

Ponto de Controle 4

Registrador de Presença por Detecção Facial

Leonardo Amorim de Araújo - 15/0039921 Josiane de Sousa Alves - 15/0038895

Email: leonardoaraujodf@gmail.com Email: josianealves.18@gmail.com

Universidade de Brasília

St. Leste Projeção A – Gama Leste, Brasília – DF, 72444 – 240

Resumo—Este documento apresenta o ponto de controle 4 para o projeto final da disciplina Sistemas Embarcados.

Keywords—Biometria, Detecção Facial, Raspberry PI, OpenCV

I. INTRODUÇÃO

As tecnologias de reconhecimento facial não são novidade nos dias atuais. Diversos dispositivos conseguem realizar trabalhos com processamento de imagens, um exemplo são os filtros digitais para fotos, a detecção de faces em redes sociais, celulares que utilizam a face como senha pessoal e etc. Além disso, estes aparelhos que realizam processamento de imagens estão bastante acessíveis. Apesar de ser um trabalho no geral trivial para seres humanos, o reconhecimento facial é uma tarefa desafiadora para computadores. Muitas são as aplicações possíveis para a detecção facial e aproveitando-se disso será construído o dispositivo para este projeto.

II. OBJETIVOS

Construir um sistema de chamada eletrônica, com registro de presença via detecção facial utilizando a Raspberry Pi .

III. DESCRIÇÃO

É alta a demanda hoje por dispositivos que realizam detecção facial, um exemplo bem atual encontra-se na referência [1]. Uma das grandes preocupações é com a proteção de dados pessoais e com a eliminação de fraudes, que costumam acontecer ainda com o uso de senhas pessoais e cartões. Além disso, tarefas que exijam controle manual de pessoas que entram e saem em um estabelecimento, ou presença no trabalho ou em sala de aula são boas alternativas para a aplicação de projetos que envolvam biometria facial para uma possível evolução em projetos mais sofisticados, como o uso pela polícia.[3] A biometria facial é, portanto, uma alternativa para estes problemas, e com base nisto que será proposto o projeto.

IV. REQUISITOS E PROTÓTIPO

Dentre os requisitos inseridos, visa-se:

- Obter o reconhecimento de todas as faces distintas cadastradas no banco de dados;
- Ter uma câmera com capacidade de detectar a face mesmo em ambientes com luz mais baixa;

- Conseguir um tempo máximo de 5 segundos para reconhecimento da imagem e registro da presença;
- Acrescentar a possibilidade do professor selecionar a turma em que a chamada será realizada para que não haja choque de dados;
- Implementar uma interface gráfica para que o aluno saiba que sua presença foi de fato registrada.

V. DESCRIÇÃO DE HARDWARE

- Raspberry Pi 3 Modelo B: Sendo o componente principal, é o computador, ou sistema embarcado, que realiza o processamento de imagens e a tomada de decisão no projeto;
- Câmera: Componente por meio do qual a Rpi verifica a presença do aluno em sala de aula, fazendo análise dos frames para tomada de decisão. Testes de identificação de face foram realizados com uma câmera de 1 MP relevando sucesso para esta escolha;
- Display: Componente que servirá de interface usuário-máquina, seu objetivo será possibilitar ao professor a escolha da turma em que a chamada será realizada (caso o professor tenha mais de uma turma), além de proporcionar ao aluno a confirmação da sua presença. Será utilizado um display LCD 16x2.
- Push buttons: Componente de seleção. Será utilizado em conjunto com o display para que o professor escolha em qual turma a chamada será realizada.

VI. DESCRIÇÃO DE SOFTWARE

Para a confecção deste projeto, está sendo utilizada a biblioteca OpenCV [4] - *Open Source Computer Vision Library*. Esta biblioteca foi desenvolvida pela Intel no início dos anos 2000 e é de uso acadêmico e comercial, para desenvolvimento de projetos na área de Visão Computacional. Possui módulos de processamento de imagens e vídeo, estrutura de dados, Álgebra Linear, GUI e muitos outros algoritmos. Inicialmente, foi desenvolvido/adaptado códigos na linguagem de programação Python para que se pudesse verificar se o projeto é factível. Já nesta etapa, a biblioteca OpenCV foi configurada na Rpi e alguns códigos em C++ foram implementados. O projeto está dividido em três partes principais, a fim de facilitar a implementação dos códigos. São elas: cadastro de nova turma, cadastro de novo aluno e detecção facial. A figuras 1,2 e 3 apresentam fluxogramas das partes citadas acima, onde

é possível verificar a ordem de execução que os códigos que serão desenvolvidos devem ter. Este fluxograma será explicado com mais detalhes logo a seguir.

