Practical Training - Task Sheet 2
Image Classification

1 Task Description

This task is about the classification of pictures. For this purpuse we will take a look at the image collection *Caltech101*[1], which contains images from 101 different classes, e.g. plane, camera or elephant. The task of this practical training is to perform a classification using this data and determining good parameters for the chosen classifier. As for the classification algorithm you are free to choose one of the ones discussed in the lecture (Nearest Neighbor, Bayes, Support Vector Machine, Decision Tree, Random Forest or Neural Network)

We will not work on the picture data directly. Instead, we will use a feature called `Edge Histogram`[2] The feature has already been extracted from the images and is available for you on Moodle as well, thus you do not have to touch the images at all. All in all you will receive the following files as part of the task:

- `Images.csv`: Contains the images which are represented by an image ID and the corresponding class. The ID and the class are separated by a semicolon. The first line of the file contains the number of images. An example of what the file might look like is shown below:

  ```
  3
  1;airplanes
  32;beaver
  454;chair
  ```

- `EdgeHistogram.csv`: Contains the feature data for the `Edge Histogram` feature for the images. Each line here consists of the image ID followed by the feature vector. The ID and the vector are separated by a semicolon. The individual dimensions are also separated by a semicolon. The first line contains the number of images followed by the number of dimensions separated by a semicolon. An example of what the file might look like is shown below:

  ```
  3;4
  1;12;32;44;8
  32;69;89;11;42
  454;130;22;45;76
  ```

When training, the following amounts of training images per class should be tried out: 3, 5, 10, 15. Also, you should choose a suitable strategy for selecting the training and test

[1] https://www.tensorflow.org/datasets/catalog/caltech101

[2] Thomas Sikora. The MPEG-7 visual standard for content description - an overview. IEEETrans. Circuits Syst. Video Techn., 11(6):696-702, 2001.3

images. Depending on the amount and your chosen classifier you should also try to find the hyperparameters that yield the highest accuracy.

Write or use an appropriate program for your experiments and make a transcript of your experiment setup. Describe, present, evaluate and discuss your results.

2 Submission

1. Submission deadline: February 6th, 2023 (11:59pm)
2. The submission is done over Moodle
3. Your digital submission must contain:
   a) Source code of the program (if you wrote one)
      - You do not have to implement the selected classification method yourself!
      - Instead, you may use libraries or applications (e.g. Weka, Scikit learn, Tensorflow, etc.), which you should specify in your documentation.
   b) A short scientific report of your experiments to understand your results (around 2 - 3 pages). This includes, for example:
      - Experiment setup
        - How were the training / test image sets selected?
        - How many different runs were executed for each amount of training images?
        - How were the parameters for the classification method optimized?

        ⋮
      - Presentation of results
        - Tables (e. g. exemplary classifiction results, test results per case, . . .)
        - Diagrams (e. g. relationship between training set size, classification parameters, accuracy, . . .)
        - Visualisations of the trained classification model (if applicable, e. g. plot of a decision tree, . . .)

        ⋮
      - Discussion of the results
   c) A short textual explanation of the installation, execution and operation (README.txt) of your program for the experiments.

3 Discussion

The discussion date will be agreed upon individually with each group.