Time: 120 minutes Max marks: 20

**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.
- True/False questions DO NOT require justification, but carry a **negative 1/2 mark** for each wrong answer. No penalty for leaving the question unanswered.
- 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. **True / False questions [CO-1, CO-2].**
   (a) (½ point) It is possible to recover a camera's intrinsic matrix solely from multiple views of a planar calibration target, even without knowing the target's absolute pose in the world frame.
   (b) (½ point) The extrinsic matrix of a camera is always invertible and represents a rigid transformation between the image and pixel coordinate systems.
   (c) (½ point) The Essential matrix has exactly 7 degrees of freedom and is defined only up to scale for any pair of images.
   (d) (½ point) Given a correct fundamental matrix, the epipolar line in one image corresponding to some point in the other image is guaranteed to pass through the image center.
   (e) (½ point) A $3 \times 3$ homography matrix estimated between two general views of an arbitrary scene can perfectly map all points in one view to their corresponding points in the other view.
   (f) (½ point) In self-attention mechanisms, the learned attention weights are invariant to spatial position when applied to image patches.
   (g) (½ point) Dilated (or atrous) convolutions allow CNNs to aggregate multi-scale context without increasing the number of parameters or reducing spatial resolution.
   (h) (½ point) Self-attention mechanisms are inherently more efficient than convolution for processing high-resolution images.
   (i) (½ point) In a well-trained GAN, the discriminator eventually becomes useless because it perfectly distinguishes real from generated samples.
   (j) (½ point) A vision-language model trained with contrastive loss, like CLIP (Contrastive Language-Image Pretraining) cannot be used for zero-shot classification, i.e., classifying images into new categories that were not defined at train time.

**Solution:**
(a) (½ point) **True**. Zhang's method that we used in the HWs estimates intrinsics from homographies without needing the absolute world pose.
(b) (½ point) **False**. Extrinsic parameters map from world to camera coordinates, not image/pixel coordinates; also it's not between image and pixel spaces.
(c) (½ point) **False**. The Essential matrix has only 5 degrees of freedom.
(d) (½ point) **False**. The epipolar line passes through the epipole, which in general may be outside of the image altogether.
(e) (½ point) **False**. A $3 \times 3$ homography is only valid for completely planar scenes.

(f) (½ point) **False**. Self-attention is permutation-equivariant by default, which is evident from the fact that the attention scores are computed between pairs of tokens. Positional embeddings are added to inject spatial information in the embeddings.

(g) (½ point) **True**. Dilated convolutions insert zeros in the kernel that lead to an increased receptive field without downsampling the feature maps or increasing the number of learnable parameters.

(h) (½ point) **False**. Self-attention has quadratic complexity w.r.t. input size, unlike convolutions which scale linearly.

(i) (½ point) **False**. In the perfect training scenario, the discriminator outputs 0.5 for all inputs, which means it CAN NOT distinguish between the real and the generated samples.

(j) (½ point) **False**. Models like CLIP use contrastive loss to enable zero-shot prediction by aligning image and text embeddings.

2. (4 points) **Image Processing and Keypoint Detection & Matching [CO-1]**
Provide short answers to the following questions (1-2 sentences).

(a) (1 point) Would you choose a Gaussian smoothing filter over an averaging filter to smooth an image? Why?

(b) (2 points) What are the key properties that you would look for in local keypoint based feature detectors?

(c) (1 point) What properties of the descriptors make SIFT features more robust to lighting and pose changes?

**Solution:**

(a) (1 point) Yes, because a Gaussian filter gives more weight to central pixels, preserving edges better while smoothing compared to a uniform averaging filter, with the latter resulting in artifacts due to aliasing effects.

**Rubric:**
0.5 for yes/no;
0.5 for reasoning.

(b) (2 points) Desirable properties of local features

- Locality - features are local, so robust to occlusion and clutter
- Quantity - should detect many keypoint features in an image with reasonable textures.
- Distinctiveness - should allow for computation of descriptors that can differentiate a large database of features.
- Repeatability & invariance - should be detected under a wide range of scenarios with changes in lighting and viewpoint conditions.
- Efficiency - should be easy to compute to achieve real-time performance where needed.

