then clustered by similarity into a visual vocabulary. Once the vocabulary is created we can then <u>assign terms</u> to the test image, we first extract the image features (in the same manner as was done for the visual vocabulary) and then assign features to the nearest visual words. We finally generate a term vector by recording the counts of each visual words to form a histogram.

(iv) We can perform image retrieval with the Bag of Features method by finding similar looking images based on the similarity of the Term vectors (histogram). This can be achieved with an inverted file index. Rather than keeping a list of the visual words in each image, we instead build a list of the images containing a specific visual word. Therefore we can quickly look up similar images to the test image. We may still apply a nearest neighbour search to find good matches.

(v) We can use the formula $t_i = \frac{n_{id}}{n_d} log \frac{N}{n_i}$ where the terms are as given in the question, so $n_{id} = 20, n_d = 2500, n_i$ =1520, $N = 750{,}000$

This gives $t_i = \frac{20}{2500} log \frac{750000}{1520} = 0.08 \times 6.02 = 0.49$

We can see that the log term increases the word frequency $\frac{n_{id}}{n_d}$ by a factor of 6.02, this is because the word occurs relatively infrequently in the database (1 in 500).

(vi) We can use the formula $t_i = \frac{n_{id}}{n_d} log \frac{N}{n_i}$ where the terms are as given in the question, so $n_{id} = 310, n_d = 3500, n_i$ =63,524, $N = 750{,}000$

This gives $t_i = \frac{310}{3500} log \frac{750000}{63524} = 0.089 \times 2.46$ =0.22

We can see in this case that the log term increases the word frequency $\frac{n_{id}}{n_d}$ by a small factor of 2.46, this is because the word occurs relatively frequently in the database (1 in ~10).

Answer Q7:

(i) The output of a neuron is the sum of inputs multiplied by the weights plus the bias, this sum then has the activation function applied. The equation for this process is $y_i = \sigma(\sum_{i=1}^{n} x_i w_i + b)$

We first multiply the weights by the inputs to determine

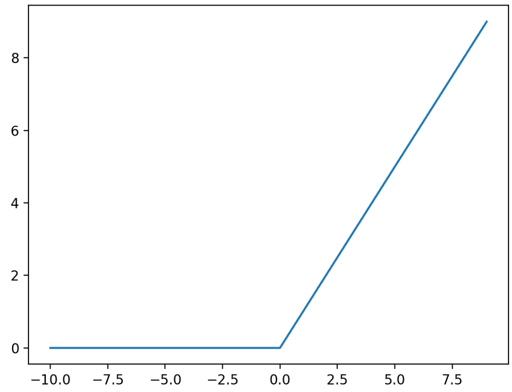
$$\sum_{i=1}^{n} x_i w_i = 0.9 \times 0.5 + 0.7 \times 0.4 + 0.2 \times -0.2 = 0.69$$

We then add the bias to this to determine $\sum_{i=1}^{n} x_i w_i + b = 0.69 + 0.3 = 0.99$

The final step is to apply the activation function, however we are told in the question that the activation function is <u>linear</u>, this means that $\sigma(z) = z$. Therefore

$$\sigma\left(\sum_{i=1}^{n} x_i w_i + b\right) = \sigma(0.99) = 0.99$$

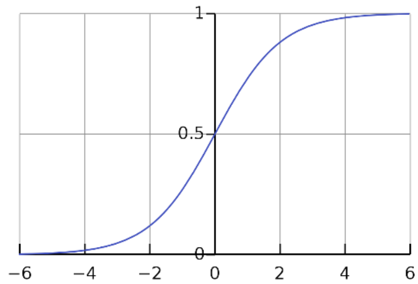
(ii) A rectified linear unit (as shown below) clips all negative values to zero while values above zero are processed linearly. Therefore an input of -0.5 will result in an output of zero.



(iii) The sigmoid activation unit relates output to input according to the given equation $\sigma(z) = \frac{1}{1+e^{-z}}$, for the input value of -0.1, we calculate:

$$\sigma(z) = \frac{1}{1+e^{0.1}} = 0.48$$

Note that the sigmoid activation function always produces a positive output value even for negative inputs as shown below.



#

(iv) The softmax activation function will produce a normalised probability value for each input according to the equation $\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$.

The first step is to obtain the individual exponential for each input (i.e. e^{z_i}): $e^2 = 7.38$, $e^{1.5} = 4.48$, $e^{2.2} = 9.03$, $e^{0.9} = 2.46$

We then obtain the sum of the exponentials $\sum_{j=1}^{K} e^{z_j} = 7.38 + 4.48 + 9.03 + 2.46 = 23.41$

The individual probabilities can then be worked out, by dividing by the sum of the exponentials. $\sigma(z_1) = 7.38/23.41 = 0.3$, $\sigma(z_2) = 4.48/23.41 = 0.19$, $\sigma(z_3) = 9.03/23.41 = 0.39$, $\sigma(z_4) = 2.46/23.41 = 0.11$

(v) If the learning rate is too high then during training large updates are applied to the parameters of the network leading to divergent behaviour (no convergence to trained network)

Question 8:

(i) Deep learning systems can be trained “End to End” this avoids developing handcrafted feature detectors and descriptors, instead many simple and complex features are learned automatically.

(ii) The deep learning revolution was due to advances in big data (providing complex data sets need to train deep learning models),Advances in Machine Learning (most of the details of how to build and train deep neural networks had been discovered) and High Performance computing (providing the computation platforms required to train deep neural networks)

(iii) There are a number of disadvantages of deep learning:

- It requires large amounts of training data that can be difficult to assemble and label
- It requires large computational resources to allow training of networks.
- Deep Learning based systems are trained on one source dataset, the network may not perform as well on data from another quite similar dataset (e.g. different perspective, illumination levels)
- It can be difficult to explain what image features a deep learning network is responding to, the performance of the network may not be particularly robust if conditions change and the failure mode could be difficult to understand.

Solutions Sample Questions : Section 3 & 4.

Dr. Tony Scanlan Machine Vision & Image Processing

Answer Q1:

(i) A good feature detector should find features that are local (robust to occlusion/clutter), Invariant (to the scale and rotation of the object), find a large no of features (quantity), be robust to noise and blur in the image and accurate (giving a precise location).

(ii) Brute force matching of descriptor vectors, looking for smallest Euclidean distance.

