National University of Computer and Emerging Sciences
Islamabad Campus

Computer Vision (AI4002)

Course Instructor(s):
Ms. Khadija Mahmood
Section(s): (if applicable)
Part:B

Final Examination

Total Time (Hrs): 2
Total Marks: 100
Total Questions: 3

Date: Dec 18, 2024

____________	____________	________________________
Roll No	**Course Section**	**Student Signature**

Do not write below this line.

Attempt all the questions.

[CLO 2: Apply algorithmic solutions related to the degree program to recent related problems]

Q2: Short Questions: **[30 marks]**

Question 2.1: Given two point clouds: **[5 marks]**

Point Cloud A: {(1,2),(3,4),(5,6)}

Point Cloud B: {(2,3),(4,5)}

Calculate the Chamfer Loss

Question 2.1

Given two point clouds:

Point cloud A: $\{(1,2), (3,4), (5,6)\}$
Point cloud B: $\{(2,3), (4,5)\}$

$$\text{Chamfer Loss} = \sum_{a \in A} \min_{b \in B} \|a-b\|^2 + \sum_{b \in B} \min_{a \in A} \|b-a\|^2$$

Step 1: Compute $\sum_{a \in A} \min_{b \in B} \|a-b\|^2$

For $a=(1,2)$
$(1-2)^2 + (2-3)^2 = 1+1 = 2$ $\quad \min(2,18) = \boxed{2}$
$(1-4)^2 + (2-5)^2 = 9+9 = 18$

For $(3,4)$
$(3-2)^2 + (4-3)^2 = 2$ $\quad \min(2,2) = \boxed{2}$
$(3-4)^2 + (4-5)^2 = 2$

For $(5,6)$ $\quad \min(8,2) = \boxed{2}$
"

Sum for A

$$\sum_{a \in A} \min_{b \in B} \|a-b\|^2 = 2+2+2 = 6$$

For $b=(2,3)$
$(2-1)^2 + (3-2)^2 = 1+1 = 2$ $\quad \min(2,2,18) = \boxed{2}$
$(2-3)^2 + (3-4)^2 = 1+1 = 2$
$(2-5)^2 + (3-6)^2 = 9+9 = 18$

For $b=(4,5)$ — $\quad \min(18,2,2) = \boxed{2}$
"

Sum for B = 4

$\boxed{\text{Chamfer loss} = 6+4 = 10.}$

Question 2.2: Write the three major disadvantages of Vision Transformer (ViT). [3 marks]

- Data-Hungry Nature

- Computational Complexity
- Lack of Inductive Bias

Question 2.3: Point Cloud P: **[4+3 marks]**

P= {(1,2,3),(4,5,6),(7,8,9)} (3 points in 3D space),

(a) For getting point features f1, f2 and f3, take a dot product of all the points with 3 by 5 matrix mentioned below.

4	2	1	6	0
5	4	4	2	3
9	6	0	5	1

What is the dimension of feature matrix?

3 by 5

(b) Write the pooled vector after applying **max pooling** across the point features.

41	28	9	25	9
95	64	24	64	21
149	100	39	119	33

Max pool vector: 149,100,39,119,33

Question 2-3.

Point Cloud P

$P = \{(1,2,3), (4,5,6), (7,8,9)\}$

$$\begin{bmatrix} 4 & 2 & 1 & 6 & 0 \\ 5 & 4 & 4 & 2 & 3 \\ 9 & 6 & 0 & 5 & 1 \end{bmatrix}$$

$(1)(4) + (2)(5) + (3)(9) = 41$
$(1)(2) + (2)(4) + (3)(6) = 28$
$(1)(1) + (2)(4) + (3)(0) = 9$
$(1)(6) + (2)(2) + (3)(5) = 25$
$(1)(0) + (2)(3) + (3)(1) = 9$

$4\times4 + 5\times5 + 6\times9 = 95$
$4\times2 + 5\times4 + 6\times6 = 64$
$4\times1 + 5\times4 + 6\times0 = 24$
$4\times6 + 5\times2 + 6\times5 = 64$
$4\times0 + 5\times3 + 6\times1 = 21$

