RDFIA December 2023

Justify ALL your answers.

1. In computer vision, is it interesting to build image descriptors that are:
   (a) invariant to a horizontal flip of the image for classification on ImageNet dataset?
   (b) invariant to a 20° rotation of the image for classification on MNIST dataset? and for a 90° rotation?

2. In traditional Bag of Word representation, explain the role of the K-means algorithm.

3. Considering the following soft-margin SVM optimization:

$$\min_{w,b} P(w,b) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \quad \text{with} \quad \begin{cases} \forall i & y_i f(x_i) \geq 1 - \xi_i \\ \forall i & \xi_i \geq 0 \end{cases}$$

   (a) explain the purpose of the ξ_i variables.
   (b) if C is chosen to be very large, explain the impact on this SVM optimization and on the values of the ξ_i.
   (c) a user set the value of C at 0.1 for his training, but then he finds that he has bad scores at test time, so he decides to retrain the system with $C = 0.01$. Is it the right thing to do?

4. In term of deep architectures for computer vision:
   (a) is the multilayer perceptron a great solution for classifying large images?
   (b) are GoogLeNet and ResNet identical except the number of layers?
   (c) is VGG able to process images of any size as input?
   (d) are the number of layers of the ResNet architectures adapted during the supervised training?

5. In GAN, with the course notation G and D, at the end of a successful training on a Face dataset:
   (a) Does D perfectly separate real face images from generated ones?
   (b) Is it possible to use G to change the face of a smiling woman into the same one but angry?

6. Consider the following optimization scheme:

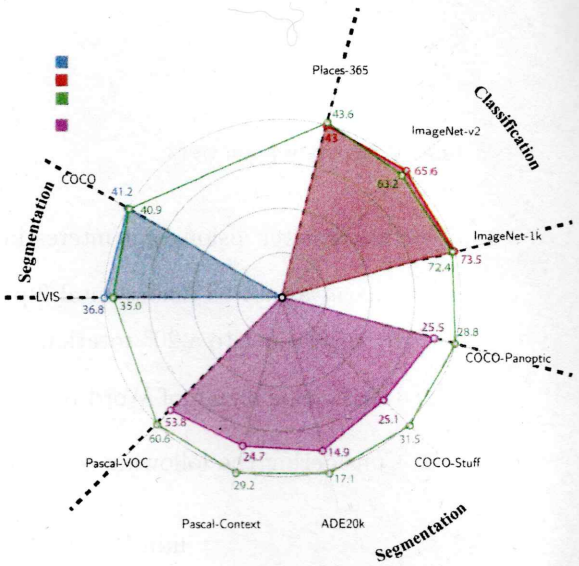
$$\min_G \max_D \left(\mathbb{E}_{\mathbf{x},\mathbf{y}\sim p_{\text{data}}(\mathbf{x},\mathbf{y})}[\log D(\mathbf{x},\mathbf{y})] + \mathbb{E}_{\mathbf{y}\sim p_{\mathbf{y}},\mathbf{z}\sim p_Z(\mathbf{z})}[\log(1 - D(G(\mathbf{z},\mathbf{y}),\mathbf{y}))]\right)$$

   explain what y is representing. Describe 2 examples with 2 different types of y.

7. Deep architectures based on transformers for vision have an essential property that differentiates them from ConvNet-type approaches. What is it and what does it allow?

8. We consider MNIST for digit classification (with training images and labels) and an extra colorMNIST dataset (with training images but without any labels). colorMNIST images are the same as MNIST ones but colored and with texture in the background. We want to learn a classifier that will be deployed later in a real context for colored images. Design a learning framework that contains two specific encoders E1 and E2, one specific branch to classify, and one extra binary classifier D. Explain the training, and the testing.

9. Let's consider the 4 methods represented by colors (blue, red, green and magenta) with their performance (the larger the %, the better the performance) on different evaluation tasks (one per axis represented).



   (a) The segmentation tasks on COCO and LVIS are instance segmentation tasks. What is the other type of segmentation called (performance shown between Pascal VOC and COCO-Panoptic on the figure)? Explain the main difference between the two.

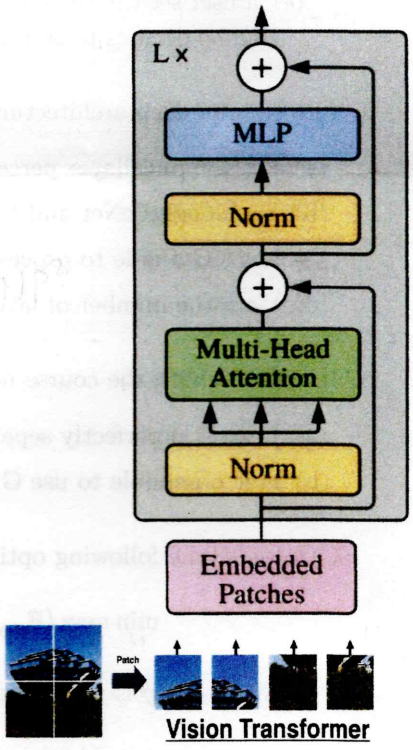
   (b) Is one method better than the others?

   (c) Can the areas of the various color curves be interpreted?

   (d) What's so special about the green method?

10. Vision Transformer framework

Let's consider the following deep architecture (seen in class) and the notations/algorithm:



$x \in \mathbb{R}^{H \times W \times C}$
$x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$
$N = \frac{HW}{P^2}$

$E \in \mathbb{R}^{(P^2 \cdot C) \times D}$
$E_{\text{pos}} \in \mathbb{R}^{(N+1) \times D}$

$z_0 = [x_{\text{class}}; x_p^1 E; x_p^2 E; \ldots; x_p^N E] + E_{\text{pos}}$

for $\ell = 1 \ldots L$
$z'_\ell = \text{MHA}(\text{LN}(z_{\ell-1})) + z_{\ell-1}$,
$z_\ell = \text{MLP}(\text{LN}(z'_\ell)) + z'_\ell$,

$y = \text{LN}(z_L^0)$

Explain x_p, E_{pos}, x_{class}, z_L^0 and y.