ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

Face Liveness and Disguise Detection Using Raspberry Pi and OpenCV

Piyush Devikar[1]

B.E. Student, Department of Electronics and Telecommunication Engineering, Vivekanand Education Society's Institute of Technology, Mumbai University, Mumbai, India[1]

ABSTRACT: Face liveness and disguise detection system eliminates the chances of a person to fake his/her identity. The face recognition systems available in the market fail to detect the fake faces shaped using high-end silicone masks and prosthetics. Also, these systems misinterpret face from a physical photograph as a real face. These are the vulnerabilities present in the available systems. This paper presents a simple approach to tackle the glaring vulnerabilities that are present in almost all face recognition systems. This system works on the principle that the surface temperatures of masks are close to ambient temperatures, unlike real faces whose temperatures are higher than ambient ones. This system captures image from the webcam connected to Raspberry Pi and then it is processed by OpenCV to detect the face in the image. The temperature of the face captured by the camera is obtained by IR temperature sensor. If the face is detected in the image and its temperature is more than the threshold value (skin temperature) then face is real otherwise, it is fake.

KEYWORDS: Face liveness, Disguise, Face recognition, Raspberry Pi, OpenCV

I. INTRODUCTION

Biometrics refers to technologies designed for the measurement and statistical analysis of people's physical and behavioral characteristics and has been widely used in authentication systems. These physical and behavioral characteristics include facial features, fingerprints, hand geometry, earlobe geometry, retina and iris patterns, voice waves, DNA, and signatures. The technology is mainly used for identification and access control, or for identifying individuals that are under surveillance. But there are vulnerabilities present in the available biometric systems[2][6].

Face Recognition systems can be spoofed by an identity thief, especially the ones based on face recognition, where the thief can obtain a photo of an authentic user from a significant distance, or even obtain it from the Internet. For instance, instead of showing one's own face to the biometric system, an unauthorized person can wear a mimic mask or display a photo of an authorized counterpart either printed on a piece paper, on a laptop, or even on a cell phone screen. The Face liveness and disguise detection system eliminate the chances of a person to fake his/her identity [1][5].

This proposed system works on the principle that the surface temperatures of masks and photographs are close to ambient temperatures, unlike real faces whose temperatures are higher than ambient ones. In this project, Raspberry Pi, Arduino, webcam and an Infrared(IR) Temperature sensor are used. Raspberry Pi makes the system portable and light. IR Temperature sensor is physically attached above the webcam so that to obtain the temperature values of the faces captured by the webcam. IR temperature sensor sends temperature values to Arduino. The captured images are processed by Raspberry Pi to recognize faces and Arduino serially sends temperature values of corresponding images to Raspberry Pi. The program for face detection is written in Python programming language that integrates OpenCV library. The temperature of the face is then compared with certain threshold temperature i.e. body temperature. If the temperature is more than the threshold one, then the face is real otherwise, it is fake.

DOI: 10.15680/IJIRCCE.2017. 0501018

II. RELATED WORK

Liveness is the act of differentiating the feature space into living and non-living. Imposters will try to introduce a large number of spoofed biometrics into the system. With the help of liveness detection, the performance of a biometric system will improve. It is an important and challenging issue which determines the trustworthiness of biometric system security against spoofing. In face recognition, the usual attack methods may be classified into several categories. The classification is based on what verification proof is provided to face verification system, such as a stolen photo, stolen face photos, recorded video, 3D face models with the abilities of blinking and lip moving, 3D face models with various expressions and so on. The Anti-spoof problem should be well solved before face recognition systems could be widely applied in our daily life [3].

A real-time and nonintrusive method based on the diffusion speed of a single image is proposed to address the problem of face spoofing using photographs or videos. In particular, the difference in surface properties between a live face and a fake one is efficiently revealed in the diffusion speed, we exploit antispoofing features by utilizing the total variation flow scheme. More specifically, defining the local patterns of the diffusion speed, the so-called local speed patterns, as the features, which are input into the linear SVM classifier is proposed to determine whether the given face is fake or not. One important advantage of the proposed method is that, in contrast to previous approaches, it accurately identifies diverse malicious attacks regardless of the medium of the image, e.g., paper or screen. Moreover, the proposed method does not require any specific user action. Experimental results on various data sets show that the proposed method is effective for face liveness detection [4].