$7\times4 \quad 8\times5 \quad 9\times9 = 149$
$7\times2 \quad 8\times4 \quad 9\times6 = 100$
$7\times1 \quad 8\times4 \quad 9\times0 = 39$
$7\times6 \quad 8\times2 \quad 9\times5 = 119$
$7\times0 \quad 8\times3 \quad 9\times1 = 33$

f1	41	28	9	25	9
f2	95	64	24	64	21
f3	149	100	39	119	33

Max pool Vector

Mp-f = 149, 100, 39, 119, 33

Question 2.4: **Transformer for segmentation** **[4+3 marks]**

(a) What changes are must requiring in traditional transformer with the backbone of CNN based architecture for semantic segmentation task. Write at least 4 in bullet points.

- Input Image: Raw image data.
- CNN Feature Extractor: Extracts low-level spatial features.
- Patch Embedding: Divides the CNN features into patches and projects them into a higher-dimensional feature space.
- Transformer Encoder: Processes the patches using multi-head self-attention and captures global dependencies.
- Feature Fusion (Skip Connections): Combines transformer outputs with low-level CNN features.
- Upsampling Block: Upsamples the fused features to the original input resolution.
- Output Segmentation Map: Pixel-wise classification.

(b) Make a block diagram to show your working

Based on above information

Question 2.5: **Overfitting in YOLO** **[3 marks]**

While fine-tuning YOLO on a small dataset, you notice that the model is overfitting. What techniques can you use to reduce overfitting during training? Write at least 3.

- **Apply Data Augmentation**: Use techniques like flipping, rotation, scaling, or adding noise to increase data variety.
- **Use Regularization**: Apply weight decay (L2 regularization) and dropout to prevent overfitting.
- **Early Stopping**: Stop training when validation loss stops improving to avoid overfitting to noise.
- **Reduce Model Complexity**: Use a smaller YOLO model variant (e.g., YOLOv5s).
- **Freeze Layers**: Fine-tune only the later layers by freezing the pre-trained layers.

Question 2.6: Write about your semester project in precise manner. **[5 marks]**

Q3: Long Question:

This question explores the core steps of the Swin Transformer, focusing on patch partitioning, linear embedding, and the attention mechanism. You will compute the attention scores and outputs for a set of image patches, gaining insights into the model's process of capturing spatial information and relationships between image regions. **[30 marks]**

You are given an 8×8 grayscale image matrix:

10	20	30	40	50	60	70	80
15	25	35	45	55	65	75	85
20	30	40	50	60	70	80	90
25	35	45	55	65	75	85	95

30	40	50	60	70	80	90	100
35	45	55	65	75	85	95	105
40	50	60	70	80	90	100	110
45	55	65	75	85	95	105	115

Answer the following questions step by step to understand the Swin Transformer pipeline.

Question 3.1: Divide the 8×8 image into non-overlapping 2×2 patches.

(a) How many patches are created? 16
(b) What are the dimensions of each patch? 2 *2*1

Question 3.2: Linear Embedding

(a) Flatten the 2×2 patch to make 1×4 vector. Perform this operation only for first four patches (row wise). Fill the values in the below mentioned tables.

V1	10	20	15	25
V2	30	40	35	45
V3	50	60	55	65
V4	70	80	75	85

(b) Use the following linear embedding matrix to project this flattened 1×4 vector into a 1×3 vector:

P=

0.1	0.5	0.9
0.2	0.6	0.1
0.3	0.7	0.2
0.4	0.8	0.3

Write the projected linear vectors for first four patches in the table.

