Midsem - Winter 2025 CSE/ECE 344/544: Computer Vision Mar. 02, 2025

Time: 60 minutes Max marks: 23

**Instructions:**
- Do not plagiarize. Do not assist your classmates in plagiarism.
- Show your full solution for the questions to get full credit.
- Attempt all questions that you can.
- In the unlikely case a question is not clear, discuss it with an invigilating TA. Please ensure that you clearly include any assumptions you make, even after clarification from the invigilator.

1. **Deep Learning CNN & loss functions [CO-1]**

(a) (5 points) i. (1 point) Let the weights of the $3 \times 3$ kernel be $w_{i,j},\ i, j \in \{0, 1, 2\}$. Write the convolution operation as a matrix multiplication when the input is the $4 \times 4$ single-channel image and the output is the $2 \times 2$ feature map. You may ignore the bias term for this part, however, for full credit, describe how would the input / output need to be processed in order to implement the convolution operation as a matrix multiplication, e.g., processing may need operations like reshape etc.

ii. (2 points) Say that we have a $5 \times 5$ single-channel image as input and a $3 \times 3$ kernel. Write the expression similar to the above matrix multiplication if you are to perform a *dilated* (or atrous) convolution, with a dilation rate of $d = 2$, i.e., inserting one $(d-1)$ zeros between kernel elements. Ignore the bias element. What would be the size of the feature map assuming no padding and a stride of 1?

iii. (2 points) Consider a traffic surveillance video. Say your multi-object tracking model returns a time-series of 2D image coordinates $\{(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t), \ldots, (x_T, y_T)\}$. Your goal is to classify these trajectories into two classes (U-turn and Right-turn) and you are asked to design a 1-D convolutional neural network for the task. Write the expression of 1-D convolution operation using a 3 length kernel. Draw a 1 conv. layer CNN-based architecture with a dense layer (of a size of your choice) with a final binary classification output layer. Describe the activation functions used and state the number of learnable parameters in your model.

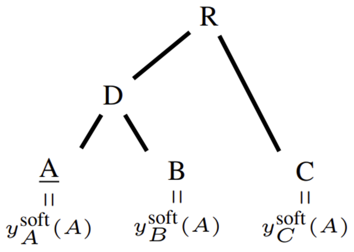

Figure 1: Class Label Hierarchy

(b) (5 points) For the class label hierarchy shown in Fig. 1, let $\mathbf{y}_A^{soft}$ be a *soft* label for a sample of class $A$, which is different from *hard* labels (one-hot encoded labels). In Fig. 1, $\mathbf{y}_A^{soft} = [y_A^{soft}(A), y_A^{soft}(B), y_A^{soft}(C)]^\top$. Here $y_A^{soft}(B)$ is the *target* probability of a sample predicted to be of class $B$, when its ground-truth class label is given to be $A$. Note that for hard-labels, this probability is always zero, when $A \neq B$.

i. (3 points) Based on the label hierarchy in Fig. 1, the soft label is based on the 'lowest common ancestor' (LCA). Say the distance $d_{LCA}(A, B) = d_{LCA}(B, A) = 1$ and $d_{LCA}(A, C) = d_{LCA}(C, A) = d_{LCA}(C, B) = d_{LCA}(B, C) = 2$. Use the *softmax* function $\sigma(-d_{LCA}(A, B))$ to define the target probability $y_A^{soft}(B)$. Write the complete soft label $\mathbf{y}_A^{soft}$.

ii. (2 points) Write the expression of the cross-entropy loss function when you choose to use the soft label as the target probability. Once you have defined the soft labels for each of the three classes, please use the symbols to write the expression of the cross-entropy loss. *Note*: Writing the expression in numerical values will be penalized.

**Solution:**

(a) (5 points) i. (1 point) Refer to solutions for quiz 2. +0.5 for mentioning processing and +0.5 for mentioning the correct multiplication of matrices

ii. (2 points) For a $3 \times 3$ kernel with $d = 2$, we have:

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}$$

Original kernel

$$\begin{bmatrix} w_{11} & 0 & w_{12} & 0 & w_{13} \\ 0 & 0 & 0 & 0 & 0 \\ w_{21} & 0 & w_{22} & 0 & w_{23} \\ 0 & 0 & 0 & 0 & 0 \\ w_{31} & 0 & w_{32} & 0 & w_{33} \end{bmatrix}$$

Dilated kernel

Also, the I/p ‘is 5x5 single channel with no padding and stride=1 , so our output feature map will be $1 \times 1$. The matrix expression for convolution will be:

$$W = \begin{bmatrix} w_{11} & 0 & w_{12} & 0 & w_{13} & 0 & 0 & 0 & 0 & 0 & w_{21} & 0 & w_{22} & 0 & w_{23} & 0 \\ 0 & 0 & 0 & 0 & w_{31} & 0 & w_{32} & 0 & w_{33} \end{bmatrix}$$

$W$ is a $1 \times 25$ vector.

