Tender Response

to Facial Recognition System for Communities

BiTwise Developers

18 March, 2014

Contents

1 General

1.1 Purpose of Document

This document provides a description of the proposed solution submitted by Bitwise Developers in response to the invitation for bids on tender for the Facial recognition system for communities from Quant Solutions.

1.2 Introduction

A community watch has identified a need to monitor and identify individuals who live and operate in their comminity, in order to distinguish between people who belong in the area, and those who might be up to no good. A possible solution to the problem has been identified by Quant Solutions, who envision a computer system that can actively monitor and identify individuals who operate in the area by using IP camera surveillance, and a back-end database to process and store captured images. The processed data can then be queried by a member of the community watch or police, who is authorised to and tasked with interrogating suspicious individuals. This can be done either through a mobile device, or a web interface.

2 Executive Summary

2.1 Stakeholders

2.1.1 Project Owner

Client: Quant Solutions (pty) ltd

Contact Person: Kobus Wolvaardt

E-mail: kobuswolf@gmail.com

2.1.2 BiTwise Developers

BiTwise Developers are a unique collection of aspiring IT professionals, each with ambition and sound judgement. We are all in our final year of study at the University of Pretoria. We strive to accomplish each task effectively and efficiently, in such a way that it always meets our high standards. We are responsible and reliable. Each individual brings something unique to our team. We all get along with each other and work well as a unit.

Lead Developer: Jacques Lewis

E-mail: u28183488@tuks.co.za

Developer: Priscilla Hammond

E-mail: u11025477@tuks.co.za

Developer: Francois Oberholzer

E-mail: u12039803@tuks.co.za

3 Technical Proposal

3.1 Project Description

3.1.1 Problem Statement

A number of real time video streams, each with possibly different resolutions or orientations, need to be monitored for faces of individuals who operate in and around a neighbourhood. These faces need to be analysed and given individual identities, such that a database can be established and maintained and later used to identify specific individuals.

3.1.2 Objectives of the project

- Gather facial feature data to be used to identify individual people
- Classify each individual
- Use a client application to attempt to identify a person of interest against the database of individuals

3.2 Proposed Solution

We believe that we can solve the problem by using our knowledge and resources, and by applying our skills as software developers. We propose a possible solution as follows:

3.2.1 Solution Overview

Our solution is to employ OpenCV libraries to do most of the facial detection and recognition tasks. These include, but are not limited to:

- Detecting a face in a video frame
- Cropping an image to remove superfluous pixels
- Pre-processing the cropped image
- Creating Eigenfaces for both the dataset of known faces, as well as for comparing input faces to known faces

Our solution includes an interface that allows an authorised member of the community watch to attempt to identify an individual by providing an image of that individual, which will then be compared to the set of known faces in the database. We also privide a management component, from where the system can be managed and monitored.

3.2.2 Component Overview

- Facial Detection and Tracking (OpenCV, c++)
  - Monitor video stream
  - Identify faces in video frames
  - Follow faces as they move within a stream

- Image Pre-Processor (OpenCV, c++)
    - Crop face image so only facial features remain
    - Align a face to an optimal position
    - Prepare image (Greyscale, Equalisation etc.)
- Facial Feature Extractor (OpenCV, c++)
    - Compose Eigenfaces
- Facial Feature Matcher (OpenCV, c++)
    - Compare Eigenfaces to known faces
    - If the comparison has a high confidence, a match was found,
    - Else, a new individual may have been classified
- Image Data Manager (OpenCV, c++)
    - Manages the image database of known/classified faces
- System Administration (Either web based or Java application)
    - Provide functionality to effectively and efficiently manage users of the system
    - Provide functionality to identify and classify faces that are in the known faces database
    - Provide a monitoring and reporting facility
- User Interfaces (Web and Android)
    - Provides the interface for users of the system to upload and compare images of persons of interest

4 Project Deliverables

4.1 Documentation

- Functional Requirement Specification
- Architectural Design Specification
- Test Plans
- User Manual
- Installation Manual

4.2 Source Code

- All project source code
- All test cases
- Build scripts
- Deployment scripts

5 Communication Channels

Meetings: BiTwise Developers would like to suggest frequent meetings with the client, at the convenience of the client.

E-mail: All BiTwise members will be reachable through their respective e-mail addresses throuhgout the duration of the project.

Phone: All BiTwise members will be contactable by their respective cellphone numbers.

Other: BiTwise Developers are open to other means of collaboration, for example Google hangouts, or Skype etc.