(iii) The value of sigma (in the equation generating the kernel values) is changed to detect blob regions at each scale. OR Can generate a Laplacian scale space and search for maxima.

(iv) The 4 key steps in the SIFT algorithm are: Scale Space Extrema Detection, Keypoint Localisation, Keypoint Characteristic Orientation, Keypoint Normalised descriptor.

(v) By including the characteristic orientation, keypoint descriptors (for the same feature) can be matched for different orientations of an object.

Answer Q2:

(i) We firstly can obtain results of convolution of the image at the indicated pixel with the x and y direction kernels (This is G_x , G_y).

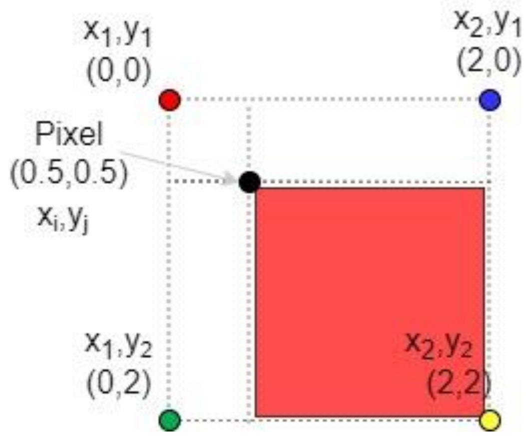
We can see that $G_x = -1 \times 8 + 0 \times 7 + 1 \times 6 = -2$ and $G_y = -1 \times 9 + 0 \times 7 + 1 \times 4 = -5$

The magnitude and direction are calculated using the simple formulae. $G = \sqrt{G_x^2 + G_y^2}$ and $\theta = tan^{-1}\left(\frac{G_y}{G_x}\right)$.

$$G = \sqrt{(-2)^2 + (-5)^2} = \sqrt{29} = 5.38$$

$$\theta = tan^{-1}\left(\frac{-5}{-2}\right) = 68^0$$

(ii) The division of the magnitude of pixel between 4 nearest histograms is performed in the SIFT method. We use a type of bilinear interpolation based on the distance to the histogram centres to work out the contribution of the magnitude μ to each histogram. The simplest way to understand how much of a contribution is made to each histogram is with a geometric picture. In the picture below we can see that the contribution to the histogram centred at the red pixel (0,0) is proportional to the red shared area (divided by the total area). This area is inversely proportional to the distance of the black pixel to the red histogram center, so when the black pixel is close to the red histogram center the shaded area is large, which means the magnitude contribution to the red histogram is also large as you would expect.



From this geometric picture the magnitude contribution to the histogram (centered at the red pixel) will be given by the shaded area divided by the total area:

$$\mu_{x_1,y_1} = \mu \left| \frac{(y_2 - y_i)}{(y_2 - y_1)} \frac{(x_2 - x_i)}{(x_2 - x_1)} \right|$$

In this case the magnitude assigned to the histogram will work out to be:

$$\mu_{x_1,y_1} = 10 \left| \frac{(2 - 0.5)}{(2 - 0)} \frac{(2 - 0.5)}{(2 - 0)} \right| = 5.6$$

Answer Q3:

(i) Supervised Learning is defined as the formalisation of learning from examples.
(ii) The key components are the dataset, representation and classifier.
(iii) Classification is when we want to map from input features to labels. Regression is when we want to find a direct relationship between input and output variables.

Answer Q4:

(i) With a high bias we are likely to see underfitting when training the model.
(ii) With high variance we are likely to see overfitting when training the model.
(iii) We can mitigate against overfitting with large dataset sizes, using a less complex classifier and regularising the parameters of the model.
(iv) Generalisation is the ability of the classifier to perform well on unseen data.

Answer Q5:

(i) From the information that we are given we can identify the number of True positives (TP) as 85, the number of False positives (FP) as 12, the number of False Negatives (FN) is 9. As the total number of predictions must equal the sum of the TP, FP, FN and True Negatives (TN) we can calculate the TN.

$$TN = Total\ Predictions - TP - FP - FN = 200 - 85 - 12 - 9 = 94$$

Now that we know all the predictions we can calculate accuracy precision and recall.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{85 + 94}{200} = \frac{179}{200} = 0.895$$

$$Precision = \frac{TP}{TP + FP} = \frac{85}{85 + 12} = 87.6\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{85}{85 + 9} = 90.4$$

(ii) With high precision but low recall, we would expect to see very few false positives and lots of false negatives.

(iii) With low precision but high recall, we would expect to see many false positives but a low number of false negatives.

Answer Q6:

(i) The Naïve bayes classifier operates by finding the class $\hat{y}$ that has the highest probability given by $\hat{y} = \underset{j \in \{1,..,K\}}{\operatorname{argmax}} p(c_j) \prod_{i=1}^{n} p(d_i|c_j)$

In this equation $p(c_j)$ is the probability of the occurrence of the class $c_j$in the dataset. We can see (from Totals column) that we have 500 "Bananas", 300 "Oranges" and 200 "Other". Thus the probabilities are

$$p("banana") = \frac{500}{1000} = 0.5$$

$$p("orange") = \frac{300}{1000} = 0.3$$

$$p("other") = \frac{200}{1000} = 0.2$$

The second term $\prod_{i=1}^{n} p(d_i|c_j)$ is the product of the individual probabilities of the feature d_i being present in class c_j. Thus in the case of the class "banana" we have $p(d_i|c_j) = p("long"|"banana")$ will have a probability of $400/500 = 0.8$, similarly for the other two features

$p("Sweet"|"banana") = \frac{350}{500} = 0.7$ and $p("Yellow"|"banana") = \frac{350}{500} = 0.9$

We can then evaluate $p(c_j) \prod_{i=1}^{n} p(d_i|c_j)$ for the case of $j = "banana"$ as

$$p("banana")\, p("long"|"banana")p("Sweet"|"banana")p("Yellow"|"banana") = 0.5 \times 0.8 \times 0.7 \times 0.9 = 0.252$$

We must also evaluate the probabilities for "Orange" and "Other" to see which class is the most likely.

$$("orange")\, p("long"|"orange")p("Sweet"|"orange")p("Yellow"|"orange") = 0.3 \times 0 \times 0.5 \times 0 = 0$$

$$("other")\, p("long"|"other")p("Sweet"|"other")p("Yellow"|"other") = 0.2 \times 0.5 \times 0.75 \times 0.25 = 0.01875$$

We can see that the banana class is $\frac{0.252}{0.01875} = 13.44$ times more likely to be a "banana" than "other", therefore we would give the fruit the label "banana".