The input vector is:

$$\beta = \begin{bmatrix} I_{11} & I_{12} & I_{13} & I_{14} & I_{15} & I_{21} & I_{22} & I_{23} & I_{24} & I_{25} & I_{31} & I_{32} & I_{33} & I_{34} & I_{35} & I_{41} \\ I_{42} & I_{43} & I_{44} & I_{45} & I_{51} & I_{52} & I_{53} & I_{54} & I_{55} \end{bmatrix}^{\mathsf{T}}$$

$\beta$ is a $25 \times 1$ vector.

Thus, the output feature map is:

$$O = W\beta$$

$O$ is a $1 \times 1$ scalar.

+1 mark for the correct matrix and +1 for the correct size of the output feature map

iii. (2 points) Assuming no padding and a stride of 1, the 1D convolution layer can be represented as:

**Input Layer:** 2 channels

**Convolutional Layer ($3 \times 3$ kernel) → Feature Map → Fully Connected Layer (100 neurons)→ Activation layer**

For the convolution layer, we use ReLU activation. For the dense layer, we also use ReLU, and the final layer has a sigmoid activation.

**Learnable Parameters:**

Convolution Layer:

$$2 \times 3 + 1 = 7$$

where:

- 2 is the input channel size
- $3 \times 3$ is the kernel size
- +1 is the bias term

Dense Layer:

$$((T-2)+1) \times 100$$

where:

- $(T-2)+1$ is the feature map size
- 100 is the number of neurons in the dense layer

Output Layer:

$$100+1=101$$

where:

- 100 represents the number of neurons
- +1 is the bias term

Total Learnable Parameters:

$$7+100(T-1)+101=100T+8$$

Please note that this number may change based on:

- Number of nodes in the dense layer
- Stride and padding in the convolution layer
- Use of activations that have learnable parameters

+0.5 for the expression for 1D convolution, +0.5 for a valid architecture, +1 for correct parameters

(b) (5 points) i. (3 points) Softmax function $\sigma(z_j) = \frac{e^{z_j}}{\sum e^{z_j}}$

$$y_a^{soft} = [\ \sigma(-d_{ca}(A,A)),\ \sigma(-d_{ca}(A,B)),\ \sigma(-d_{ca}(A,C))\ ]$$

$$y_a^{soft} = [\ \frac{e^{-0}}{e^{-0}+e^{-1}+e^{-2}},\ \frac{e^{-1}}{e^{-0}+e^{-1}+e^{-2}},\ \frac{e^{-2}}{e^{-0}+e^{-1}+e^{-2}}\ ]$$

$$y_a^{soft} = [\ 0.665,\ 0.245,\ 0.090\ ]$$

**Rubric:** $y_a^{soft}$ is a 3-dimensional vector.
**+1 mark** for each correct entry in the vector (leaving your solution in powers of $e$ is okay).
**+0.5 marks** per entry if answer left in terms of $\sigma$.

ii. (2 points)

$$y_b^{soft} = [\ \frac{e^{-1}}{e^{-0}+e^{-1}+e^{-2}},\ \frac{1}{e^{-0}+e^{-1}+e^{-2}},\ \frac{2}{e^{-0}+e^{-1}+e^{-2}}\ ]$$

$$y_c^{soft} = [\ \frac{e^{-2}}{1+2e^{-2}},\ \frac{e^{-2}}{1+2e^{-2}},\ \frac{1}{1+2e^{-2}}\ ]$$

For $N$ samples, the cross-entropy loss can be written as:

$$
\begin{aligned}
L_{CE} = -\sum_{i=1}^{N} \Bigg\{ & \Big[ y_A^{soft}(A) \log P(\hat{y}_i = A|x_i) \\
& + y_A^{soft}(B) \log P(\hat{y}_i = B|x_i) \\
& + y_A^{soft}(C) \log P(\hat{y}_i = C|x_i) \Big] \cdot \mathbf{1}[y_i = A] \\
& \\
& + \Big[ y_B^{soft}(A) \log P(\hat{y}_i = A|x_i) \\
& + y_B^{soft}(B) \log P(\hat{y}_i = B|x_i) \\
& + y_B^{soft}(C) \log P(\hat{y}_i = C|x_i) \Big] \cdot \mathbf{1}[y_i = B] \\
& \\
& + \Big[ y_C^{soft}(A) \log P(\hat{y}_i = A|x_i) \\
& + y_C^{soft}(B) \log P(\hat{y}_i = B|x_i) \\
& + y_C^{soft}(C) \log P(\hat{y}_i = C|x_i) \Big] \cdot \mathbf{1}[y_i = C] \Bigg\}
\end{aligned}
$$

where $\mathbf{1}[\mathbf{y_i} = \mathbf{A}]$ is an indicator function that is 1 when the $i$th sample belongs to class $A$, and $P(\hat{y}_i = A|x_i)$ is the predicted probability that the $i$th sample $x_i$ belongs to class $A$.

