

# SISTEMA DE ANÁLISE FACIAL PARA CHAMADA ESCOLAR

*Victor Hugo B. Tavares, Wilton Miro Barros Júnior*

Programa de Graduação em Engenharia Eletrônica, Faculdade Gama  
Universidade de Brasília  
Gama, DF, Brasil  
email: victorhugo.tavares@hotmail.com, wiltonjrfla@gmail.com

## 1. INTRODUÇÃO

Um dos problemas que os professores enfrentam com os alunos é saber a frequência que os alunos participam das suas aulas, pois há várias maneiras de o aluno marcar sua presença mesmo não estando em sala de aula. Pensando nisso, teve-se a ideia de criar um sistema que realizasse a chamada de uma turma através de um reconhecimento facial. Na Universidade de Brasília, a frequência mínima do aluno em uma matéria é de 75% e sua chamada é realizada de forma de assinatura, no qual cada aluno assina sua presença em uma folha. Esse sistema pode ser facilmente fraudado, pois um aluno pode assinar para um ou mais colegas de turma. Sabendo disso, alguns professores recorrem a chamada oral para tentar diminuir esse fraude, mas há casos de quando a turma é muito grande, o professor pode não conhecer todos os seus alunos e outra pessoa responder por ele, além de esse sistema perder um tempo considerável da aula.

Desta forma, teve-se a ideia de se criar um sistema que possa evitar ou diminuir a inassiduidade dos alunos, pois evitando as fraudes, os alunos terão que ter a presença mínima para ser aprovado na disciplina além de garantir que o aluno está presente na aula e obrigatoriamente interagindo com a matéria.

## 2. OBJETIVO

O objetivo deste projeto é criar um sistema que diminua a inassiduidade dos alunos através de um reconhecimento facial e sensor de temperatura infravermelho. Visando aumentar a frequência dos alunos, uma menor probabilidade de evasão de alunos em uma instituição, além de ter um controle maior da instituição e dos pais [1].

## 3. REQUISITOS

Para a realização desse projeto, será construído um sistema inicialmente composto de um sensor de temperatura infravermelho, uma câmera e a Raspberry Pi B+. O sensor de

temperatura será usado como sistema de segurança para prevenir que se burle, com uma fotografia por exemplo, o sistema de detecção facial. Detectando a temperatura corporal validando assim que se trata de uma pessoa e não de uma fotografia. Já a câmera será usada para identificar o aluno cadastrado no banco de dados correspondente à sua respectiva sala de aula, e assim podendo validar sua presença. Todo o sistema será controlado pela Raspberry Pi.

## 4. BENEFÍCIOS

Com o emprego desse projeto será possível um maior controle de acesso dos alunos bem como a criação de um portfólio com esses respectivos dados que poderá ser usado para futuras análises. Podendo o sistema ser abrangido para diferentes setores correspondentes ao escopo do projeto.

## 5. DESENVOLVIMENTO

### 5.1. Descrição de Hardware

Para a realização deste projeto foi utilizada a seguinte lista de materiais:

Material	Quantidade
Raspberry Pi 3B+ (Micro-processador)	1
mlx90614esf (Sensor de temperatura)	1
Jumpers	4
Protoboard	1
Camera Raspberry Pi 5mp	1
Webcam (Uso temporário)	1
Estrutura do Sistema (A definir)	1

O Hardware consiste em basicamente em sistema composto por um sensor de temperatura infra-vermelho, uma câmera e a Raspberry Pi B+ (anexo 9.1). O sensor de temperatura será usado como sistema de segurança para prevenir que se burle, com uma fotografia por exemplo, o sistema de detecção facial, detectando a temperatura corporal e validando deste modo que se trata de uma pessoa e não de uma fotografia. Já a câmera será usada para identificar o aluno cadastrado no banco de dados correspondente à sua respectiva sala de aula, e assim podendo validar sua presença. Todo o sistema será controlado pela Raspberry Pi.

O termômetro infravermelho da MLX90614esf foi desenvolvido para sensoriamento de temperatura sem contato. Um conversor ADC de 17-bits e um poderoso DSP contribuem para que este sensor tenha uma alta precisão e resolução. O MLX90614 possui dois métodos de saída: PWM e SMBus (ou seja, TWI, I2C). A saída PWM de 10-bit possui uma resolução de 0,14°C, enquanto que a interface TWI possui uma resolução de 0,02°C. Este sensor sai de fábrica calibrado em uma grande faixa de temperatura: -40 a 85°C para ambientes e -70 a 382,2°C para temperaturas de objetos. O valor medido é a média de temperaturas de todos os objetos do campo de visão do sensor. O MLX90614 oferece uma precisão padrão de 0,5°C em temperatura ambiente.

