

Ponto de Controle 3

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 3 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. Ainda serão realizados testes para decidir qual display será utilizado para este propósito;
- 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. A figura 1 apresenta o fluxograma do projeto, 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. Abaixo serão apresentados alguns códigos em C++ que serão úteis para o projeto que a dupla realizou com o intuito de verificar se é possível a confecção do projeto.

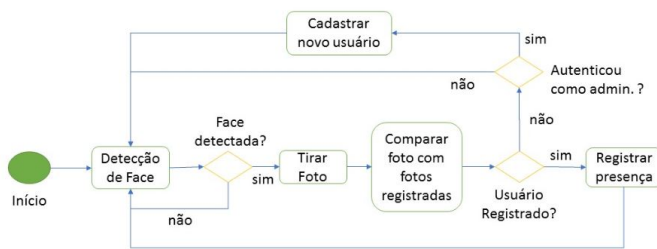


Figura 1. Fluxograma do Projeto

A. Detecção Facial

O código que realiza a detecção facial foi denominado *FaceDetection.cpp* e utiliza um algoritmo de detecção de objetos chamados de *cascade*. Quando se consegue detectar uma face, é desenhado um retângulo no local onde está a face da pessoa e consegue-se ainda verificar onde estão os olhos, com outro *cascade* de reconhecimento. A detecção facial será interessante para o projeto pois após a verificação de uma face com a câmera, será tirada uma foto, o que será explicado no tópico a seguir.

B. Tirar e Salvar Fotos

Foi criado também um código que consegue tirar fotos com a câmera principal do computador e salvá-las. Este código foi denominado *TakePicture.cpp*. No código, ao pressionar a tecla *ESC*, uma foto é tirada e logo após aparece na tela para que possa ser verificado como a foto ficou. Depois disto, pode-se pressionar qualquer tecla e o programa terminará. Apesar de ter utilizado o mecanismo de pressionar teclas para tirar fotos, a ideia é que este programa consiga se comunicar com o programa anterior de detecção facial, ou que seja implementado em um mesmo código, de modo que ao se detectar uma face, uma foto seja tirada. Uma vez que um rosto for detectado, uma foto será tirada automaticamente após um período (este período será somente para evitar falsos positivos, ou seja, falsas detecções de face). A foto capturada será salva (também automaticamente) no diretório padrão onde o programa estiver instalado. Com a imagem salva, será feita a verificação conforme explicado abaixo.

C. Verificação da Foto Tirada

Esta etapa vai utilizar os algoritmos de reconhecimento facial para identificar se a foto tirada na etapa anterior corresponde a alguma face cadastrada no banco de dados. Esta etapa ainda não foi realizada por completo, e por enquanto foi implementado um código que somente abre uma imagem, que foi denominado *DisplayImage.cpp*.

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).

F. 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. Este é um novo requisito de projeto e ainda será dimensionado para o ponto de controle 4.

VII. RESULTADOS

Para este ponto de controle, os resultados obtidos foram:

- Um código para abertura de imagens;
- Um código para reconhecimento facial;
- Um código para tirar uma foto quando uma tecla do teclado é pressionada.

Os códigos citados acima foram implementados em C++, conforme era esperado para este ponto de controle e funcionaram corretamente. Entretanto, alguns dos resultados esperados ainda não foram alcançados, como juntar alguns códigos de forma que realizem a tarefa especificada, como a detecção de face e a foto de amostra desta face, e a verificação desta com o banco de dados. Além disso, como foi proposto, o sistema deve ter uma comunicação usuário-máquina, o que deve ser discutido e implementado pela dupla até o próximo ponto de controle. Portanto, para o próximo ponto de controle, espera-se ter toda a parte funcional do projeto implementada, faltando apenas integrar os circuitos que o compõem (como em uma embalagem de produto), de forma que como resultado final, obtenha-se o projeto funcionando corretamente, e um produto visualmente atrativo.

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 até o próximo ponto de controle 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, o que será definido e implementado até o ponto de controle 4.

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>