**Rubric: 0.5 marks** for calculating $y_b^{soft}$, **0.5 marks** for calculating $y_c^{soft}$, **1 mark** for calculating $L_{ce}$ (no step marking, but leaving your solution in powers of $e$ is okay).

2. **Coordinate Transformation and Image Formation [CO-2]**

(a) (5 points) A UAV has the forward direction as the $X^U$-axis, the right direction as the $Y^U$-axis and the up direction as the $Z^U$-axis. The superscript denotes the local coordinate system, i.e., the UAV's frame in this case. It is equipped with a LIDAR that has its left direction as the $X^L$-axis, its forward direction as the $Y^L$-axis and the bottom direction as the $Z^L$-axis. Assume that the $X - Y$ planes in the UAV's frame and the LIDAR's frame are parallel.

i. (1 point) Is there a Euclidean transformation that would transform a point from the UAV's frame to the LIDAR's frame of reference? Justify or refute.

ii. (3 points) Either way, can you write the transformation from the LIDAR to the UAV frame, assuming that the origins in the two frames coincide? If yes, please do.

iii. (1 point) In such a case, what are the number of degrees of freedom in the transformation?

(b) (5 points) (1+1+3=5) Let $\mathbf{H}_{3\times 3}$ be a planar homography that yields the mapping $\mathbf{p}' = \mathbf{H}\mathbf{p}$, where $\mathbf{p}, \mathbf{p}'$ are 2D points represented in homogeneous coordinates.

i. (1 point) How many degrees of freedom does $\mathbf{H}$ have? Explain why.

ii. (1 point) How many point correspondences would you need to solve for $\mathbf{H}$?

iii. (3 points) To solve for $\mathbf{H}$, you will need to write a system of linear equations of the form $\mathbf{P}\mathbf{h} = \mathbf{0}$, where $\mathbf{h} = \text{vec}(\mathbf{H})$ is the 9-dimensional vectorized (reshaped) version of the matrix $\mathbf{H}$. Derive this linear system of equations, starting from $\mathbf{p}' = \mathbf{H}\mathbf{p}$. You may assume $\mathbf{p} = [x, y, w]^\top$ and $\mathbf{p}' = [x', y', w']^\top$ and $w, w' \neq 0$.

(c) (3 points) (a) (1 point) Write all the components of the image formation pipeline.

(b) (1 point) What can you say about the relationship between the image and the scene if all the points in the scene have the same depth? Only answers in terms of geometric transformations will be credited.

(c) (1 point) What can you say about the relationship between the image and the scene if the focal length of the camera is infinite, but the scene points are not necessarily at the same depth? Only answers in terms of geometric transformations will be credited.

**Solution:**

(a) (5 points) i. (1 point) We have defined a Euclidean transformation as 3D Rotation and translation. In the given setup, UAV coordinate frame is left-handed while LIDAR coordinate frame is right-handed. We cannot combine a 3D Rotation and Translation to achieve a translation between these frame. (Other explanations include: Showing determinant of the transformation matrix, provided the matrix is correct)

If someone has mentioned reflection and still called it Euclidean fetches 1 mark.

Rubric:
Case 1:
0.5 for Euclidean transformation does not exist + 0.5 for correct explanation

Case 1:
0.5 for Euclidean transformation exists (ONLY if reflection, multiplying z axis by -1 is mentioned) + 0.5 for correct explanation

ii. (3 points) It is given that the origins coincide hence the translation is 0 and we need a pure reflection to generate the mapping.

Representing a point in LIDAR frame as $(X_L, Y_L, Z_L)$ and a point in UAV frame as $(X_U, Y_U, Z_U)$. We can create the transformation matrix by inspection.
$X_L \rightarrow -Y_U$
$Y_L \rightarrow X_U$
$Z_L \rightarrow -Z_U$
The transformation matrix to convert a point from LIDAR to the UAV frame,

$$^{U}T_L = {}^{U}R_L = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \text{ with } |^{U}R_L| = -1$$

We see that the transformation is an orthogonal matrix but has a determinant of -1. This property makes it a reflection necessary for switching between the right and left handed coordinate frame.