(ii) As in the previous question we will go through the process of working out the terms of the equation $p(c_j)\prod_{i=1}^{n} p(d_i|c_j)$.

Firstly the $p(c_j)$ is the probability of the occurrence of the class in the database. For the two classes these are easily obtained from the totals in the table.

$$p(\text{"}labrador\text{"}) = 0.3$$
$$p(\text{"}chihuahua\text{"}) = 0.7$$

Next we must obtain the conditional probability of each feature belonging to each class $p(d_i|c_j)$ for each class.

$$p(\text{"}small\text{"}|\text{"}labrador\text{"}) = \frac{50}{300} = 0.167$$
$$p(\text{"}Not\ black\ coat\text{"}|\text{"}labrador\text{"}) = \frac{180}{300} = 0.6$$
$$p(\text{"}Not\ short\ hair\text{"}|\text{"}labrador\text{"}) = \frac{50}{300} = 0.167$$

The product of these and the $p(c_j)$ is then obtained

$$p(c_j)\prod_{i=1}^{n} p(d_i|c_j) = 0.3 \times 0.167 \times 0.6 \times 0.167 = 0.0052$$

For the Chihuahua class we similarly evaluate individual conditional probabilities and the product

$$p(\text{"}small\text{"}|\text{"}chihuahua\text{"}) = \frac{650}{700} = 0.93$$
$$p(\text{"}Not\ black\ coat\text{"}|\text{"}chihuahua\text{"}) = \frac{550}{700} = 0.78$$
$$p(\text{"}Not\ short\ hair\text{"}|\text{"}chihuahua\text{"}) = \frac{500}{700} = 0.714$$
$$p(c_j)\prod_{i=1}^{n} p(d_i|c_j) = 0.7 \times 0.93 \times 0.78 \times 0.714 = 0.36$$

We can see by inspection that the $chihuahua$ class is the most likely in this case.

Answer Q7:

In this image we will primarily have problems with intra class variation and occlusion:

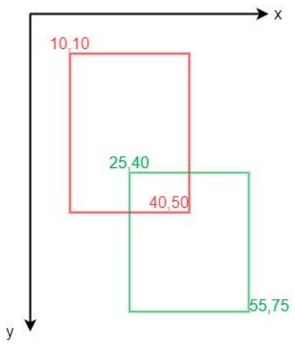
Answer Q8:

(i) A sliding window is necessary to find objects in images when the detector only can operate on a small patch of the image.

(ii) Detectors are often only trained one images patches of one size, using an image pyramid changes the size of the image so that large objects are reduced in size and can be detected by the detector. The problem that is solved by the image pyramid is Scale (i.e. objects appearing at different scales in images)

(iii) The main disadvantage is that computation time is increased with sliding windows and image pyramids.

(iv) The intersection over union is calculated as $IoU = \frac{area(b \cap b_g)}{area(b \cup b_g)}$ where b is the predicted bounding box, b_g is ground truth bounding box. In the diagram we are given 2 coordinates corresponding to the top left and bottom right for each of the bounding boxes, so we can easily determine the required areas as the product of the width × height of each area.



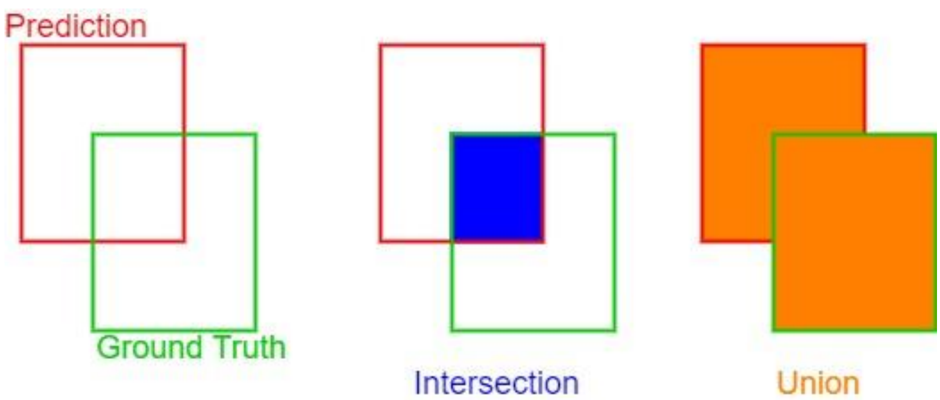
Firstly lets determine the "Intersection" area $area(b \cap b_g)$ between the two boxes. This is obtained from the smaller overlapping area between the two boxes (blue area in image below). We calculate this area as

$$area(b \cap b_g) = \text{width} \times \text{height} = (x_{max} - x_{min}) \times (y_{max} - y_{min})$$
$$= (40 - 25) \times (50 - 40) = 150$$

Secondly we must determine the "Union" area between the two boxes (see diagram below). This can be most easily calculated as the sum of the area of the two boxes minus the intersection area that we just calculated.

$$area\left(b \cup b_g\right) = area(b) + area\left(b_g\right) - area\left(b \cap b_g\right) = \left((40 - 10) \times (50 - 10)\right)$$
$$+ \left((55 - 25) \times (75 - 40)\right) - 150 = 1200 + 1050 - 150 = 2100$$

We finally then calculate the $IoU = \frac{area(b \cap b_g)}{area(b \cup b_g)} = \frac{150}{2100} = 7.1\%$



Answer Q9:

I. When using simple correlation for template matching, false local maxima are often found as high correlation values can occur with the template in regions of the image with high intensity.

II. Normalised Cross Correlation is computationally expensive as the mean value of each region under the template must be independently calculated for every template position in the image.

III. The shortest distance may return the same value when the shapes being matched are overlapping or non-overlapping but separated by a short distance.

