MULTIMEDIA IMAGE PROCESSING Using OpenCV

Votary Softech Solutions Pvt. Ltd.

Plot No: 76, Lumbini layout,  
Near Euro school,  
Gachibowli-I (V), Hyderabad,  
Telangana - 500032,  
India.

**Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version (x.y) | Date of Revision  (DD-MM - YYYY) | Description of Change | Reason for Change | Affected Sections | Approved By |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Approval History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version (x.y) | Prepared By | Reviewed By/Date | Approved By/Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contents**

[1 Define 4](#_Toc483323802)

[1.1 Objective 4](#_Toc483323803)

[1.2 Deliverables 4](#_Toc483323804)

[1.3 Prerequisites 4](#_Toc483323805)

[1.4 Assumptions 4](#_Toc483323806)

[1.5 Limitations 4](#_Toc483323807)

[1.6 Business Risks 4](#_Toc483323808)

[1.7 Glossary 4](#_Toc483323809)

[2 Specify 4](#_Toc483323810)

[2.1 Scope 4](#_Toc483323811)

[2.2 Schedule and Milestones 4](#_Toc483323812)

[2.3 Resources 4](#_Toc483323813)

[2.4 Cost 4](#_Toc483323814)

[2.5 Market Analysis 5](#_Toc483323815)

[2.5.1 Competitor Information 5](#_Toc483323816)

[2.5.2 Substitutes in Market for Product 5](#_Toc483323817)

[2.5.3 Targeted Domains 5](#_Toc483323818)

[3 Architecture 5](#_Toc483323819)

[3.1 Block Diagram 5](#_Toc483323820)

[3.2 Internal Interfaces 5](#_Toc483323821)

[3.3 External Interfaces 5](#_Toc483323822)

[4 Design 5](#_Toc483323823)

[4.1 Flowcharts 5](#_Toc483323824)

[4.2 Development Environment 5](#_Toc483323825)

[4.3 IP Identified 5](#_Toc483323826)

[5 Implement 5](#_Toc483323827)

[5.1 System Requirements 5](#_Toc483323828)

[5.1.1 <Module based> Functional Requirements 6](#_Toc483323829)

[5.2 Revenue Realization Plan 6](#_Toc483323830)

[5.3 User Acceptance Criteria 6](#_Toc483323831)

[5.4 Quality Plan 6](#_Toc483323832)

[6 Validate 6](#_Toc483323833)

[7 Deploy 6](#_Toc483323834)

[8 Maintain 6](#_Toc483323835)

# Define

Image Acquisition system in the Gateway acquires raw data either from the Camera or from local storage or database .Image Acquisition system in the Gateway is implemented in python.

Image Processing System process data(captured image) to identify the things or Images or objects based on the pattern implemented by the Algorithm. Image Processing System in the Gateway is implemented in python.

Database to store the processed image and details from the image processing system which is implemented in python.

## Objective

- Image Acquisition, Image Processing and Face Recognition from the raw data either from the Camera or from local storage or database .

## Deliverables

* Applications which runs on Raspberry pi board.

## Prerequisites

* Raspberry pi board.
* Camera.
* Python & OpenCV

## Assumptions

Pre-configured data should be present in database for image processing.

## Limitations

Process one image at a time.

## Business Risks

NA

## Glossary

|  |  |
| --- | --- |
| OpenCV | Open Source Computer Vision. |
| NumPy | Numeric Python |
| PIL | Python Imaging Library |

# Specify

* **Image Acquisition system** in the Gateway acquires raw data either from the Camera or from local storage or database.
* **Image Processing System** process data(captured image) to identify the things or Images or objects based on the pattern implemented by using OpenCV Algorithms.
* **Database** to store the processed image and details from the image processing system.

# Architecture

## Block Diagram



Input Image Data Base

output



Name : Baby

Age : 1 year

Gender : Female

# Design

## Flowcharts

→ Flow Chart for B&I

Image Acquisition

Compare Image With

Database Images

Match

Display

Details

Store image in

Database

With meta

data

Yes

No

## Development Environment

* Vim Editor
* Python Compiler
* GCC
* OpenCV

# Implement

## System Requirements

### Functional Requirements

|  |  |  |
| --- | --- | --- |
| SL # | Requirements | Description |
| 1 | REQ – API001 | Implement GUI |
| 2 | REQ - API002 | GUI should be able to access Images from local directory. |
| 3 | REQ - API002 | GUI should be able to access Images using Camera. |
| 4 | REQ - API002 | GUI should be able to access Images from Data Base.. |
| 5 | REQ - API002 | Train the Data |
| 6 | REQ - API002 | Face detection of a persons in a given Image. |
| 7 | REQ - API002 | Face Recognition of a persons in a given images. |
| 8 | REQ - API002 | Write wrappers for OpenCV Modules. |
| 8 | REQ – API002 | Write wrappers for BSP. |
| 9 | REQ - API002 | Write Camera Driver and Interface to camera. |
| 10 | REQ - API002 | Build Os Image using Yocto and Linaro. |

## User Acceptance Criteria

GUI