# BBM 418 - Computer Vision Laboratory

HACETTEPE UNIVERSITY
Department of Computer Engineering

**Programming Assignment-2**
Spring 2020-2021
Doç. Dr. Nazlı İkizler Cinbiş

**Due Date: 16 April 2021, 23:59**

## Scene Reconition

In this assignment, you will get familiar with basic image recognition techniques. First of all, you are going to preprocess your images and extract features using two methods; *Tiny Image* and *Bag of SIFT/SURF/ORB* features. After that as a classification technique you will use two techniques; *Nearest Neighbor* and *Linear SVM* classifier. At the end you will compare and analyze feature extraction and classification techniques. As a dataset you will use a scene dataset provided here [1].

## Feature Extraction

**Tiny Image Features:** One of the simplest feature extraction method for images is *Tiny Image Features*. The idea is simple; you just simply need to resize images to a small and fixed resolution; 16x16.
**Bag of SIFT/SURF/ORB Features:** Bag of word models[2] are used in computer vision as well as in natural language processing. The idea is; representing an image using visual words thus the classification model classifies based on histogram of the frequency of visual words. In this step; you first need to establish a vocabulary of visual words by sampling either *SIFT*, *SURF*, or *ORB* features from training set images.

## Classification Techniques

**Nearest Neighbor Classifier:** The nearest neighbor classifier will predict the category for every test image by finding the training image with most similar features. Instead of 1 nearest neighbor, vote can be based on k nearest neighbors which will increase performance.
**Linear SVM:** In a linear classifier, the feature space is partitioned by a learned hyperplane and test cases are categorized based on which side of the hyperplane they fall on. This function will train a linear SVM for every category and then use the learned linear classifiers to predict the category of every test image.

## Dataset

Dataset [1] consists of sample of scene images[3]. There are 6 categories; *Bedroom*, *Kitchen*, *Living-Room*, *Highway*, *Mountain*, and *Office*. Each of the image belongs to one the 6 categories.

## The Implementation Details

1. First, you need to create a train and test set out of the given dataset.
2. Second you are expected to create Tiny Image features and Bag of Visual Words. For Bag of Visual Words you can either use; SIFT, SURF, or ORB. You need to implement Bag of Words but you can use available libraries for SIFT, SURF, or ORB.
3. Then you are expected to train with your extraction features using K-Nearest Neighbor and Linear SVM classifier on the train set using with Tiny Image features and Bag of Visual Words separately.
4. You are expected to print accuracy results on test set for; Tiny Image Feature and K-Nearest Neighbor, Bag of Visual Words and K-Nearest Neighbor, Tiny Image Feature and Linear SVM, Bag of Visual Words and Linear SVM
5. You should pay attention to code readability such as comments, function/variable names and your code quality: 1) no hard-coding 2) no repeated code 3) cleanly separate and organize your code 4) use consistent style,indentation

6. You should use Python 3 for the assignment.
7. You may use available libraries for feature and classification methods. You need to implement Bag of Visual Words approach but you can use available libraries for SIFT/SURF/ORB, linear SVM, K-Nearest Neighbor.

## What should you write in the report?

1. Explain how you divide the dataset into train and test with relevent code snippet you have written. Give train and test set image counts per category.
2. Explain how you extracted Tiny Image and Bag of visual word features with relevent code snippet you have written. Explain how you set the parameters with your reasoning.
3. Explain how you implement Bag of Visual Words approach.
4. Explain how you trained and tested Nearest Neighbor and Linear SVM classifiers with relevant code snippets. Explain how you set the parameters with your reasoning.
5. Explain how you calculated accuracy with relevant code snippets for all of the models; Tiny Image Feature and K-Nearest Neighbor, Bag of Visual Words and K-Nearest Neighbor, Tiny Image Feature and Linear SVM, Bag of Visual Words and Linear SVM. Which accuracy metric you have used and why?
6. Plot confusion matrices for all 4 models on the test set.
7. Put 5 test set image samples you have correctly classified for each category.
8. Put 5 test set image samples your model fails to classify correctly and explain possible reasons for it.
9. Compare 4 experiments with respect to your accuracy metric and confusion matrix, correct and wrong image classification results. Explain which of the 4 models works better for the problem.
10. You should write your report in LaTeX
11. You should give visual results by using a table structure.

## What to Hand In

Your submission format will be:

- README.txt *(give a text file containing the details about your implementation, how to run your code, the organization of your code, functions etc.)*
- code/ *(directory containing all your code)*
- report.pdf

Archieve this folder as **b<studentNumber>.zip** and submit to `https://submit.cs.hacettepe.edu.tr`.

## Grading

The assignment will be graded out of 100:

- CODE: 0 (no implementation), 15 (a partial solution – only feature extraction), 30 (a partially correct solution – get results for 2 models), 50 (a correct solution) and REPORT: 50

## Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

# References


1. https://drive.google.com/file/d/1c_kTQNm-SXN6-ZmVcC8ZMOiLssRhy-1I/view?usp=sharing
2. Sivic, J., & Zisserman, A. (2008). Efficient visual search of videos cast as text retrieval. IEEE transactions on pattern analysis and machine intelligence, 31(4), 591-606.
3. A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," IJCV, 2001.