Abstract

Current existing facial recognition algorithms are sensitive to gross variation in lighting direction and facial expression. The Eigenface technique, is a method based on linearly projecting the image space to a low dimensional subspace, and then finding and comparing linear combination of features that maximizes total variance data [BHK97]. It has been shown that this method creates errors, because classes are not considered and discriminative information may be lost throwing away components [BHK97]. The Fisherface method uses Linear Discriminant Analysis to maximize the between-classes to within-classes scatter ratio [BHK97]. However, just like the Eigenface method varying light across the face will cause error [BHK97][GBK01]. Experimental results demonstrate that the proposed Fisherface method has error rates that are lower than those of the Eigenface technique for tests on the Harvard and Yale Face Databases [BHK97]. The Local Binary Patterns Histograms method utilizes a 2D texture analysis by describing only local features of an object [AHP07]. This means that features will be extracted with low-dimensionality implicitly. However, the Local Binary Patterns method will then suffer, because it is not robust towards scale, rotation, or translation [KM01].

The objective of our work is to implement these face recognition techniques on a mobile device to verify the above empirical characterization of the techniques with a camera embedded on the device. We develop an Android application using OpenCV library, which implements various image processing and computer vision algorithms. The presentation provides a quick tutorial of how to incorporate OpenCV on Android application development and results on our performance characterization of the face recognition techniques.

Using The Application

![Face detection and recognition on Android using OpenCV](image)

Eigenfaces

The Eigenfaces span a subspace of Eigenvectors, a non-zero vector that, when the matrix multiplies, yields a constant multiple, that are associated with the largest Eigenvalues, a variable within the formula that makes up Eigenvectors. This makes the “face space” whose origin is the average face. Recognition is then performed by computing the distance from a new face after it has been projected onto the already existing face.

Fisherfaces

The Fisherfaces are computed in an almost similar way. The Fisherfaces are mapped so that the sample vectors of the same class in a single spot of the feature representation and those different classes as far apart from each other as possible. This is known as discriminant analysis(DA). A commonly known form of DA in Linear Discriminant Analysis(LDA) which was proposed by R.A. Fisher. When LDA is used to find the subspace representation of a set of face images, the resulting basis vector defining that space are known as Fisherfaces.

Local Binary Pattern Histograms

In the LBP approach for texture classification, the occurrences of the LBP codes in an image are collected into a histogram. The LBP uses texture descriptors to build several local descriptions of the face and combine them into a global description. The classification is then performed by computing simple histogram similarities. Such local descriptions have been gaining interest lately which is understandable given the limitations of the holistic representations.

Introduction

Our application was built using the Android SDK(software development kit), Android NDK(native development kit), Javacpp, javacv, and javacv-android-arm.jar. This allowed us to write code using the android language as well as using OpenCV and C/C++ libraries. Testing was done by using both the Android Virtual Device Emulator as well as tangible mobile devices. OpenCV was designed for computational efficiency with a strong focus on real-time applications. It is written in optimized C/C++, and can utilize multi-core processing.

References


Acknowledgements

We thank National Science Foundation (CCF-1117439) and Susquehanna University for their support of our research.