L1	19.5	47.5	21.5
L2	39.5	99.5	51.5
L3	59.5	151.5	81.5
L4	79.5	203.5	111.5

Question 7.2 b

$$\begin{bmatrix} 10 & 20 & 15 & 25 \\ 30 & 40 & 35 & 45 \\ 50 & 60 & 55 & 65 \\ 70 & 80 & 75 & 85 \end{bmatrix} \quad \begin{bmatrix} 0.1 & 0.5 & 0.9 \\ 0.2 & 0.6 & 0.1 \\ 0.3 & 0.7 & 0.2 \\ 0.4 & 0.8 & 0.3 \end{bmatrix}$$

$(10)(0.1) + (20)(0.2) + (15)(0.3) + (25)(0.4)$
$= \boxed{19.5}$

$\cancel{(30)(0.5) + (40)(0.6) + (35)(0.7) + (45)}$

$(10)(0.5) + (20)(0.6) + (15)(0.7) + (25)(0.8)$
$= \boxed{47.5}$

$(10)(0.9) + (20)(0.1) + (15)(0.2) + (25)(0.3)$
$= \boxed{21.5}$

$(30)(0.1) + (40)(0.2) + (35)(0.3) + (45)(0.4)$
$= \boxed{39.5}$

$(30)(0.5) + (40)(0.6) + (35)(0.7) + (45)(0.8)$
$= \boxed{99.5}$

$(30)(0.9) + (40)(0.1) + (35)(0.2) + (45)(0.3)$
$= \boxed{51.5}$

$(50)(0.1) + (60)(0.2) + (55)(0.3) + (65)(0.4)$
$= \boxed{59.5}$

$(50)(0.5) + (60)(0.6) + (55)(0.7) + (65)(0.8)$
$= \boxed{151.5}$

$(50)(0.9) + (60)(0.1) + (55)(0.2) + (65)(0.3)$
$= \boxed{81.5}$

Question 3.3: For Stage 1, apply window-based attention with a window size of 2×2.

(a) How many windows are created? 4

(b) Make a table for first window to illustrate window in an image.

10	20	30	40
15	25	35	45
20	30	40	50
25	35	45	55

(c) What is the size (including depth 'C') of each window? 3

(d) Assume that we have completed the patch embedding step for each patch, and we are now at the Multi-Head Self-Attention (MSA) stage. Here, the input embeddings for a 2×2 window of patches are denoted as E1, E2, E3, E4. For further computations, For the simplicity, consider the following flattened 1 by 3 embedding for whole window in this case:

E1: [1,2,3], E2= [4,5,6], E3=[7,8,9], E4= [10,11,12].

Compute the **attention score** for the embeddings using the scaled dot-product attention formula:

$$\text{Attention score}=\text{Softmax}\left(Q\cdot K^{T}.\frac{1}{\sqrt{C}}\right).V$$

0.1	0.2	0.3
0.4	0.5	0.6
0.7	0.8	0.9

WQ

0.3	0.1	0.2
0.5	0.4	0.3
0.7	0.6	0.5

WK

0.2	0.1	0.3
0.4	0.3	0.5
0.6	0.5	0.7

WV

13.6	**10.3**	**16.9**
13.6	**10.3**	**16.9**
13.6	**10.3**	**16.9**
13.6	**10.3**	**16.9**

$(70)(0.1) + (80)(0.2) + (75)(0.3) + (85)(0.4)$

$= \boxed{79.5}$

$(70)(0.5) + (80)(0.6) + (75)(0.7) + (85)(0.8)$

$= \boxed{203.5}$

$(70)(0.9) + (80)(0.1) + (75)(0.2) + (85)(0.3)$

$= \boxed{111.5}$

Question 3.3 (d)

$E_1 = [1,2,3]$ . . $E_2 = [4,5,6]$ $E_3 : [7,8,9]$ $E_4 = [10,11,12]$

$Q_1 = E_1 . WQ$, $V_1 = E_1 . WV$, $K_1 = E_1 . WK$

$$Q_1 = [1,2,3]\begin{bmatrix} 0.1 & 0.2 & 0.3 \\ 0.4 & 0.5 & 0.6 \\ 0.7 & 0.8 & 0.9 \end{bmatrix} = [3, 3.6, 4.2]$$