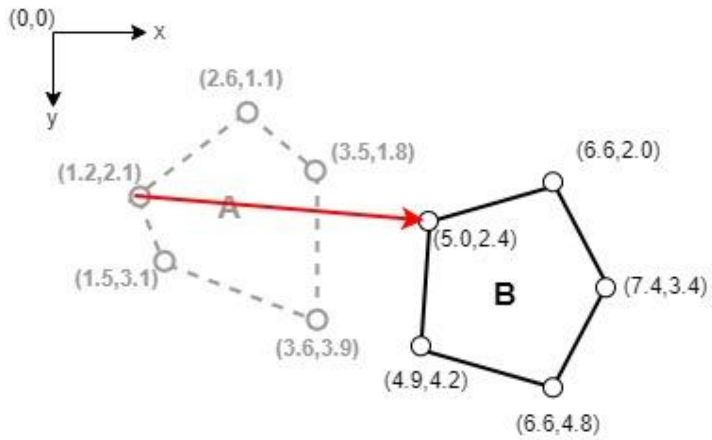
Answer Q10:

(i) The equation can be interpreted as a simple algorithm. For every $a \in A$ find the smallest distance to any point of b $\in B$, <u>keep the maximum distance among all points</u>. In the table below we carry out this algorithm. Let's assume we start with the element of A at $a = (2.6,2.1)$ and move clockwise through the other elements of A. We highlight the shortest distance between a and the points of b in each case. The distance $||a - b||$ is simply the Euclidean distance (or L2 norm) $||a - b|| = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2}$, where a_x, a_y and b_x, b_y are x, y coordinates of the points.

$a = (2.6,1.1)$	$\|\|a - b\|\|$
$b = (5.0,2.4)$	2.72
$b = (6.6,2.0)$	4.1
$b = (7.4,3.4)$	5.23
$b = (6.6,4.8)$	5.45
$b = (4.9,4.2)$	3.86
$a = (3.5,1.8)$	
$b = (5.0,2.4)$	1.61
$b = (6.6,2.0)$	3.10
$b = (7.4,3.4)$	4.2
$b = (6.6,4.8)$	4.3
$b = (4.9,4.2)$	2.7
$a = (3.6,3.9)$	
$b = (5.0,2.4)$	2.05
$b = (6.6,2.0)$	3.55
$b = (7.4,3.4)$	3.83
$b = (6.6,4.8)$	3.13
$b = (4.9,4.2)$	1.33
$a = (1.5,3.1)$	
$b = (5.0,2.4)$	3.57
$b = (6.6,2.0)$	5.22
$b = (7.4,3.4)$	6.67
$b = (6.6,4.8)$	5.37
$b = (4.9,4.2)$	3.57
$a = (1.2,2.1)$	
$b = (5.0,2.4)$	3.81
$b = (6.6,2.0)$	5.4
$b = (7.4,3.4)$	6.33
$b = (6.6,4.8)$	6.03

$b = (4.9,4.2)$	4.25

Reviewing the table we see that out of all the shortest distances obtained between the elements of a and b, the maximum distance of 3.81 occurs between $a = (1.2,2.1)$ and $b = (5.0,2.4)$. We show this distance on the image below. We note that instead of performing all the calculations above we could be selective and only calculate the distances for the elements of a where the shortest distance to b is likely to be a maximum.

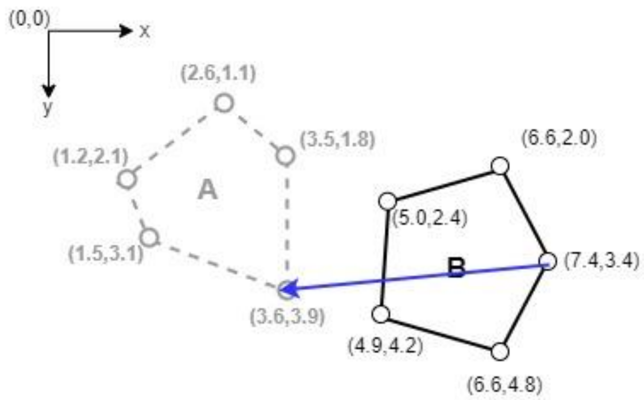


(ii) In order to obtain the Hausdorff directed distance from B to A $h(B, A)$, we apply the same algorithm but this time obtain the distances from the elements of B to A and once again look for the maximum from the shortest distances. To save effort we will only calculate the values for the elements of b at (6.6,2.0), (7.4,3.4) and (6.6,4.8) as these points are furthest from A and will contain the maximum shortest distance.

$b = (6.6,2.0)$	$\|a - b\|$
$a = (1.2,2.1)$	5.4
$a = (2.6,1.1)$	4
$a = (3.5,1.8)$	3.1
$a = (3.6,3.9)$	3.55
$a = (1.5,3.1)$	5.2
$b = (7.4,3.4)$	
$a = (1.2,2.1)$	6.33
$a = (2.6,1.1)$	5.77
$a = (3.5,1.8)$	4.22
$a = (3.6,3.9)$	3.83
$a = (1.5,3.1)$	5.9
$b = (6.6,4.8)$	
$a = (1.2,2.1)$	6.21
$a = (2.6,1.1)$	5.82
$a = (3.5,1.8)$	4.45
$a = (3.6,3.9)$	3.22
$a = (1.5,3.1)$	5.56

The maximum shortest distance 3.83 occurs from $b = (7.4,3.4)$ to $a = (3.6,3.9)$. We show the Hausdorff directed distance from B to A $h(B, A)$on the image below. We note that the

Hausdorff directed distance from B to A $h(B,A)$ is a different value to $h(A,B)$ and involves different points.



(iii) Now that we have obtained the directed distances, it is easy to obtain the overall Hausdorff distance, it is simply the largest value from the two directed distances. $H(A,B) = \max\big(h(A,B), h(B,A)\big) = \max(3.81{,}3.83) = 3.83$

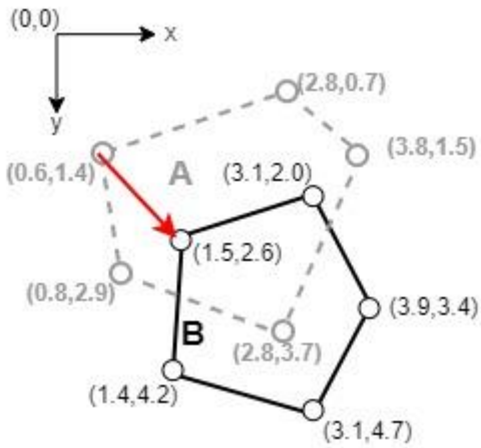
Answer Q11:

(i) In this case the shapes (point sets) are overlapping, we still apply the same algorithm and find the distance from each of the points of A to the points of B and record the shortest distance in each case.