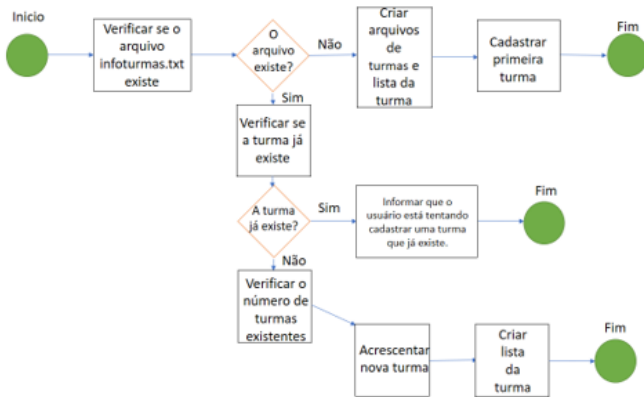


Figura 1. Fluxograma - Cadastro de nova turma

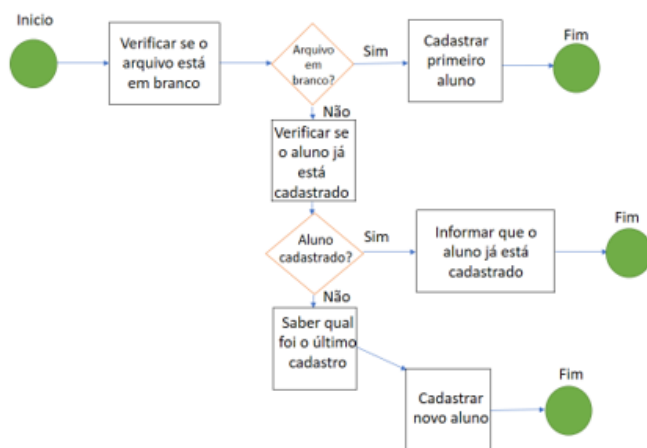


Figura 2. Fluxograma - Cadastro de novo aluno

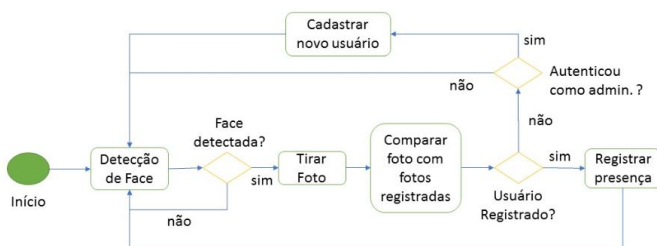


Figura 3. Fluxograma - Detecção facial

Abaixo são apresentados alguns códigos em C++ que serão utilizados para o projeto.

A. Take Picture

Este código tira fotos e guarda as mesmas em uma pasta específica. Ao mandar compilar o código, uma nova pasta

é criada dentro da pasta de uma turma, esta, será a pasta individual de cada usuário. Em seguida, ao pressionar a tecla *ESC* são tiradas várias fotos, que são utilizadas como amostras positivas. Este código faz parte da etapa de cadastro de novos usuários, e quanto mais fotos forem tiradas nessa etapa, melhor será para a detecção da face. Devido as limitações de projeto, especialmente com relação a velocidade com que as fotos são tiradas, determinou-se para este código, que 400 fotos sejam tiradas, o que leva aproximadamente 20 segundos.

B. Use Samples

Código que cria um arquivo.txt com o caminho para todas as amostras positivas e negativas; cria um vetor usando o *OpenCV* - *creatsamples* das amostras positivas, e treina um classificador usando o *OpenCV* - *train cascade* com o vetor criado e o caminho para as amostras negativas. A grande limitação para esta etapa do projeto, tem sido a dificuldade em se criar um classificador 100% eficaz, uma vez que não é possível para a realidade do projeto utilizar o número de amostras recomendado na literatura (entre 1000 e 10000). Dessa forma, ao compilar o código, ainda não foi possível detectar as faces que foram cadastradas no classificador.

C. Face Detection

Esta etapa vai utilizar os algoritmos de reconhecimento facial para identificar se a face lida corresponde a alguma face cadastrada no banco de dados. Esta etapa ainda não foi implementada. Porém a ideia é que, uma vez que a face foi reconhecida, uma comparação seja feita, de modo que se possa verificar a porcentagem de similaridade entre a face lida e as faces já cadastradas no banco de dados, computando a presença para o aluno, cuja face cadastrada tem com maior nível de similaridade com a face lida.

D. Registro da Presença

Nesta etapa, já que foi verificado que uma pessoa quer registrar sua presença e está cadastrada no banco de dados, o programa anterior irá criar ou abrir um novo arquivo para que seja computado o nome, matrícula e a hora da presença da pessoa, assim como a data. Após a presença ser registrada, o aluno será informado através do display que sua presença foi computada. Se possível, esses dados podem ser disponibilizados em uma planilha no *Google Drive*, através do método *sockets*, aprendido em sala de aula, conforme a disponibilidade.

