Assignment-2: Low Level Vision

Instructions:

[1] Any plagiarism will lead to award of **F grade** STRICTLY

[2] Use python only for the implementation of all the assignments

[3] Use NumPy to represent the vector and array

[4] Do not use the inbuilt functionality of any library including NumPy until suggest so.

[5] PyTorch must be used to implement the deep learning-based methods.

[6] One mark will be deducted for each late day.

[7] Submit via Moodle only. Email submissions won't be considered.

No.	Question	Marks
1	Data source: dataset generated in Assignment-1 Q-3 Write a program for the SVD from scratch using eigen decomposition on the data matrix X. Use the top k singular vectors to reconstruct the data matrix $\hat{X}$. Plot the reconstruction error $\|\|X - \hat{X}\|\|_F$ for $k = 1,2, \ldots$	10
2	Data source: Any standard/non-standards pair of images I. Implement the 8 point algorithm for the image rectification display the two images before and after rectification. II. Implement the Harris Corner Detection Algorithm plot the corners back to images and display.	20 10 10
3	Data source: Any standard single Face image-based dataset Write a program to I. Implement the Histogram of Local Binary Patterns (LBP) from screech using python. Read images from a directory Train/Val/Test and generate a Train.csv/Val.csv/Test.csv II. Use KNN classifier or One-vs-Rest SVM classifier (choose best distance metrics and K in KNN and best kernel and hyperparameters in SVM case using Val.csv) report the accuracy on Test.csv for best case of each III. Collect non-face images from any source with the same number as the face and train a binary SVM for face vs. non-face (*you can use the sklearn library for SVM*) IV. Implement a sliding window-based detection approach for any images having multiple faces and detect the face using the trained SVM model	20 10 5 3 2
4	Data source: Any standard Human/Person detection dataset	20

	Write a program to I. Implement the Histogram of Oriented Gradients (HOG) descriptor from screech using python. Read images from a directory Train/Val/Test and generate a Train.csv/Val.csv/Test.csv II. Use KNN classifier or One-vs-Rest SVM classifier (*choose best distance metrics and K in KNN and best kernel and hyperparameters in SVM case using Val.csv and report the accuracy on Test.csv for best case of each*) III. Collect non-Human images from any source with the same number as the Humans and train a binary SVM for Human vs. non-Human (*you can use the sklearn library for SVM*) IV. Implement a sliding window-based detection approach for any images having multiple persons and detect the persons using the trained SVM model	10 5 3 2
5	Data source: MNIST: http://yann.lecun.com/exdb/mnist/ Write a program to I. Use the following features and construct the Kernal Matrix for each using Linear Kernel, Polynomial Kernel with degree 2,3 and 5, RBF kernel with sigma=0.5, 1, and 2 a) Vectorization (flattened vector) of the image b) Histogram of Intensities c) Histogram of Local Binary Patterns (LBP) d) Histogram of Oriented Gradients (HOG) descriptor II. Display both Train and Test Kernel Matrices side-by-side using matplotlib for all the above case and write your observation. III. Display the effect of Kernel hyperparameters for Polynomial Kernel and RBF kernel IV. Use precompiled kernel into linear SVM for classification for all the cases in part-I (*you can use the sklearn library for SVM*) V. (Bonus): Implement SMO algorithm and use in place of IV.	20(+5) 3 3 3 3 3 2 3 +5