Optimization of Eigenface Algorithm for Facial Recognition Using AVX2

A Project Update Presented to
the Faculty of the College of Computer Studies
De La Salle University

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Science in Computer Science

by
Ibaoc, Christian Gabriel P.
Sia, Chantal Hyacynth S.
Valero, Nigel Kristoffer C.
Veloso, Antonio Luiz G.

Submitted to:
Professor Roger Luis Uy

June 27, 2024

Project Updates Summary:

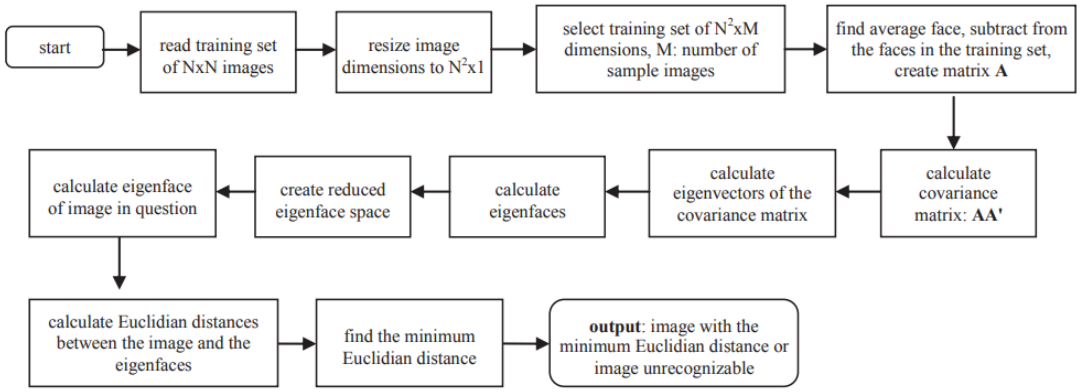
1. Adjustment of Implemented Language
2. Incomplete Implementation of Algorithm without Optimizations
3. Additional Libraries

I. Adjustment of Implemented Language

In the initial proposal, the group planned to implement the program in C, utilizing the OpenCV library for any image acquisition. This is because the group was misinformed about the support that OpenCV has available, and while they do support C, that particular support has ended almost a decade ago. The implementation of the library itself at that time was not optimal, and it led to errors that the group could not resolve. The next idea was to use C and its default libraries, like IMF or DirectShow to somehow connect a camera, but the code to implement this part is simply impractical to its function, and was also not working properly. Moreover, the samples provided by the SDK's, which at this time were over 10 years old, were already incompatible with Visual Studio, leading to even more problems. Since the original implementation was in C, with external calls to assembly, the group decided to implement it instead in C++, still with external calls to assembly, and including the OpenCV library.

II. Incomplete Implementation of Algorithm without Optimizations

So far, the group was able to implement a significant part of the algorithm, specifically up to the eigenvector calculations, in C++. The Eigenvalues are approximately calculated by applying the Jacobi Rotation Method to the Covariance Matrix, and the Eigenvectors are also a byproduct of this method. To recall, here is the flowchart of the Eigenface algorithm:



The next step is to calculate the eigenfaces as well as continue on with the recognition phase. The accuracy of the eigenvectors and eigenvalues are also being checked through multiple online calculators. These calculations were initially implemented manually to emulate matrices and vectors without optimizations to better compare it to the optimized version, but the group also included libraries meant for calculations involving matrices, mainly for efficiency and checking purposes. The manual implementation will be finalized once simple optimizations are implemented. In light of this incomplete implementation, no SIMD has been implemented, although the group has been taking note of potential methods to implement it.

III. Additional Libraries

The group added additional libraries to the implementation, referring to the previous section. For the library-implemented matrix operations, the Eigen library was included. The Cmath library was also included for the Jacobi-related operations, namely the basic trigonometric functions.