(c) (1 point) SIFT uses gradient orientation histograms over local patches, making it invariant to illumination, scale, and moderate pose changes.

**Rubric**: For part (b) Any 2 properties gets full; 1 gets 1; otherwise 0. Other parts: 0, 1/2 or 1 based on how complete the answers are.

3. (5 points) **Evaluation metrics. [CO-4]**
For evaluating computer vision models, two commonly used performance metrics are mean Average Precision (mAP) for object detection and mean Intersection over Union (mIoU) for semantic segmentation.

   (a) (2 points) For an object detection task, suppose you are given the model's predicted bounding boxes and corresponding confidence scores for three classes. The ground truth annotations for the same set of images are also available. Briefly explain how mAP@0.5 is computed. Then, describe how this differs from mAP@[0.5:0.95].

   (b) (2 points) For a semantic segmentation task, consider a model that outputs a pixel-wise label map for a multi-class segmentation problem (say, with 5 classes). Explain how IoU is calculated for a single class, and then how mIoU is derived across all classes. How would class imbalance affect this metric?

   (c) (1 point) Suppose an object detection model achieves a high mAP@0.5 but low mAP@[0.5:0.95]. What does this tell you about the model's predictions?

**Solution:**

(a) (2 points) mAP for object detection.

- Computing mAP@0.5:
  To compute mAP@0.5, you first compute the Average Precision (AP) for each class using a fixed Intersection over Union (IoU) threshold of 0.5. This means a predicted bounding box is considered a true positive if its IoU with a ground truth box is $\geq 0.5$ *and* the predicted class matches the ground truth class. AP is computed from the precision-recall curve, i.e., by averaging the precision obtained for different values of recall, by ranking predictions by their confidence scores. The mean of these AP scores across all classes gives the mAP@0.5.
- Computing mAP@[0.5:0.95]:
  This metric, used in COCO evaluation, averages the AP across multiple IoU thresholds from 0.5 to 0.95, typically in steps of 0.05 (i.e., 0.5, 0.55, ..., 0.95). It evaluates how well the model performs under *increasingly strict localization requirements*. The final mAP is the average of the AP scores computed at each threshold, averaged across all classes.
- mAP@0.5 only checks for rough localization, whereas mAP@[0.5:0.95] penalizes poor localization more heavily and is a more comprehensive and stricter metric.

  **Rubric:**
  1 point for explaining mAP;
  0.5 for mAP@[0.5 : 0.95];
  0.5 for difference.

(b) (2 points) IoU and mIoU in Semantic Segmentation.
IoU for a Single Class: For a given class $c$, Intersection over Union (IoU) is calculated as:

$$\text{IoU}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c}$$

where:

- $\text{TP}_c$ (True Positive): Number of pixels correctly predicted as class $c$.
- $\text{FP}_c$ (False Positive): Number of pixels incorrectly predicted as class $c$.
- $\text{FN}_c$ (False Negative): Number of pixels belonging to class $c$ but predicted as another class.

Mean IoU:
Mean IoU is computed by averaging the IoU values across all classes:

$$\text{mIoU} = \frac{1}{C}\sum_{c=1}^{C}\text{IoU}_c$$

where $C$ is the total number of classes.

Effect of Class Imbalance:
mIoU assigns equal weight to all classes, irrespective of their frequency. As a result, classes with fewer pixels (rare classes) can disproportionately affect the overall score. Conversely, strong performance on dominant classes may obscure poor performance on rare classes. This sensitivity to class imbalance can pose challenges when evaluating models on imbalanced datasets.

**Rubric:**
1 point for explaining IoU;
0.5 for mIoU;
0.5 for class imbalance.

(c) (1 point) High mAP@0.5 but Low mAP@[0.5:0.95] This implies that the model is good at roughly identifying the presence and class of objects (i.e., it achieves high recall and coarse localization), but its bounding boxes are not tightly aligned with the ground truth. As the IoU threshold increases (requiring more precise localization), the model fails to meet the stricter criteria, causing the mAP to drop. This suggests poor localization accuracy of the predicted boxes.