III. SYSTEM OVERVIEW

The proposed system involves face liveness and disguise detection system that integrates an IR sensor, Arduino and Raspberry Pi. The Raspberry Pi processes the captured imageto detect face and the temperature data of that face to identify if the face is real or fake.

A. *RASPBERRY PI*

Raspberry Pi is a series of credit card-sized single-board computers developed in by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries. This is the perfect computer for this project due to its low cost, low power requirements, and active community. It can be plugged into a monitor or TV and uses a standard keyboard and mouse. It is a capable small device that enables people of all ages to explore computing and to learn how to program in languages like Scratch and Python. In this project, Raspberry Pi 2 Model B+ is used in which 1.2 GHz 32-bit quad-core ARM Cortex-A53 CPU is present along with 1 GB LPDDR2 RAM at 900 MHz [7].

B. *ARDUINO UNO*

Arduino Uno is a microcontroller board based on the ATmega328P microcontroller. It has 14 digital input/output pins of which 6 can be used as PWM outputs, 6 analog inputs, a 16 MHz quartz crystal, a USB port, a power jack, a reset button and an ICSP header. It contains everything needed to support the microcontroller. It can simply be connected to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started [8].

C. *MLX90614 TEMPERATURE SENSOR*

The Melexis MLX90614 is an infrared temperature sensor for non-contact temperature measurements. In this sensor, a low noise amplifier, 17-bit ADC and powerful DSP unit are integrated thus having high accuracy and resolution of the thermometer. The thermometer comes factory calibrated with a digital SMBus output giving full access to the measured temperature in the complete temperature range(s) with a resolution of 0.02°C. The user can also configure the digital output to be pulse width modulation (PWM). As a standard, the 10-bit PWM is configured to continuously transmit the measured temperature in a range of -20 to 120°C, with an output resolution of 0.14°C.

D. *OpenCV*

OpenCV i.e. Open Source Computer Vision Library is an open source computer vision and machine learning software library. OpenCV was made to provide a common infrastructure for computer vision applications and to accelerate the use of machine learning applications at the commercial level. OpenCV consists of more than 2500 optimized algorithms, which includes a wide set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces,identify objects, classify human actions in videos, track moving objects, extract 3D models of objects, stitch images together to produce a high-resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. It has C, C++, Java, Python and MATLAB interfaces and supports Windows, Linux, Android and Mac OS [9].

E. *PYTHON*

Python is an interactive, interpreted object-oriented programming language. It is a widely used high-level, general-purpose, dynamic, interpreted programming language.It incorporates modules, dynamic typing,exceptions, very high-level dynamic classes and data types. Python combines remarkable power with very clear syntax and hence it allows programmers to express logics in fewer lines of code than would be possible in languages such as C++ or Java. It has interfaces to many system calls and libraries, as well as to various window systems, and is extensible in C or C++. It can also be used as an extension language for applications that need a programmable interface. Python is portable as it runs on many Unix variants, on the Windows, and on Mac also.

IV. PROPOSED ALGORITHM

```
Step 1: Capture image from the Webcam
Step 2:  Detect Face in the Captured image
Step 3:  Get temperature array from IR temperature sensor
Step 4:  Calculate temperature of Face ($T_F$)
              if ($T_F$>Threshold Temperature)
   Face is Real.
             else
         Face is Fake
             end
Step 8: End.
```

DOI: 10.15680/IJIRCCE.2017. 0501018

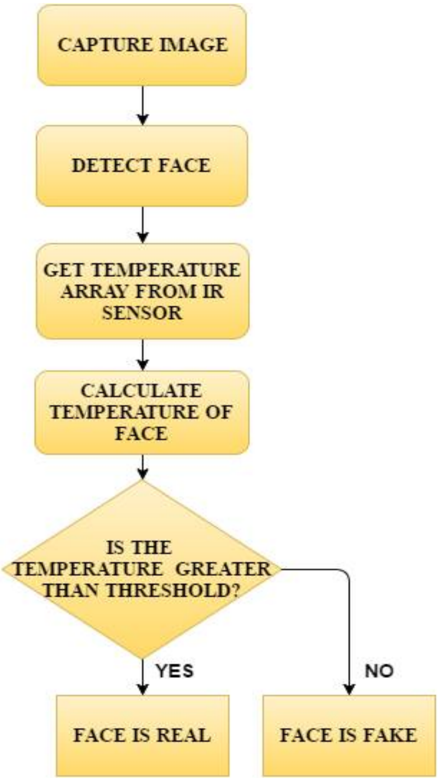
CAPTURE IMAGE
DETECT FACE
GET TEMPERATURE ARRAY FROM IR SENSOR
CALCULATE TEMPERATURE OF FACE
IS THE TEMPERATURE GREATER THAN THRESHOLD?
YES
NO
FACE IS REAL
FACE IS FAKE

Fig.1: Flow Chart of the Process

V. PROPOSED SYSTEM

The proposed system consists of Raspberry Pi, Arduino Uno, USB Webcam and an IR Temperature Sensor. In Fig.2, the connections of the whole system are shown. The overview of working of this system is shown in Fig. 1.