$(1)(0.1) + (2)(0.4) + (3)(0.7) = 3$

$(1)(0.2) + (2)(0.5) + (3)(0.8) = 3.6$

$(1)(0.3) + (2)(0.6) + (3)(0.9) = 4.2$

$Q_2 = E_2 . WQ = [6.6, 8.1, 9.6]$

$Q_3 = E_3 . WQ = [10.2, 12.6, 15]$

$Q_4 = E_4 . WQ = [13.8, 17.1, 20.4]$

$(4\times3)(3\times3)$

4×3

Key Matrix

$$K_1 = [1, 2, 3] \begin{bmatrix} 0.3 & 0.1 & 0.2 \\ 0.5 & 0.4 & 0.3 \\ 0.7 & 0.6 & 0.5 \end{bmatrix} = [3.4, 2.7, 2.3]$$

Similarly we'll get Query, Key and Value Matrices

3	3.6	4.2
6.6	8.1	9.6
10.2	12.6	15
13.8	17.1	20.4

Query

3.4	2.7	2.3
7.9	6	5.3
12.4	9.3	8.3
16.9	12.6	11.3

Key

2.8	2.2	3.4
6.4	4.9	7.9
10	7.6	12.4
13.6	10.3	16.9

Value.

Now we'll calculate attention Score.

first we'll calculate $Q \cdot K^T$.

$$\underset{Q}{\begin{bmatrix} 3 & 3.6 & 4.2 \\ 6.6 & 8.1 & 9.6 \\ 10.2 & 12.6 & 15 \\ 13.8 & 17.1 & 20.4 \end{bmatrix}} * \underset{K^T}{\begin{bmatrix} 3.4 & 7.9 & 12.4 & 16.9 \\ 2.7 & 6 & 9.3 & 12.6 \\ 2.3 & 5.3 & 8.3 & 11.3 \end{bmatrix}} = \underset{QK^T}{\begin{bmatrix} 29.5 & 67.5 & 105.5 & 143.5 \\ 66.3 & 151.6 & 236.8 & 322.0 \\ 103.4 & 235.6 & 368.1 & 500.6 \\ 140.0 & 319.7 & 499.4 & 679.2 \end{bmatrix}}$$

$$\frac{Q \cdot K^T}{\sqrt{C}} = \begin{bmatrix} 16.8 & 38.5 & 60.1 & 81.8 \\ 37.8 & 86.4 & 135.0 & 183.5 \\ 58.8 & 134.3 & 209.8 & 285.3 \\ 79.8 & 182.2 & 284.6 & 387.1 \end{bmatrix}, V = \begin{bmatrix} 2.8 & 2.2 & 3.4 \\ 6.4 & 4.9 & 7.9 \\ 10 & 7.6 & 12.4 \\ 13.6 & 10.3 & 16.9 \end{bmatrix}$$

$= 3.$

$\frac{Q.K^T}{\sqrt{c}}$ we need to apply Softmax on it.

$$\text{Softmax}\left(\frac{Q.K^T}{\sqrt{c}}\right) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now we calculate

$$\text{Softmax}\left(\frac{Q.K^T}{\sqrt{c}}\right).V = \begin{bmatrix} 13.6 & 10.3 & 16.9 \\ 13.6 & 10.3 & 16.9 \\ 13.6 & 10.3 & 16.9 \\ 13.6 & 10.3 & 16.9 \end{bmatrix}$$

Question 3.4: After Stage 1 (window-based attention), perform patch merging.

(a) How many patches remain after merging? 4
(b) What are the new spatial dimensions and feature dimensions? 4* 4*6

Question 3.5: Shift patches by 1 position to the right and 1 position down. Write the new arrangement of patch indices. Fill the table with shifted values.

115	45	55	65	75	85	95	105
80	10	20	30	40	50	60	70
85	15	25	35	45	55	65	75
90	20	30	40	50	60	70	80
95	25	35	45	55	65	75	85
100	30	40	50	60	70	80	90
105	35	45	55	65	75	85	95
110	40	50	60	70	80	90	100

Question 3.6: **Track dimensions through the pipeline, Fill the table.**

Stage	Spatial Dimension	Feature Dimension
Input	8 by 8	1
Stage 1	4 by 4	3
Stage 2	2 by 2	6
Stage 3	1 by 1	12