Homework 4: Detection and Classification

Kaveh Fathian, Email: fathian@ariarobotics.com

Handout: 2024-11-11
Due: 2024-11-20, at 3:00 PM on Canvas

Instructions:

- Homework is on a "**rolling"** basis and more **questions will be added until 1 week** before the due date. There will be an announcement (on Discord or in class) when new questions are added.
- For all problems in this homework, you can convert your images to **grayscale** for simplicity. So, no need to work with RGB images (unless you want to).
- You can get help from your teammates (or others) for all problems and/or code. However, you will need to code the problems and submit your report **individually** on Canvas. Reports/code that are **identical** will receive a grade of **zero**.
- The **quiz** will strongly resemble homework questions, and if you understood/coded the homework yourself, you will be able to answer the quiz questions immediately. Because the quiz will be closed-notes & no internet access, understanding the homework is crucial!
- **Deliverables**: You will submit a **single PDF** file to Canvas. The PDF must contain your **answers**, your **code** (copy-paste in the document), and any requested **outputs** (like images). For convenience, you may use Jupyter notebook and convert it to a PDF.
- Only use **images provided** in the homework material, as requested by each problem. Using any other image, will result in a **grade of zero**.
- **Grading:** This homework will be scaled to **10pts** of your final grade. Grading **rubric** will be posted on **Canvas** after the assignment due date.

Problem 1 (2 pts): Dalal-Triggs Pedestrian Detector
Use OpenCV's trained classifier provided in cv2.HOGDescriptor() and cv2.HOGDescriptor_getDefaultPeopleDetector() to detect pedestrians in in images "**csm1.jpg**", "**csm2.jpg**", and "**csm3.jpg**". You can use the template code provided in the homework material.

- Provide a copy of your code in the report
- Display **input images**
- Display **output images** with pedestrian detections **marked by rectangles**, as shown below

- Do you notice any issues with the detected pedestrians? Discuss at least **two major issues.**

Problem 2 (2 pts): Viola-Jones Face Detection
Use OpenCV's trained Haar classifier weights `'haarcascade_frontalface_default.xml'` in `cv2.CascadeClassifier()`, and write a script that detect faces in images "**csm1.jpg**", "**csm2.jpg**", and "**csm3.jpg**". You can use the template code provided in the homework material.

- Provide a copy of your code in the report
- Display **input images**

Homework 4: Detection and Classification

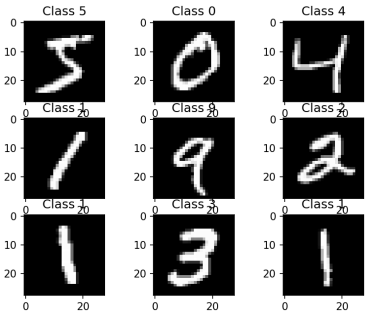
Kaveh Fathian, Email: fathian@ariarobotics.com

Handout: 2024-11-11
Due: 2024-11-20, at 3:00 PM on Canvas

- Display **output images** with face detections **marked by rectangles**, as shown below

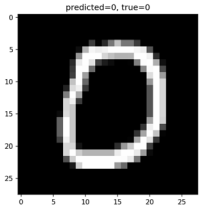
- Do you notice any issues with the detected faces? Discuss at least **two major issues.**

Problem 3 (3 pts): Digit detection
The MNIST handwritten digit dataset has 60,000 training samples, and 10,000 test samples. Each image is represented by 28x28 pixels, containing a grayscale values between 0-255 (https://yann.lecun.com/exdb/mnist/). Using the **k-nearest neighbors (kNN)** algorithm, design a classifier that can detect the digit from in an MNIST image. You may optionally use the `mnist_knn_template.py` template provided in the homework material. In your code, you must **NOT** use the **test samples** for training/classification; **only use the training** set.



Hint: Given a test point x_t, the kNN classifier finds the k nearest points $x_1, x_2, \dots, x_k$ to x_t in the training dataset. As the training data x_i includes correct labels y_i, that is, we have $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$, the predicted label for x_t can therefore be considered as the label that appears the most in the set of labels $\{y_1, y_2, \dots, y_k\}$.