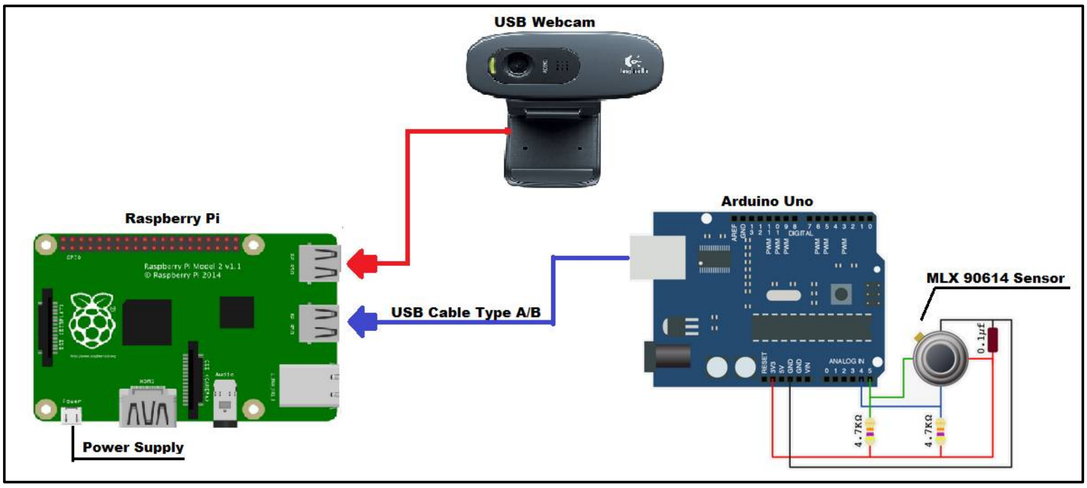

Fig. 2: System Overview

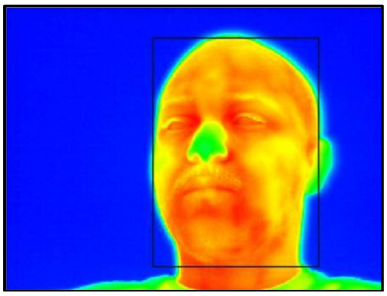

Fig.3: Human Face in Infrared Spectrum

A. *IMAGE CAPTURE*

In this process, webcam which is connected to Raspberry Pi captures the image. After capturing of the image, this image gets stored in the memory storage of Raspberry Pi. The Raspberry Pi then processes this image to detect faces using OpenCV. The program for face detection is written in Python programming language that integrates OpenCV library.

B. *FACE DETECTION*

The captured image from the previous step is processed by Raspberry Pi to detect the face. If the face is detected, Raspberry Pi fetches the temperature array of that face from IR temperature sensor via Arduino. The data transfer between Arduino and Raspberry Pi takes place by serial communication via USB A/B type cable.

C. *FACE TEMPERATURECALCULATION*

In this process, the temperature of the face is calculated from the temperature array values obtained from IR temperature sensor. From Fig.3, it is clear that the face temperature is always greater than the ambient one. So, the temperature of a photograph or a mimic mask will always be less than the skin temperature. This face temperature is then compared with a predefined threshold temperature value which is equal to the human skin temperature (36^0). If the face temperature to be compared is more than the threshold value, then the face is real otherwise, the face is fake and photograph or mimic mask may have used to bypass this layer of biometric security.