$a = (2.8{,}0.7)$	$\|a - b\|$
$b = (3.1{,}2.0)$	1.33
$b = (3.9{,}3.4)$	2.92
$b = (3.1{,}4.7)$	4.01
$b = (1.4{,}4.2)$	3.77
$b = (1.5{,}2.6)$	2.30
$a = (3.8{,}1.5)$	
$b = (3.1{,}2.0)$	0.8
$b = (3.9{,}3.4)$	1.9
$b = (3.1{,}4.7)$	3.28
$b = (1.4{,}4.2)$	3.61
$b = (1.5{,}2.6)$	2.55
$a = (2.8{,}3.7)$	
$b = (3.1{,}2.0)$	1.83
$b = (3.9{,}3.4)$	1.14
$b = (3.1{,}4.7)$	1.04
$b = (1.4{,}4.2)$	1.49
$b = (1.5{,}2.6)$	1.7
$a = (0.8{,}2.9)$	
$b = (3.1{,}2.0)$	2.46
$b = (3.9{,}3.4)$	3.14
$b = (3.1{,}4.7)$	2.92
$b = (1.4{,}4.2)$	1.43
$b = (1.5{,}2.6)$	0.76
$a = (0.6{,}1.4)$	

$b = (3.1,2.0)$	2.57
$b = (3.9,3.4)$	3.86
$b = (3.1,4.7)$	4.14
$b = (1.4,4.2)$	2.91
$b = (1.5,2.6)$	1.5

From our calculations we see the maximum of the shortest distance is from $a = (0.6,1.4)$ to $b = (1.5,2.6)$ and therefore the hausforff directed distance from A to B $h(A, B) = 1.5$. We show this distance on the image below.



(ii) In this case the shapes (point sets) are overlapping, we still apply the same algorithm and find the distance from each of the points of A to the points of B and record the shortest distance in each case.

$b = (3.1,2.0)$	$\|\|a - b\|\|$
$a = (2.8,0.7)$	1.33
$a = (3.8,1.5)$	0.86
$a = (2.8,3.7)$	1.72
$a = (0.8,2.9)$	2.47
$a = (0.6,1.4)$	2.57
$b = (3.9,3.4)$	
$a = (2.8,0.7)$	2.9
$a = (3.8,1.5)$	1.9
$a = (2.8,3.7)$	1.14
$a = (0.8,2.9)$	3.14
$a = (0.6,1.4)$	3.8
$b = (3.1,4.7)$	
$a = (2.8,0.7)$	4.01
$a = (3.8,1.5)$	3.2
$a = (2.8,3.7)$	1.004
$a = (0.8,2.9)$	2.92
$a = (0.6,1.4)$	4.14
$b = (1.4,4.2)$	
$a = (2.8,0.7)$	3.77
$a = (3.8,1.5)$	3.61
$a = (2.8,3.7)$	1.49
$a = (0.8,2.9)$	1.43

$a = (0.6,1.4)$	2.91
$b = (1.5,2.6)$	
$a = (2.8,0.7)$	2.3
$a = (3.8,1.5)$	2.55
$a = (2.8,3.7)$	1.7
$a = (0.8,2.9)$	0.76
$a = (0.6,1.4)$	1.5

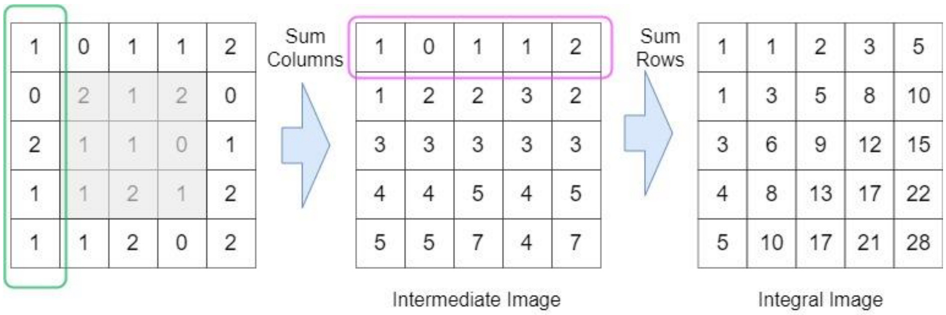
From our calculations we see the maximum of the shortest distance is from $a = (0.6,1.4)$ to $b = (1.5,2.6)$ and therefore the hausforff directed distance from A to B $h(A,B) = 1.5$. We show this distance on the image below.

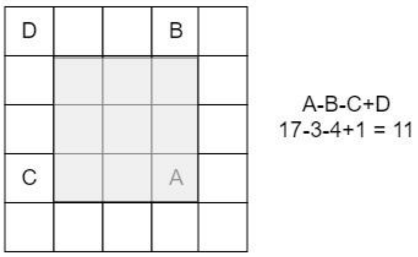
Answer Q12:

(i) The key enhancements or approaches within the Viola Jones Algorithm are:

- Integral Images for fast computation of feature detectors.
- Adaboost algorithm to select and combine weak classifiers.
- Cascade of classifiers for fast rejection of non-face patches.

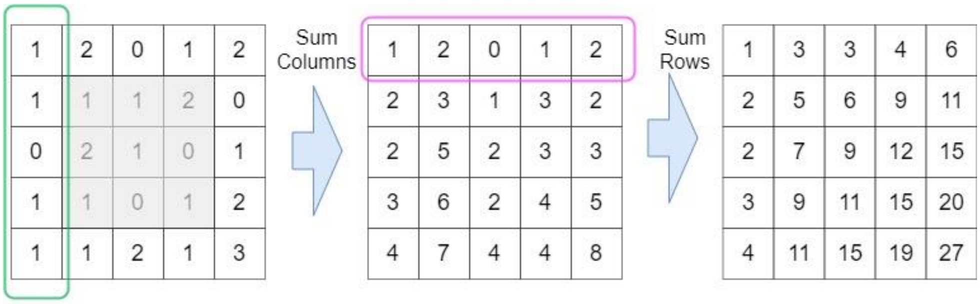
(ii) We can calculate the integral image and find the sum of the pixels under the shaded area as follows:

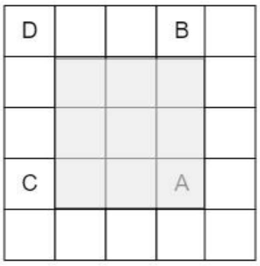




Location of values to use in Sum Calculation

(iii) We can calculate the integral image and find the sum of the pixels under the shaded area as follows.