4. (6 points) **Coordinate transformation [CO-2]**.
Consider a point $(2, 5, 1)$, which is rotated by $\pi/2$ about the Y-axis, followed by a rotation about the X-axis by $-\pi/2$, and finally translated by $(-1, 3, 2)$.

(a) (1 point) What is the coordinate transformation matrix in this case?

(b) (1 point) Find the new coordinates of this point. Where does the origin of the initial frame of reference get mapped to?

(c) (2 points) What is the direction of the axis of the combined rotation in the original frame of reference and what is the angle of rotation about this axis?

(d) (2 points) Using Rodrigues' rotation formula, show that you achieve the same rotation matrix as you get by sequentially applying the two rotations.

**Solution:**

(a) (1 point)

$$\mathbf{R}_y = \begin{bmatrix} \cos\frac{\pi}{2} & 0 & \sin\frac{\pi}{2} \\ 0 & 1 & 0 \\ -\sin\frac{\pi}{2} & 0 & \cos\frac{\pi}{2} \end{bmatrix}$$

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 & \\ 0 & \cos\frac{-\pi}{2} & 0 & -\sin\frac{-\pi}{2} \\ 0 & \sin\frac{-\pi}{2} & 0 & \cos\frac{-\pi}{2} \end{bmatrix}$$

$$\mathbf{t} = \begin{bmatrix} -1 \\ 3 \\ 2 \end{bmatrix}$$

The transformation to be applied is:

$$\mathbf{T} = \begin{bmatrix} \mathbf{R}_x\mathbf{R}_y & \mathbf{t} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 3 \\ 0 & -1 & 0 & 2 \end{bmatrix}$$

(b) (1 point) The new coordinates are:

- Orign: $[-1, 3, 2]^\top$
- Point: $\mathbf{Tp} = [0, 1, -3]^\top$, where $\mathbf{p}$ is the original point with coordinates $[2, 5, 1]^\top$.

(c) (2 points) The final rotation matrix is:

$$\mathbf{R} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

The axis and angle of rotation can be retrieved by:

$$\theta = \cos^{-1}\left(\frac{trace(\mathbf{R}) - 1}{2}\right) = 120^\circ$$

$$\mathbf{n} = \frac{1}{2\sin\theta}\begin{bmatrix} R_{32} - R_{23} \\ R_{13} - R_{31} \\ R_{21} - R_{12} \end{bmatrix} = \begin{bmatrix} -0.5769 \\ 0.5769 \\ -0.5769 \end{bmatrix}$$

**Rubric:**
1 point for correct axis of rotation;
1 point for correct angle of rotation.

(d) (2 points) If we use the equation $\mathbf{n}$ and $\theta$ calculated above using the equation,

$$\mathbf{R} = \mathbf{I} + \sin\theta\mathbf{N} + (1 - \cos\theta)^2$$

where,

$$\mathbf{N} = \begin{bmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{bmatrix}$$

we will get the rotation matrix as:

$$\begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

5. (4 points) (*Extra Credit*) **Multi-view Geometry [CO-2]**.
Let F be the fundamental matrix with the singular value decomposition, $\mathbf{F} = \mathbf{UDV}^\top$, where

$$\mathbf{U} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{bmatrix} \mathbf{V} = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 0 & 0 & 1 \\ 1/\sqrt{2} & 1/\sqrt{2} & 0 \end{bmatrix} \quad (1)$$

Locate the epipoles of this stereo camera system, i.e., write them as $3 \times 1$ vectors. What can you say about the value of $d_3$ and the image planes of this stereo setup?

**Solution:**

- (1 point) $\mathbf{F}$ is always rank 2 , therefore, $d_3 = 0$
- (2 points) The last column of $\mathbf{V}$ is the null space and the last column of $\mathbf{U}$ is the left null space. Therefore, the epipoles are:

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \text{ and } \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} \tag{2}$$

- (1 point) These epipoles are at infinity, which implies that the image planes are coplanar.