VI. SIMULATION AND RESULT

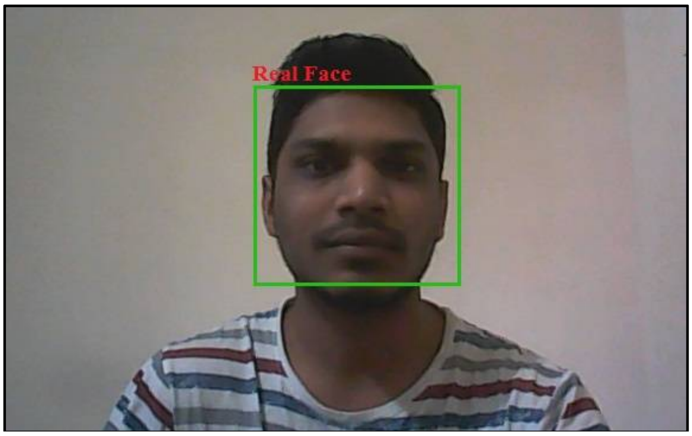

Fig .4: Real Face Detected by the system

In Fig.4, a real face is present in front of the webcam and the system detects this as real face. In this case, the temperature of the face is more than the threshold temperature and hence system detects it as a real face.

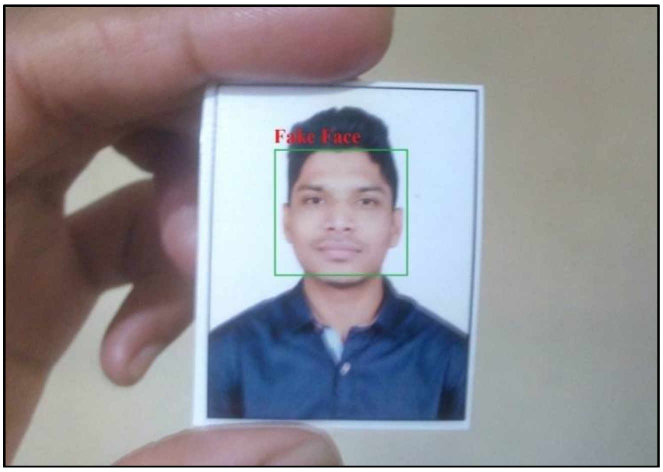

Fig. 5: Fake Face Detected by the System

In Fig.5, a photograph i.e. Fake face is placed in front of the camera and the system detects it as a fake face. In this case, the temperature of the face is less than the threshold temperature and hence system detects it as a fake face.

VII. CONCLUSION AND FUTURE WORK

The face recognition systems available in the market have certain vulnerabilities. They fail to detect the fake faces shaped using high-end silicone masks and prosthetics and from physical photographs also. This project proves to be highly efficient in overcomingthese vulnerabilities. The system is portable, light and has high processing speed.

The same system can be modified to detect people (or animals) with abnormally high or low body temperature. It can be used for detecting people with fever symptoms. Also, skin temperature may give insight into a computer user's emotional state. Therefore, it may be able to detect frustration levels for user experience testing or be integrated with online therapy systems.

REFERENCES

1. Bruno Peixoto, Carolina Michelassi, and Anderson Rocha, "Face Liveness Detection Under Bad Illumination Conditions", 18th IEEE International Conference on Image Processing, 2011.
2. Wonjun Kim, Member, IEEE, Sungjoo Suh, Member, IEEE, and Jae-Joon Han, Member, IEEE, "Face Liveness Detection from a Single Image via Diffusion Speed Model", IEEE Transactions on Image Processing, VOL. 24, NO. 8, AUGUST 2015.
3. Saptarshi Chakraborty and Dhrubajyoti Das, "An Overview of Face Liveness Detection", International Journal on Information Theory (IJIT), Vol.3, No.2, April 2014
4. Wonjun Kim, Sungjoo Suh, Jae-Joon Han, "Face Liveness Detection from a Single Image via Diffusion Speed Model", Volume: 24 Issue: 8
5. William Robson, Schwartz Anderson, Rocha HelioPedrini, "Face Spoofing Detection through Partial Least Squares and Low-Level Descriptors", Institute of Computing University of Campinas, Campinas, SP, Brazil, Av. Albert Einstein, 1251, 13083-852
6. Geetha, S., Phamila, AsnathVicty, "Combating Security Breaches and Criminal Activity in the Digital Sphere", June, 2016
7. https://www.raspberrypi.org/documentation
8. https://www.arduino.cc/en/tutorial
9. http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html

BIOGRAPHY

Piyush Devikar is currently pursuing B.E. in Electronics and Telecommunication Engineering at Vivekanand Education Society's Institute of Technology, Mumbai University, India. His research interests lie in the area of Embedded Systems, Image Processing and VLSI.

DOI: 10.15680/IJIRCCE.2017. 0501018