A-B-C+D
15-4-3+1 = 9

Location of values
to use in Sum
Calculation

Answer Q13:

(i) The HOG representation is sensitive to fine scale edges (in silhouette contours).

(ii) The highlighted cell has an angle of $\theta = 45$ degrees. This means that we can divide it between bins with centres at 30^0 and 50^0. The magnitude is allocated between the two bins inversely proportionally to the distance from the bins (related to bilinear interpolation).

The magnitude allocated to the lower bin (30^0) i.e. μ_c^1 is given by $\mu\frac{|\theta_c^2-\theta|}{\theta_c^2-\theta_c^1}$. Where $\theta_c^2$is the larger bin centre (50^0) in this case θ_c^1 is the lower bin centre (30^0) and μ is the magnitude of the pixel which is 35.

$$\mu_c^1 = \mu\frac{|\theta_c^2-\theta|}{\theta_c^2-\theta_c^1} = 35\frac{|50-45|}{50-30} = 35\times\frac{5}{20} = 8.75$$

Similarly the magnitude contribution to the higher bin centre at 50^0 can be calculated as

$$\mu_c^2 = \mu\frac{|\theta_c^1-\theta|}{\theta_c^2-\theta_c^1} = 35\frac{|30-45|}{50-30} = 35\times\frac{15}{20} = 26.25$$

(iii) From the notes we know that we produce one histogram for each cell. Thus the number of pixels in a cell doesn't matter. Thus the length of the vector will be the number of cells in each block (i.e. 4), times the number of orientations in each histogram. There are $(6\times 10) = 60$ blocks, so we must further multiply by the number of blocks:

$$length\ of\ vector = 4\times 9\times 60 = 2160$$

Solutions to Sample Questions : Section 1 & 2.

Dr. Tony Scanlan

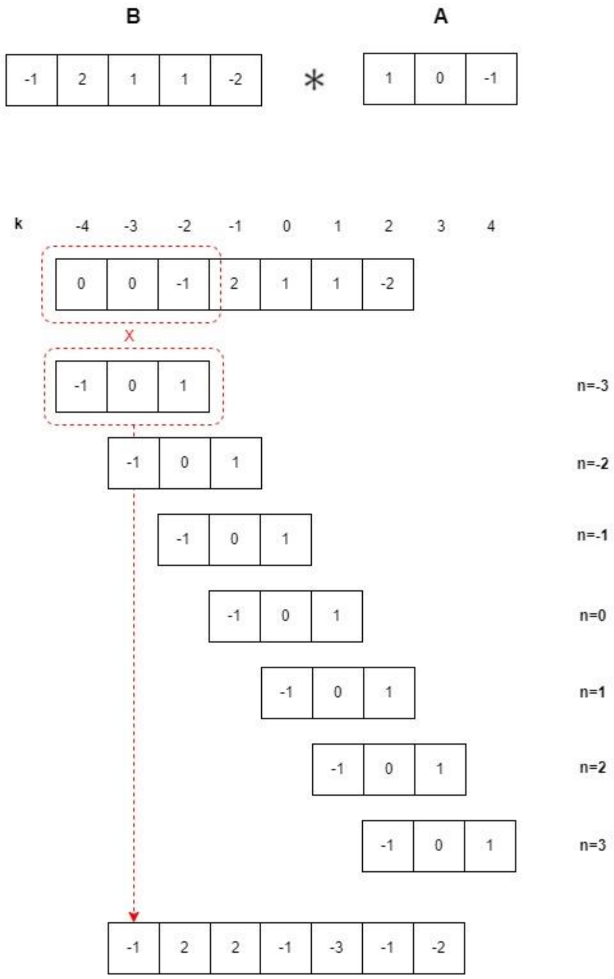
Machine Vision & Image Processing

Answer Q1:

(i) The two key stages are sampling and quantization.
(ii) The shutter is required to control the sampling instant of the camera. (An image is a sample in time as well as spatial sampling)
(iii) This is typically performed using a bayer pattern overlaid on the sensor.
(iv) Most modern display screen use the RGB colour space.
(v) The photoelectric effect is used to convert light to electrons in both types of image sensor.

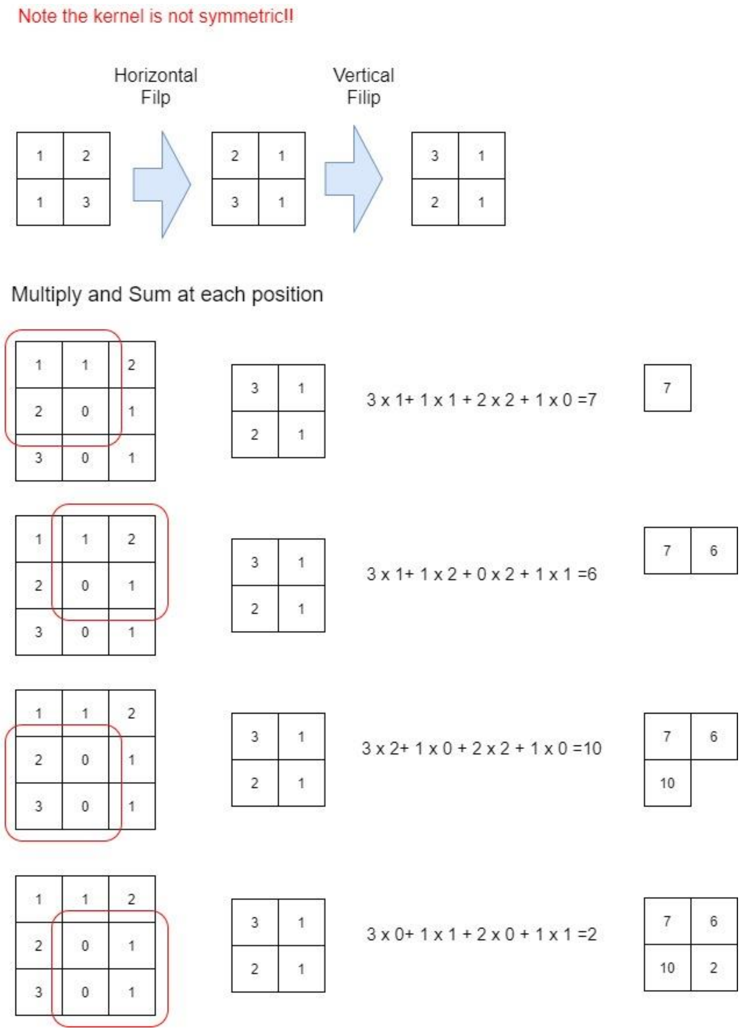
Answer Q2:

We can write the reversed and shifted kernel for each value of n as described by the term $h[n-k]$. The positive values of n result in a delayed kernel (shifted right) and the negative values of n shift of the kernel to the left. In order to calculate the final convolved values the sequence is multiplied by the reversed kernel in each position and the values summed.



