Facial Recognition Using Open CV

Sneha Arora[1] Rajiv Munjal[2]
[1]M.tech scholar [2]Assistant Professor
[1,2]Computer Science & Engineering Department
[1,2]Cbs Group Of Institutions,India

Abstract— As we know the internet is advancing so there is growing interest in computer vision.and so face detection is one of the growing and popular area of research in computer science. Therefore we use open cv with open source computer vision library, it uses library function which are used for loading image, creating window,save image and access pixels in a particular domain.

***Key words*: -** face detection, recognition, open CV, eigenface

I. Introduction

The main motto of this thesis is to providing interaction routine between human and machine or human –machine interaction through face detection and face recognition and we use open source computer vision library to work on it so that we could get much better results.

We use regular web camera and a machine which is used to recognise and detect a person or any new face , a custom login screen will appear and all filter pixels got activated and based on facial features it filters the ability to detect the face approximately 95%.

The objective is to provide a set of algorithms which identify face and its pixels in much relevant manner which provides successful recognition rate upto 95%.- As we know the internet is advancing so there is growing interest in computer vision.and so face detection is one of the growing and popular area of research in computer science. Therefore we use open cv with open source computer vision library, it uses library function which are used for loading image, creating window,save image and access pixels in a particular domain. We use open source computer vision library which gives advance vision research by providing not only open but also optimizd code for basic vision infrastructure.

It gives common infrastructure that developers could built on so that code would be more easily readable and transferrable.

II. Introduction To Open Cv

Open cv stands for open source computer vision library which includes over 500 functions implementing computer vision, image processing and general purpose numeric algorithms. It is very portable and very efficiently implemented in c/c++.It focuses mainly on real time image processing and facial recognition system.

The library is written in C and C++and runs under Linux, Windows and provides interfaces for Python, Ruby, Matlab and other languages. OpenCV library contains abundant advanced math functions, image processing functions, and computer vision functions that span many areas in vision.

A. Basic Class

OpenCV 1.0 includes the following five modules [4]:

1. CxCore: Some basic functions (various data types and basic operations, etc.).
2. CV: Contains image processing and computer vision function(image processing, structure analysis, motion analysis, and object tracking, pattern recognition, and camera calibration).
3. CvAux: Some experimental functions (View Morphing, Three-dimensional Tracking, PCA, HMM).
4. HighGUI: Contains user interface GUI and image/video storage and recall.
5. CvCam: Camera interface (After OpenCV 1.0 version, CvCam will be completely removed.). B. Environment Configuration

Under windows vista using visual studio 2005 to call the library of OpenCV, my setting steps are as follows: The first step, download and install OpenCV. Download the appropriate version according to the operating system. For Linux, the source distribution is the file opencv-1.0.0.tar.gz; for windows, you want OpenCV_1.0.exe.

The second step, it is necessary for global settings in Visual
Studio 2005:

1. In Visual Studio, choose “Tools->Options”;
2. In popup dialog, then choose “Projects and Solutions -> VC++ Directories”;
3. In the above dialog, from drop-down list box “Show Directories for:” select “Library files”;
4. In Library files listing, add such a path as “C:\Program Files\OpenCV\lib”;
5. From 2) dialog’s drop-down list box “Show Directories for:” choose “Include Files”; then add the following directory:
   “C:\Program Files\OpenCV\cv\include” C:\Program Files\OpenCV\cxcore\include” “C:\Program Files\OpenCV\otherlibs\highgui” “C:\Program Files\OpenCV\cvaux\include”
   “C:\ProgramFiles\OpenCV\otherlibs_graphics\includ

Choose “source files” of the drop-down list box “ShowDirectories for:”, then add the following paths
to it:
“C:\Program Files\OpenCV\cv\src”
“C:\ProgramFiles\OpenCV\cxcore\src”
“C:\Program Files\OpenCV\cvaux\src”
“C:\Program Files\OpenCV\otherlibs\highgui”
“C:\ProgramFiles\OpenCV\otherlibs_graphics\src”

Doing it like this, the global variable has been set well in Visual Studio 2005. The third step, we can create an project For example, create a project named OpenCV Video Encryption with“Win32 Application”. Do not forget to include the following several header files which should be put behind
stdafx.h, otherwise it will go wrong. They are #include <cv.h>, #include<cxcore.h>,
#include<highgui.h> and #include <cvcam.h>.
If lucky, we can compile successfully now. Maybe there are some link errors, so we need to open “Project” -> “Properties”
and add the following lib libraries to “Linker” -> “Input” -> “Additional Dependencies”, including cxcore.lib, cv.l

III. BASICS OF FACIAL RECOGNITION

Face detection is mainly concerned with finding whether or not there are any faces in a given image and if present return the image location and content of each face.This is the first step of any automatic system that analyses the information contained in faces.Automatic face recognition try to find the identity of a given face image according to their memory and the memory is generally stabilised by a training set.

A general statement of the face recognition problem (in computer vision) can be formulated as follows: given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces.

Facial recognition works in two stages

A. Face detection

In this the photo is searched furst and then crop and extracted.there should be fixed pose according to the stimulations.

Face recognition detected and cropped face is compared with database of known faces ot know who the person is.

Since 2002, face detection can be performed fairly easily and reliably with Intel's open source framework called OpenCV [1]. This framework has an in-built Face Detector that works in roughly 90-95% of clear photos of a person looking forward at the camera. However, detecting a person's face when that person is viewed from an angle is usually harder, sometimes requiring 3D Head Pose Estimation. Also, lack of proper brightness also create problems. CV namespace contains image processing and camera calibration methods. The computational geometry functions are also located here.

CVAUX namespace is described in OpenCV's documentation as containing obsolete and experimental code. However, the simplest interfaces for face recognition are in this module. The code behind them is specialized for face recognition, and they're widely used for that purpose.

ML namespace contains machine-learning interfaces. HighGUI namespace contains the basic I/O interfaces and multi-platform windowing capabilities.

CVCAM namespace contains interfaces for video access through DirectX on 32-bit Windows platforms

IV. PROPOSED SOLUTION

When image quality is taken into consideration there are factors that influence accuracy of the system .it is important to apply preprocessing techniQues to grasp the images.these images are sensitive to lightning conditions or if the person is wearing sunglasses sO the image would not consider into amount. therefore use processing filters before applying facial recognition .we can also remove pixels which are not in use.

So to avoid this problem we convert images into grayscale images and then apply histogram equalization, contour detection etc.

Open cv uses a face detector called Haar cascade classifier in this face detector examines each image location and classifies image as known face or not an works on each and every pixel.classifiers search the database stored image and match training set with the original image if match then ok otherwise repeat the process.

V. CONCLUSION

You can usually improve the face recognition accuracy by using more input images, at least 50 per person, by taking more photos of each person, particularly from different angles and lighting conditions. If you can't take more photos, there are several simple techniques you could training images, by generating new images from your existing ones:

You could create mirror copies of your facial images, so that you will have twice as many training images and it won't have a bias towards left or right.

You could translate or resize or rotate your facial images slightly to produce many alternative images for training, so that it will be less sensitive to exact conditions.

You could add image noise to have more training images that improve the tolerance to noise.

REFERENCES

[1] facaial recognition using open cv by Shervin EMAMI1, Valentin Petruţ SUCIU2
[2] Sophisticated image encryption using open cv by ashish pant
[3] face detection by Ming-Hsuan-Yang
[4] face recognition using principal component analysis