E. Cadastro de Novo Usuário

Caso na etapa da verificação a foto não corresponda a nenhum dado no sistema, o usuário terá a opção de tirar uma nova foto ou cadastrar a foto capturada, como já citado acima. Caso a segunda opção seja escolhida, os dados do novo usuário deverão ser fornecidos, como nome, matrícula e cadastro da nova foto. O cadastro deverá ser autorizado pelo administrador do sistema (root) e então tiradas as 400 fotos para que o cadastro seja realizado.

F. Cadastro de Nova Turma

Uma nova turma pode ser cadastrada pelo administrador do sistema. Neste código, o administrador deve informar no nome da disciplina e a turma, e então uma pasta para esta nova turma será criada dentro do arquivo *infoturmas.txt*, que contém as informações de todas as turmas cadastradas. Dentro desta pasta, um arquivo chamado *listaturmax.txt*, com $x = 1, 2, 3 \dots$ será criado, este arquivo corresponde a lista de chamada da turma cadastrada.

G. Escolha da Turma

O professor poderá ter a opção de escolher qual turma este quer realizar a chamada, e um código para esta escolha deverá ser desenvolvido, de forma que se utilize push buttons e o display para a comunicação.

H. Conversão-Display

Como já citado acima, será utilizado um display LCD 16x2 e push buttons para a interface gráfica do projeto. Um código que recebe um caractere e retorna um número em hexadecimal a ser inserido na entrada de dados do display foi criado, para que assim o display possa ser utilizado.

I. Botões

Os push buttons fazem parte da interface gráfica do projeto. Para esta parte, um código que cria duas threads para verificar se os botões foram pressionados, assim o menu poderá ser acessado, para que a escolha da turma em que a chamada será realizada seja feita.

VII. RESULTADOS

Para este ponto de controle, os resultados obtidos foram:

- Um código para cadastro de nova turma;
- Um código para cadastro de novo usuário;
- Um código tirar fotos para realizar o cadastro de novo usuário;
- Um código para detecção da face;
- um código para a conversão dos caracteres do display em hexadecimal;
- Um código para detecção de quando os botões forem pressionados;
- Um código para classificação de amostras positivas e negativas de uma face.

Os códigos citados acima foram implementados em C++, conforme era esperado para este ponto de controle e funcionaram corretamente. Para esta fase final de projeto, espera-se solucionar o problema do reconhecimento facial e finalizar o projeto, fazendo a parte de interface usuário-máquina e integrando todos os códigos.

VIII. CONCLUSÃO

Nesta etapa, foram criados códigos em C++ e realizado a instalação da biblioteca OpenCV, de forma que estes atenderam em parte o que a proposta do projeto define, pois todos os códigos para os problemas especificados ainda não foram totalmente implementados. A dupla verificou com sucesso a possibilidade real da implementação do projeto na Rpi, e espera-se que para a apresentação final estes códigos estejam funcionando em conjunto, de forma que realizem a tarefa total especificada. Além disso, espera-se que uma interface usuário-máquina esteja funcionando corretamente, de forma que o produto final seja facilmente utilizado por professores e alunos. A dupla decidiu também adquirir, por conta própria, dois cursos, um de C++ e outro de OpenCV com implementações em C++, de forma que possa-se conseguir uma base teórica mais sólida para a confecção do projeto e para futuras implementações de projeto, sejam na carreira acadêmica ou profissional.

Tomando como base os conhecimentos adquiridos até o presente momento na disciplina de Sistemas Embarcados, bem como o que já foi feito do projeto, concluiu-se ainda que o projeto final está proporcionando à dupla conhecimento prático da disciplina, além de melhorias no domínio da linguagem C++ e nas ferramentas de processamento de imagens com a biblioteca OpenCV.

IX. ANEXOS

A. Códigos

Todos os códigos podem ser visualizados no link para o github a seguir:

https://github.com/leonardoaraujodf/Sistemas_Embarcados/tree/master/2_PCs/codigos_pc3

REFERÊNCIAS

- [1] Biometria facial começa a ser testada no transporte público do DF, acesso em 04/10/2017. http://www.correiobraziliense.com.br/app/noticia/cidades/2017/05/16/interna_cidadesdf,595226/biometria-comeca-a-ser-testada-no-distrito-federal.shtml
- [2] Aprenda a desbloquear seu notebook por reconhecimento facial, acesso em 04/10/2017. <http://www.techtudo.com.br/dicas-e-tutoriais/noticia/2015/06/aprenda-desbloquear-seu-notebook-por-reconhecimento-facial.html>
- [3] 100+ Projects in Image Processing and Fingerprint Recognition, acesso em 04/10/2017. <http://projectabstracts.com/list-of-projects-on-image-processing>
- [4] OpenCV library, acesso em 04/10/2017. <http://projectabstracts.com/list-of-projects-on-image-processing>