Answer Q3:

This is a simple convolution, but we must be careful to check if the kernel is symmetric or not. It is good practice to check this by flipping it horizontally and vertically before performing the multiplications and sums. We do not include any padding in this case as we are told not to in the question.



Answer Q4:

The kernel is symmetrical in this case, so we can go ahead and multiply and sum in the position indicated by the red square.

Result is 11

Answer Q5:

We can use Relationship 1 from the Guide to Convolutional Arithmetic: For any input size i and kernel size k with unit stride $s = 1$. Then the output size o is given by:

$$o = (i - k) + 1$$

In our case we are told $i = 16$ and $k = 3$, then $o = (16 - 3) + 1 = 14$

Answer Q6:

We can use Relationship 3 from the Guide to Convolutional Arithmetic: For any input size i and kernel size k (must be odd) with unit strides, the padding is calculated as $p = \lfloor k/2 \rfloor$ (The half brackets denote a floor operation after division)

$$p = \lfloor k/2 \rfloor = \lfloor 5/2 \rfloor = 2$$

We can then then calculate the output size from the equation for $o = i + 2\lfloor k/2 \rfloor - (k - 1)$. Substituting values we get: $o = 24 + 2.2 - (5 - 1) = 24$. So we have verified the output matches the input when we use 2 rings of padding.

Answer Q7:

I. Only Frequencies up to half the sample rate can be sampled.

II. Aliasing is the phenomenon where high frequencies (greater than half the sample rate) appear as low frequencies during the sampling process. Aliasing is mitigated by employing low pass filtering before sampling to remove high frequencies that may be aliased.

III. Image Frequencies arise in a sampled system, as the set of samples that represent signals less than half the sample rate, could also represent a set of higher frequencies.

IV. The interpolation filter fills in missing part of the signal when reconstructing the signal. Perfect reconstruction can be theoretically achieved using a Sinc shaped filter.

V. If a sampled signal is reconstructed using linear interpolation, then the image frequencies may be present in the reconstructed signal. If the signal is then resampled, the image frequencies may be aliased causing distortion in the re-sampled signal.

VI. Frequency in images is measured as the change in pixel intensity. Rapid changes in intensity corresponding to edges or repeating patterns represent high frequency information.

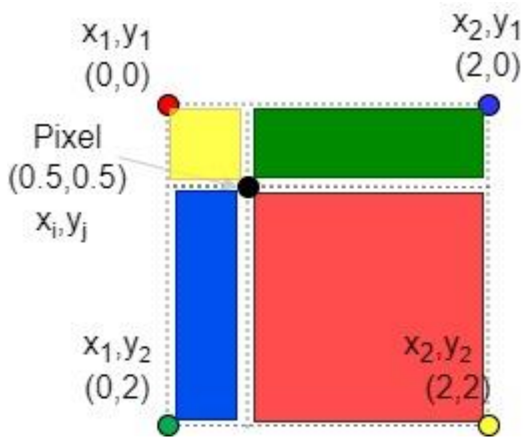
VII. Aliasing in a digital image can appear as a low frequency pattern superimposed on the image (where the incident light or original image contained high frequency information).

VIII. Nearest Neighbour interpolation has a sinc shaped response in the frequency domain. This means image frequencies are only partially removed. When the image size is reduced aliasing is likely to occur and this is why jagged lines due to aliasing are observed compared to using other image interpolation schemes.

IX. The Gibbs phenomenon is the presence of visible ringing around edges in an image and occurs when a Sinc Interpolation filter is used for image resizing. The Gibbs phenomenon occurs due to the practical implementation of the Sinc filter kernel. The kernel is truncated or finite in size and does not completely represent the Sinc function, this non-ideality results in the observed ringing in the image.

Question 8:

In bilinear interpolation the contribution of the intensity from each (original) pixel to the interpolated pixel is inversely proportional to the distance from the original pixel to the interpolated pixel. Rather than calculating distances, the simplest way calculate the bilinear interpolation is with the geometric approach. The contribution values from each original pixel may simply be determined as the area diagonally opposite the original pixel. For example the contribution from the red pixel is proportional to the red shaded area, similarly the contribution for the blue area can be determined for each original pixel.



We will calculate the shaded areas corresponding to each colour pixel (Note that for images the coordinate system has the origin in the top left corner):

$$A_{red} = (y_2 - y_i)(x_2 - x_i) = (2 - 0.5)(2 - 0.5) = 2.25$$

$$A_{blue} = (y_2 - y_i)(x_i - x_1) = (2 - 0.5)(0.5 - 0) = 0.75$$

$$A_{green} = (y_i - y_1)(x_2 - x_i) = (0.5 - 0)(2 - 0.5) = 0.75$$

$$A_{yellow} = (y_i - y_1)(x_i - x_1) = (0.5 - 0)(0.5 - 0) = 0.25$$

We then multiply the pixel intensities (given in the question) by the calculated areas and normalise the answer by the sum of the areas (i.e. the total area) to find the intensity of the interpolated pixel.

$$I_{x_i,y_i} = \frac{I_{red}A_{red} + I_{blue}A_{blue} + I_{green}A_{green} + I_{yellow}A_{yellow}}{A_{red} + A_{blue} + A_{green} + A_{yellow}}$$

$$I_{x_i,y_i} = \frac{10 \cdot 2.25 + 5 \cdot 0.75 + 12 \cdot 0.75 + 8 \cdot 0.25}{4} = 9.31$$

Question 9:

In order to perform separable convolution we must separate the kernel. (This may not always be easy or obvious!) In this case, we can see that the kernel is a Sobel kernel and is formed from the outer product of a 1D simple difference kernel and an averaging kernel.