Rubric:
3 for correct final transformation matrix for LIDAR to UAV
max 1.5 if correct transformation matrix shown for UAV to LIDAR
0 if transformation matrix is wrong
-1 for not mentioning the axis which coordinate system do x,y and z belong to when writing the transformation

iii. (1 point) 3 D.O.F for 3D rotation as translation is given to be 0. To resolve the signs we still need 3 constraints coming from the 3 axes. Rubric:
1 for correct D.O.F

(b) (5 points) i. (1 point) 8 d.o.f as h is a 3x3 projective transformation matrix
**Rubric:**
+1 point for correct d.o.f

ii. (1 point) 4 correspondence points
**Rubric:**
+1 point for mentioning the correct number of correspondence points.

iii. (3 points)

$$p' = \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \end{bmatrix} p$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

One way to show the constraint is via cross multiplication.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} h_1^T p \\ h_2^T p \\ h_3^T p \end{bmatrix}$$

$$\begin{bmatrix} x'/z' \\ y'/z' \end{bmatrix} = \begin{bmatrix} h_1^T p/h_3^T p \\ h_2^T p/h_3^T p \end{bmatrix}$$

$$x' h_3^T p = z' h_1^T p \tag{1}$$

$$y' h_3^T p = z' h_2^T p \tag{2}$$

Another way to show equality of vectors via the cross product.

$$p' \times Hp = 0$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \times \begin{bmatrix} h_1^T p \\ h_2^T p \\ h_3^T p \end{bmatrix} = 0$$

$$\begin{bmatrix} 0 & -z' & y' \\ z' & 0 & -x' \\ -y' & x' & 0 \end{bmatrix} \begin{bmatrix} h_1^T p \\ h_2^T p \\ h_3^T p \end{bmatrix} = 0$$

Rewriting this as $Ah = 0$ where $h = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}_{9\times 1}$ will give a $3 \times 9$ matrix A, for which only 2 of the 3 rows are linearly independent. This can be shown by Gaussian elimination or any of your favorite methods for reducing the matrix to its row-echelon form.

$$\mathbf{P} \overset{\text{def}}{=} \begin{bmatrix} \mathbf{P}_1^T & \mathbf{0}^T & -u_1 \mathbf{P}_1^T \\ \mathbf{0}^T & \mathbf{P}_1^T & -v_1 \mathbf{P}_1^T \\ \vdots & \vdots & \vdots \\ \mathbf{P}_n^T & \mathbf{0}^T & -u_n \mathbf{P}_n^T \\ \mathbf{0}^T & \mathbf{P}_n^T & -u_n \mathbf{P}_n^T \end{bmatrix}$$

**Rubric:**
+1 for the correct way of writing constraints via cross-multiplication or vector cross-product
+2 for the final matrix expression (or an equivalent formulation)

(c) (3 points) i. (1 point) The image formation pipeline is as follows

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f_x & s & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_x \\ R_{21} & R_{22} & R_{23} & t_y \\ R_{31} & R_{32} & R_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3}$$

Simply, pixel points in 2D (p)$_{3\times1}$)= $K_{3\times3} \times P_{3\times4} \times [R|t]_{4\times4}\times$ Pixels in 3D (p$_{4\times1}$)

ii. (1 point) The result will be equivalent to a scaled version of the original scene.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f_x/Z & 0 & p_x/Z \\ 0 & f_y/Z & p_y/Z \\ 0 & 0 & 1/Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = 1/Z \begin{bmatrix} f_x + p_x \\ f_y + p_y \\ 1 \end{bmatrix} \tag{5}$$

iii. (1 point) An infinite focal length would mean an orthogonal projection instead of a perspective projection in the image plane.

**Rubric:**

i. (1 point) +1 point only assigned for given matrix form of image formation pipeline. Points are not given for only text or incorrect answers.

ii. (1 point) +1 point assigned for the correct answer with proof, only 0.5 points for textual explanation.

iii. (1 point) +1 point for correct answer, 0 otherwise.

3. **Evaluation Metrics. [CO-4]**

(a) (3 points) (*Extra Credit*)

i. (1 point) How is mean Average Recall (mAR) computed for evaluating object detection?

ii. (2 points) What are the different kinds of errors in object detection, as one may discover using a tool like TIDE.

**Solution:**

(a) (3 points) i. (1 point) 1. Order predictions in decreasing order of confidence score.

2. Associate each prediction with a ground truth based on an IoU threshold from the range.

3. Compute TP, FN and calculate cumulative recall as $\frac{TP}{TP+FN}$.

4. Average recall over all IoU thresholds.

**Rubric:**

+1 if the algorithm is vaguely correct.

+0.5 for partially correct answers.

ii. (2 points) 1. Classification Error.

2. Localization Error.

3. Classification + Localization Error.

4. Duplicate Detection Error.

5. Background Error.

6. Missed Detections Error.

Any 4 of the above 6 errors suffice for full credit. For each error, just the name is sufficient. Otherwise, the corresponding definition or a description thereof should be mentioned.

**Rubric:**

+0.5 for each correct error