- Provide a copy of your code in the PDF report
- Explain which **distance function** you chose for the kNN, and why
- Explain what **value of k** you used in kNN, and what is the impact of k (i.e., large k vs. small k)
- Display **image indexed 10 in the test samples** (zero indexed) and it's true and predicted class given by your kNN classifier (see the template code)



- Display the **precision** of the results on the first 1000 **test** samples

Problem 4 (3 pts): Image classification
The CIFAR-10 dataset consists of 60000, 32x32 color images in 10 classes. There are 6000 images per class, 50000 training images, and 10000 test images (https://www.cs.toronto.edu/~kriz/cifar.html).

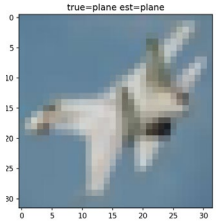
Homework 4: Detection and Classification

Kaveh Fathian, Email: fathian@ariarobotics.com

Handout: 2024-11-11
Due: 2024-11-20, at 3:00 PM on Canvas

Train a **support vector machine (SVM)** classifier that can identify the class of a given image. You may optionally use the `cifar10_svm_template.py` template provided in the homework material. In your code, you must **NOT** use the **test data** for training/classification; **only use the training** data.
Hint: Use the SVM module provided by the open-source Python machine learning library scikit-learn (https://scikit-learn.org/stable/modules/svm.html). Read the instructions, see the examples, and carefully review the tips and tricks on the scikit-learn website, as they can help you pre-process your data to get better results using SVM.

- Provide a copy of your code in the PDF report
- Explain if/how you **pre-processed the data**. Did you remove the mean, normalize, or remove redundant data?
- Explain which **kernel** you chose for the SVM, and why. Did you try different kernels? Which one performed better and why?
- Display **image indexed 10 in the test samples** (zero indexed) and it's true and predicted class given by your SVM classifier (see the template code)



- What is the **precision** (i.e., classification score) of your SVM on the **training set**? Compute and display the results. **NOTE: You must achieve a precision of greater than 0.27 on the training set, otherwise, you will not get any points for this problem!**

Problem 5: Review
To make sure you review and understand the lecture materials, answer the following questions. Please do **NOT** provide your answers in your homework submission as these questions are only for your review. The quiz **MAY** be based on some of these questions, but is not limited to them and you are expected to review the slides.

- What are the strengths and shortcomings of SVMs?
- What are the effects of the number of model parameters in terms of bias and variance? How can you identify if the model is overfitting or underfitting using bias and variance?
- Identify various recognition tasks given an image, e.g., classification, detection, identification, semantic segmentation, scene categorization, event recognition.
- What the challenges of object recognition in images?
- What are the 3 main components of the standard BOW pipeline?
- Describe how the visual dictionary is constructed for BOW, and how BOW vectors are generated for each image given a dictionary.
- What are 3 common classification methods given BOW vectors?
- How the precision and recall are computed in the image retrieval application (e.g., given a query image, and top 5 results in a dataset of 100 images)?
- Describe how VLAD descriptors are computed

Homework 4: Detection and Classification

Kaveh Fathian, Email: fathian@ariarobotics.com

Handout: 2024-11-11
Due: 2024-11-20, at 3:00 PM on Canvas

- Describe (in detail) how the HOG descriptor of an 8x8 image cell is computed. How many elements the HOG descriptor vector has?
- What are the key 4 steps in the Dalal-Triggs pedestrian detector? What is the classifier used by this algorithm? How pedestrians at different scales are detected?
- Describe (in detail) how the Haar features are computed for an image.
- What is the computation cost for computing a Haar feature based on an NxM Haar filter that is is half black and half white?
- Understand how a Haar feature can be computed given an Integral image.
- What are the key 4 ideas/steps in the Viola-Jones face detector?
- How does Viola-Jones detector builds a 'strong' classifier using just a small subset of all possible Haar features?
- How does the attention cascade work, and what is its use in the Viola-Jones face detector?
- What are the steps of the perceptron algorithm? Compute the weight updates for an example that classifies a set of 2D point that are labeled as green and red.
- What are input, hidden, and output layers in a neural network (NN)? Given an NN, identify the number of neurons, weights, and learnable parameters.
- Describe the steps of the gradient descent algorithm. What is the difference of gradient descent and stochastic gradient descent (SGD)? What are the properties of SGD algorithm for training NNs?
- Given a NN (i.e., layer structure and activation functions), use back propagation to compute the gradients of the loss function. Then, compute the weight updates using the gradient descent algorithm.