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Each of the 1D kernels is then convolved sequentially with the image (we add 1 ring of zero padding as detailed in the question). The order of convolution does not matter due to the associative property of convolution.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 1 & 2 & 0 \\ 0 & 2 & 4 & 1 & 0 \\ 0 & 3 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 & -2 & -4 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 4 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -2 & -4 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 4 & 1 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -8 & -7 & -6 \\ 1 & 2 & 1 \\ 8 & 11 & 6 \end{bmatrix}$$

Question 10:

I. A kernel is separable if it can be formed from the product of two 1D kernels.

II. The superposition principle as applied to Image Processing can be defined as: Convolving the sum of two images with a kernel is the same as convolving the images with the kernel individually and summing the results. We could also state this as: The sum of an image convolved with two kernels individually is the same as convolution of the image with the sum of the kernels.

III. A Filter displays shift invariance if when the input is shifted spatially then the output will be shifted by the same amount.

IV. The commutative property of convolution means that the order of a convolution operation does not change the result.

V. The associative property of convolution means that when more than two images are kernels are convolved, it does not matter how the operations are grouped.

Question 11:

We can apply median filtering by getting the median value in each 3 x 3 region of the image as shown:

11	10	8	7	15
4	9	6	8	12
7	2	5	9	14
11	10	8	13	11
6	12	19	20	18

Values in first 3 x 3 region

11,10,8,4,9,6,7,2,5

2,4,5,6,7,8,9,10,11

Result of median Filtering of Image

7	8	8
7	8	9
8	10	13

Answer Q12:

Dilation: The border background pixels are converted from black to white by dilation increasing the size of the object.

Answer Q13:

Erosion: The border pixels of the object are converted from white to black by dilation decreasing the size of the object.

Answer Q14:

Opening: The opening operation applies Erosion followed by a dilation, thus small noisy regions of pixels are removed but the object size is also reduced. Following this with dilation restores the size of the object while noise remains removed.

Answer Q15:

Closing: The closing operation first applies dilation which fills in gaps in the object but increases it's apparent size. By then applying erosion the correct size of the object is restored.

Answer Q16:

(i) Edges can be detected by finding a high rate of change in pixel intensity. This may be achieved with derivative based detectors.

(ii) Edge detectors are very sensitive to noise, the noise will disturb the edge detection creating false edges.

(iii) The X-direction refers to the direction of the gradient in the image, therefore the x-direction kernel recovers vertical edges.

(iv) Any of Prewitt, Sobel, Laplacian, Gaussian Derivative.

(v) The laplacian is a 2^{nd} order Kernel.

(vi) The σ parameter controls the filtering of the gaussian, it determines the scale of the edges found in the image.

(vii) After filtering with the Laplacian kernel, a black and white line is produced either side of the true edge peak.

Answer Q17:

G_X

3	5
5	7

G_Y

-3	-5
-5	-7

G

4.24	7.07
7.07	9.9

θ

-45	-45
-45	-45

Answer Q18:

- 1. Apply Smoothing
- 2. Find Gradient Magnitude & direction (using Sobel filters).
- 3. Apply Non-maximum suppression.
- 4. Apply Hysterersis Thresholding.

Answer Q19:

As Pixel C is above the Min Val and is connected to pixels above Max Val it will be retained as part of an edge.

Answer Q20:

(i) All the possible lines through each edge point are considered, if the same line goes through two points then the same value in parameter space (m,b) will occur twice. By counting which parameter values occur most frequently the lines corresponding to edges in the image are identified.

(ii) The problem is that as the slope of the lines approach vertical the slope approaches infinity, this makes the algorithm impractical to use due to the need to store a large number of slope values.

Answer Q21:

"The Harris corner detector operates by finding intensity changes moving in a small neighbourhood of the pixel" is the correct statement as it summarises the principle of operation of the Harris corner algorithm. The second statement is incorrect as the algorithm uses gradient information in both the X and Y directions. The final statement is also incorrect as the harris algorithm does not use pixel frequency occurrence at any point in the algorithm.

Answer Q22:

As we have been given the 2nd moment matrix $M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} = \begin{bmatrix} 10 & 1 \\ 1 & 10 \end{bmatrix}$, we can calculate the corner response function R using the equations (see Lesson 1.4, slide 11) that relate R to the terms of the 2nd moment matrix.

$$R = Det(M) - \alpha\,(Trace(M))^2$$

Where the Determinant $Det(M)$ and Trace $Trace(M)$ of the second moment matrix are given by:

$$Det(M) = AB - C^2 = 10 \times 10 - 1^2 = 99$$
$$Trace(M) = A + B = 10 + 10 = 20$$

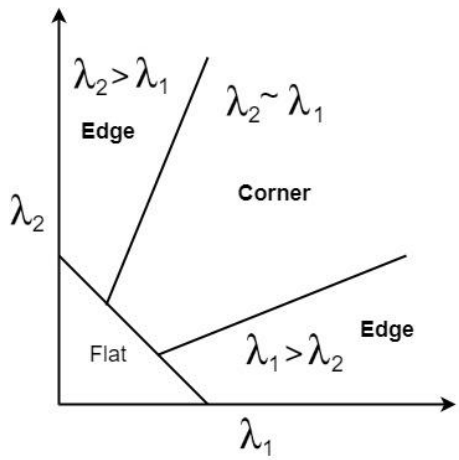
We therefore calculate R as

$$R = 99 - 0.04 \times 20^2 = 83$$

This is greater than zero, therefore there is a corner present.

Answer Q23:

We can also estimate if there is a corner present from the “curvature” or eigenvalues of the second moment matrix. From the image shown in the slides, we can see that we will have an edge when one of the eigenvalues or curvatures is high and the other small, a corner is detected with two larger eigenvalues or curvatures and small values or low curvature is a flat region.



Looking at statement from the question we can now that:

(i) In the Harris corner algorithm an Edge will have one high and one low curvature. **TRUE**

(ii) In the Harris corner algorithm all edges will have all high curvatures. **FALSE** (Only corners has all high curvatures)

(iii) In the Harris corner algorithm all edges will have all minimal curvatures. **FALSE** (only flat regions have minimal curvature)

(iv) In the Harris corner algorithm all corners will have all high curvatures. **TRUE**

Answer Q24:

The image histogram shows no clear threshold between regions with different pixel values or intensity. Therefore we cannot use the Threshold algorithm.

Answer Q25:

(a) The local minima
(b) Dams are built which form edges between regions
(c) Over segmentation
(d) In order to find object centres, which can be used to create markers.

Answer Q26:

K means algorithm.