

Cairo University

Faculty of Engineering

Computer Engineering Department

Second Year

**Pattern Recognition and Neural Networks**

**Term Project**

Handwriting-Based Gender Classifier

Presented by: Team 9

Ahmed Gamal Abdelsamie Sec: 1 BN: 3

Abdelrahman Jamal Sec: 1 BN: 40

Karim Taha Sec: 2 BN: 8

Mostafa Mahmoud Kamal Sec: 2 BN: 28

Ahmed Adel Sec: 1 BN: 7

Spring 2022

Literature Review

There have been 2 essential papers where we drew our approaches from:

1. Automatic analysis of handwriting for gender classification

<https://link.springer.com/article/10.1007/s10044-014-0371-0>

The main features used in this paper were:

* 1. Slant and curvature: comprised distribution of chain codes, distribution of curvatures, and distribution of segment slopes
  2. Fractal features
  3. Textural features: comprised Grey Level Co-occurrence Matrix (GLCM) and local binary patterns (LBP)

The models used to train using these features were Support Vector Machine (SVM) and Artificial Neural Networks (ANN).

1. Handwriting Based Gender Classification Using COLD and Hinge Features

<https://www.researchgate.net/publication/342767331_Handwriting_Based_Gender_Classification_Using_COLD_and_Hinge_Features>

The paper mainly used 2 features:

1. COLD feature: “a feature inspired by the shape context descriptor, with respect to the extracted information into a log-polar histogram”
2. Hinge feature: a contour-based feature that captures the curvature of ink in document images

Both features were trained using SVM

1. Improving handwriting based gender classification using ensemble classifiers

<https://coek.info/pdf-improving-handwriting-based-gender-classification-using-ensemble-classifiers-.html>

This paper using discussed using different machine learning models to improve accuracy including:

1. ANN
2. SVM
3. K- Nearest Neighbor (KNN)
4. Decision Trees (DT)
5. Random Forest (RF)

Considering these options, the time we had to implement the project, and availability of resources implementing these features and models, we decided to use the following features:

1. GLCM
2. Hinge
3. COLD
4. Hinge + COLD

and decided to train using the following models:

1. SVM
2. Random Forest
3. Decision Tree

We also tried applying PCA before training to some of the extracted features.

System Description

The code is partitioned so that after each feature extracted, training and scoring is done. We have done this to allow for ease of readability and collaboration.

Based on the previously mentioned features and models, we have proposed the following pipeline

1. Preprocessing Module

We have mainly used the CMP23 dataset. We first read the images from the input directory, and split the images into training set and testing set. The preprocessing module mainly applies morphological closing to enhance the handwriting lines, performs median blur to get rid of noisy pixels, and finally applies binary thresholding.

Note that some of the images in the dataset after preprocessing may contain dark spheres or spots, as a result of using flash during scanning the data set.

1. Feature Extraction

Extraction functions for the features we selected were available and pre-implemented, we have extracted the GLCM using the energy and homogeneity features using skimage library. It provided necessary functions including: skimage.feature.texture.greycomatrix(image, ...), skimage.feature.texture.greycoprops(P[, prop]).

For Hinge and COLD features, we have used functions to extract the features from the following GitHub repository: <https://github.com/Swati707/hinge_and_cold_feature_extraction>

We have forked this repository and added some our preprocessing methods to the existing methods.

1. Training Module

After extracting each feature, we try out the three previously mentioned models on each feature. All three models are available in the sklearn library.

1. Performance Analysis

We perform scoring after training each feature using the three mentioned machine learning models. An average of the scores for the features and models is as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | GLCM | Hinge | Hinge - PCA | COLD | COLD - PCA | Hinge + COLD |
| SVM | 62% | 62% | - | 62% | - | - |
| Decision Tree | 61% | 57% | - | 65% | - | - |
| Random Forest (500 estimators) | 61% | 75% | 60% | 74% | 60% | 77% |

We have selected the Hinge classifier since it offers a high accuracy and reasonable time when run using the test script, as opposed to the Hinge + COLD features which have a slightly better accuracy but take almost twice as much time.

Future Work

We may try to extract fractal features, or slant and curvature features in future work, and implement a neural network to perform training.

Workload Distribution

|  |  |
| --- | --- |
| Ahmed Gamal | Hinge and COLD features extraction  Train using SVM and RF |
| Abdelrahman Jamal | Hinge and COLD features extraction  Train using SVM and RF  Concatenated Hinge + COLD |
| Karim Taha | GLCM feature extraction  Train using SVM and RF |
| Mostafa Kamal | Preprocessing |
| Ahmed Adel | Train using Decision Tree Classifier  Applying PCA |