Cairo University

Faculty of Engineering

[CMPS446 - Image Processing and Computer Vision]
Signature Verification

Team Members :

Yasmin Hashem Niazy 4200014
Sohad Hossam ELdin 1190019
Bassant Hisham 1190018
Yasmin Zaki Bassiouny 1190352

Project Idea

Abstract:

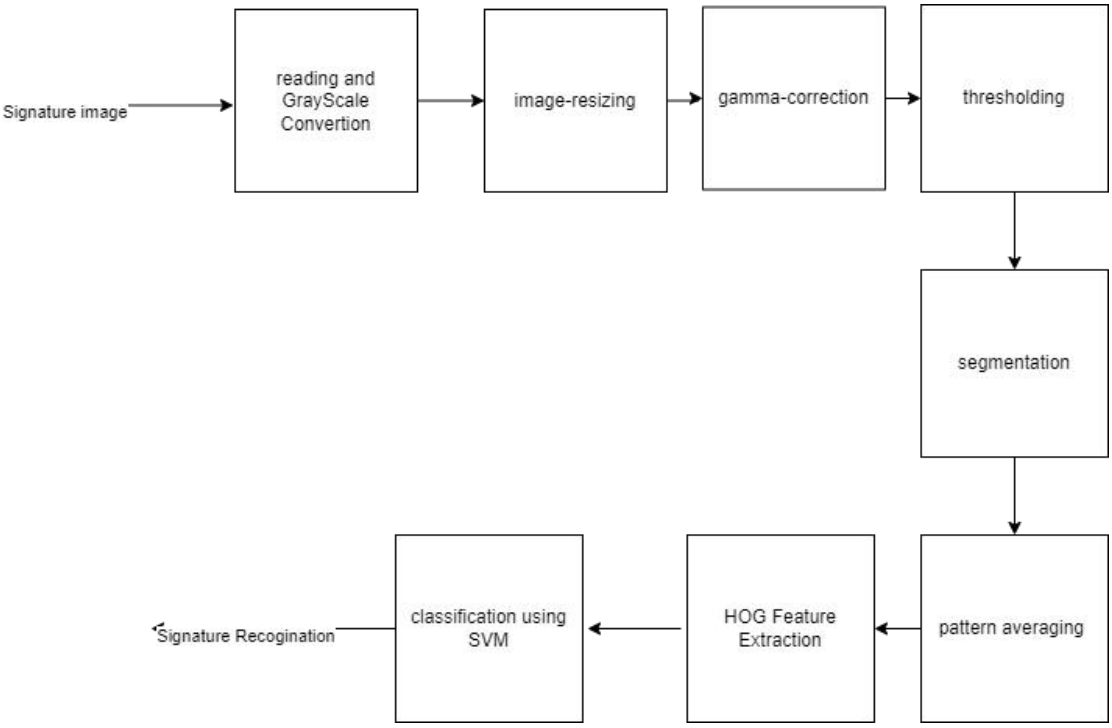
As technology advances, it opens up new possibilities for enhancing security frameworks by introducing various methods to identify individuals. One of these methods involves recognizing signatures. In the context of an image processing and computer vision course project, an offline signature recognition system is presented. This system employs image processing techniques. The preprocessing of signatures includes tasks such as converting color images to grayscale, applying filters, making adjustments, setting thresholds, and using edge detection methods. Additionally, image scaling is employed to speed up processing. Classification using a backpropagation neural network with specific numbers of neurons and hidden layers.

Applications and needs:

- **Employee Time and Attendance:** To ensure accurate tracking of employee attendance, signature recognition can be used as a biometric timekeeping system.
- **Document Authentication:**This technology can be used to verify the authenticity of signed documents, such as legal contracts, financial agreements, and official paperwork. It can help ensure that signatures have not been forged.
- **Identity Verification:** It can be applied in identity verification processes, such as during the issuance of identification cards, passports, or driver's licenses.

- **Security at ATMs:** To prevent unauthorized access to automated teller machines (ATMs), users' signatures can be verified before granting access to financial transactions.

Block Diagram



Methods:

1. Read the RGB images: The project starts by reading images of signatures.
2. Convert the images into gray-scale: images are converted into grayscale. making them easier to work with.

3. Re-scale the image size to 256x256 pixels

4. Adjust the image using gamma correction: to enhance the intensity of the pixel values in the image, improving image quality and preparing it for further processing.

5. Thresholding :Fixing the threshold to separate the signature from the background.

6. Use Canny edge detection for segmentation: to identify edges or boundaries within the image. to segment the signatures from the rest of the content in the image.

7. pattern averaging to re-scale the image to 64x64 pixels: to reduce the image size while preserving essential features.

8. HOG Feautre extraction : Compute Histogram of Oriented Gradients (HOG) features from the preprocessed image. HOG features capture the shape and texture information of the signature.

9. Support Vector Machine (SVM): Use a Support Vector Machine for recognizing and classifying the signature.

10. Train the SVM: Before it can make accurate predictions, the SVM needs to be trained. This involves feeding it a dataset of images with known signatures and their corresponding labels so that it can learn to recognize and classify signatures.

Needed non-primitive functions:

- RGB2Gray
- gamma correction (implemented from scratch)
- Canny edge detection
- Resize from CV2
- HOG from skimage
- PCA reduction from Sklearn
- Machine Learning algorithm (SVM)

References:

- Kiran, P. V. R. S., Parameshachari, B. D., Yashwanth, J., & Bharath, K. N. (2021). Offline signature recognition using image processing techniques and back propagation neuron network system. *SN Computer Science*, *2*(3). https://doi.org/10.1007/s42979-021-00591-y

- Taşkıran, M., & Çam, Z. G. (2017). Offline signature identification via HOG features and artificial neural networks. *IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*. https://doi.org/10.1109/sami.2017.7880280

- Özgündüz, E., Şentürk, T., & Karslıgil, M. E. (2005). Off-line signature verification and recognition by Support Vector Machine. *ResearchGate*. https://www.researchgate.net/publication/228653202_Off-line_signature_verification_and_recognition_by_Support_Vector_Machine/link/09e4150bb11ec25831000000/download