O sensor de temperatura possui um VCC, GND e os pinos SDA e SCL. Esses pinos fazem parte do protocolo de comunicação do sensor, esse protocolo de comunicação é o I2C, neste protocolo precisa dos dois pinos já citados, o SDA significa serial data e esse pino transfere todos os dados obtidos pelo sensor e o SCL significa serial clock e serve para a temporização entre os dispositivos, de modo que a comunicação pela SDA possa ter confiabilidade. Sabendo que os pinos 1, 3, 5 e 6 da raspberry pi são respectivamente 3,3 volts, SDA, SCL e ground, foi realizado a montagem conforme circuito em anexo 1.

A câmera escolhida para o projeto, como mencionado no ponto de controle 2, foi o módulo câmera pi 5MP. Porém o módulo comprado veio com defeito e foi devolvido, e até a aquisição de um novo módulo, que já está em andamento, usaremos como alternativa para a caminhada do projeto uma webcam, que apesar de não ter a mesma qualidade de imagem do módulo que pretendemos usar, acreditamos que será suficiente para as futuras validações. A webcam está representada na imagem do anexo 3.

## 5.2. Descrição de Software

Nesse ponto de controle foi desenvolvido o código (anexo 9.2) para testar o funcionamento do sensor de temperatura medindo a temperatura corporal por meio da face em conjunto com a Raspberry pi. Para a câmera foi possível somente a instalação da biblioteca opencv, pois obtivemos a webcam um dia antes do ponto de controle e tivemos que fazer a instalação da biblioteca opencv por duas vezes, pois na pri-

meira vez, após seis horas e de instalação e compilação, deu erro na hora de instalar.

A principal biblioteca usada para o funcionamento do sensor de temperatura foi a BCM 2835. Esta é uma biblioteca C para o Raspberry Pi que fornece acesso ao GPIO e outras funções de in / out no chip Broadcom BCM 2835, usado no RaspberryPi, permitindo acesso aos pinos GPIO no plugue IDE de 26 pinos da placa RPi para poder controlar e fazer a interface com vários dispositivos externos. Ela fornece funções para leitura de entradas digitais e configuração de saídas digitais, usando SPI e I2C, para acessar os timers do sistema.

Para a execução do reconhecimento facial realizado com a câmera, foi realizada a instalação da biblioteca OpenCV (Open Source Computer Vision Library), que possui módulos de Processamento de Imagens e Video I/O, Estrutura de dados, Álgebra Linear, GUI (Interface Gráfica do Usuário) Básica com sistema de janelas independentes, Controle de mouse e teclado, além de mais de 350 algoritmos de Visão computacional como: Filtros de imagem, calibração de câmera, reconhecimento de objetos, análise estrutural e outros. O seu processamento é em tempo real de imagens. Esta biblioteca será importante para que possa realizar o reconhecimento facial.

## 6. RESULTADOS

Após algumas aferições com o sensor de temperatura apontado para a face, notou-se que a temperatura variava na faixa de 33 para 34 graus quando o sensor estava a um palmo de distância da face e essa temperatura diminuía a medida que o sensor se afastava, assim pensando na estrutura final do protótipo, isso será levado em consideração para que o sensor esteja aproximadamente nesta distância. Porém essa faixa de temperatura ainda não é a ideal já que a temperatura corporal normal varia entre 36,1 e 37,2[3], desta forma buscaremos uma forma de calibrar o sensor para que a temperatura se aproxime o mais próximo do valor real.

A nível de teste, considerou a temperatura corporal na faixa da qual o sensor mediu para criar uma condição no código para que indicasse quando o sensor medisse qualquer valor entre essa faixa de temperatura, imprimisse no terminal que era um humano, caso contrario, imprimia-se que não era um humano.

## 7. CONCLUSÃO

Para o sensor de temperatura, foi modificado o código, se comparado com o que foi apresentado no ponto de controle dois, para que a saída indicasse, a partir da temperatura, se era um humano ou não pois esse dado será uma das validações para que o aluno tenha sua presença confirmada na chamada. Mas há detalhes a se ajeitar como a aferição de

temperatura que não está de acordo com a temperatura corporal, sendo necessário uma forma de calibração.

Já com a câmera, além dos problemas com o módulo já citado no ponto de controle dois, foi possível ter a webcam apenas um dia antes do ponto de controle, desta forma não foi possível testá-la pois, além do período curto no qual a tivemos, tivemos problemas para instalar a biblioteca opencv porque o tutorial que foi seguido resultou na não instalação da biblioteca após seis horas de compilação e problemas de aquecimento da raspberry pi. Assim, todo nosso planejamento para este ponto de controle foi prejudicado.

Com todos esses fatores, sabe-se que estamos atrasados e esperamos no próximo ponto de controle mostrar o protótipo bem adiantado esperando no mínimo já termos as validações funcionando em conjunto.

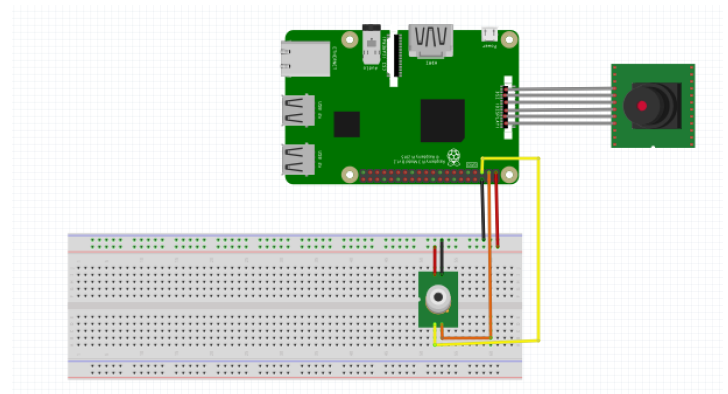
## 8. REFERENCIAS

- [1] <https://oglobo.globo.com/brasil/sistema-digital-avisara-pais-sobre-presenca-de-alunos-em-escola-do-interior-de-sp-3138190>
- [2] SHAH, SAMARTH (2015) *Learning Raspberry Pi*, 1ª edição, ISBN 9781783982820. Editora Packt Publishing Ltd.
- [3] <https://medicoresponde.com.br/qual-e-a-temperatura-normal-do-corpo-humano/>

## 9. ANEXOS

### 9.1. Anexo 1

Imagem do esquemático do circuito feito no Fritzing:



**Fig. 1.** Fluxo esquemático do sistema

### 9.2. Anexo 2

Código para o teste do sensor de temperatura:

```
#include <stdio.h>
#include <bcm2835.h>
#include <stdlib.h>
#include <fcntl.h>
#include <string.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <time.h>
#define AVG 1    //averaging samples

int main(int argc, char **argv)
{
    unsigned char buf[6];
    unsigned char i, reg;
    double temp=0, calc=0, skytemp, atemp;
    bcm2835_init();
    bcm2835_i2c_begin();
    bcm2835_i2c_set_baudrate(25000);
    // set address
    bcm2835_i2c_setSlaveAddress(0x5a);

    printf("\ndevice is working!!\n");

    calc=0;
    reg=7;

    for (i=0; i<AVG; i++)
```

```

{
bcm2835_i2c_begin();
bcm2835_i2c_write (&reg, 1);
bcm2835_i2c_read_register_rs(&reg,&buf[0],3);
temp = (double) (((buf[1]) << 8) + buf[0]);
temp = (temp * 0.02)-0.01;
temp = temp - 273.15;
calc+=temp;
sleep(1);
}

```

```
skytemp=calc/AVG;
```

```

if(skytemp > 33){

printf("É Humano\n");
}

else{

printf("Não é humano\n");
}

}

return 0;
}

```

### 9.3. Anexo 3

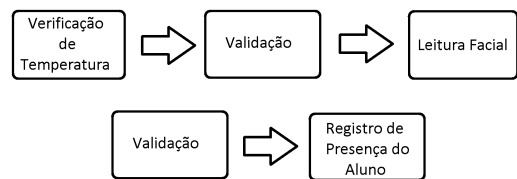
Imagem da webcam utilizada:



**Fig. 2.** Imagem webcam LG

### 9.4. Anexo 4

Fluxograma do sistema:



**Fig. 3.** Fluxo esquemático do sistema