



ISSN: 1984-3151

# VISÃO COMPUTACIONAL APLICADA EM RECONHECIMENTO FACIAL NA BUSCA POR PESSOAS DESAPARECIDAS

## APPLIED COMPUTER VISION IN FACIAL RECOGNITION IN THE MISSING PERSON SEARCH

Claudio Augusto de Paulo Resende<sup>1</sup>; Moisés Henrique Ramos Pereira<sup>2</sup>

1 Bacharel em Ciência da Computação. UniBH, 2015. Analista de sistemas da Bematech Rj Consultores. Belo Horizonte, MG. [claudioapr.particular@gmail.com](mailto:claudioapr.particular@gmail.com).

2 Mestre em Modelagem Matemática e Computacional. CEFET-MG, 2012. Professor do Centro Universitário de Belo Horizonte - UniBH. Belo Horizonte, MG. [moises.ramos@prof.unibh.br](mailto:moises.ramos@prof.unibh.br).

Recebido em: 20/09/2015 - Aprovado em: 23/11/2015 - Disponibilizado em: 30/11/2015

*RESUMO: Este artigo descreve o desenvolvimento de um sistema com a tecnologia de visão computacional utilizando ferramentas gratuitas, como a OpenCV, linguagem de programação Java, banco de dados MySQL etc. O sistema utiliza reconhecimento facial para localizar uma pessoa cadastrada em um banco de dados. O trabalho em questão faz um apanhado das tecnologias utilizadas como por exemplo algoritmos Eigenfaces e Viola Jones. Esses recursos possibilitaram a criação de um sistema que demonstrou eficiência no reconhecimento de pessoas desaparecidas para os testes que foi submetido. A biblioteca OpenCV utilizada no sistema em questão apresentou-se de forma bastante robusta no que tange documentação e funcionalidades.*

*PALAVRAS-CHAVE: OpenCV. Eigenfaces. Viola Jones.*

*ABSTRACT: ABSTRACT: This article describes the development of a system with computer vision technology using free tools such as OpenCV, Java, MySQL database etc. The system uses facial recognition to find a person registered in a database. The work in question makes a gathering of the technologies used such as Eigenfaces and Viola Jones. These features enabled the creation of a system that has shown efficiency in recognition of missing persons to the tests that he underwent. The OpenCV library used in the system in question presented is quite robust in relation documentation and features.*

*KEYWORDS: OpenCV. Eigenfaces. Viola Jones.*

### 1 INTRODUÇÃO

No Brasil, o índice de pessoas desaparecidas é bastante elevado. Segundo Zylberkan (2013), apenas no estado de São Paulo, desaparecem em média 57 pessoas por dia. De acordo com a estatística retirada do site do Jusbrasil (2013), pouco mais de 60% das pessoas que desaparecem retornam aos seus lares. Atualmente existem poucos recursos tecnológicos que auxiliam na busca dessas pessoas. Devido a essa

carência, os autores utilizaram a tecnologia de visão computacional para desenvolver uma ferramenta que auxilie nas buscas de desaparecidos.

O conceito de visão computacional, segundo Szeliski (2010), baseia-se em técnicas matemáticas para recuperar a forma tridimensional da imagem e a aparência de objeto. Aplicações utilizando visão computacional são encontradas em diversas áreas como robótica, medicina, física, biologia, indústria

entre outras. A robótica é um bom exemplo de aplicação da visão computacional, muitos robôs utilizam reconhecimento de objetos através de câmera para executar diversas tarefas. Dentro desse conceito utilizou-se para o trabalho principalmente o conceito de reconhecimento facial.

Muitas vezes principalmente em casos de doenças mentais, as pessoas desaparecidas não conseguem se lembrar de informações como, residência, nome, RG. Esse fator dificulta ainda mais que o desaparecido retorne ao seu lar. Utilizando reconhecimento facial pode-se resolver esse problema, pois uma foto pode servir como identificador.

O objetivo geral desse trabalho foi desenvolver um sistema que auxilie na localização de pessoas desaparecidas usando reconhecimento facial por meio de fotos. Os objetivos específicos foram:

- Alcançar conhecimento hábil na área de visão computacional.
- Demonstrar os conhecimentos obtidos na utilização da biblioteca *OpenCV*.
- Estudar e adequar o algoritmo de Viola Jones e *Eigenfaces* a fim de atender as necessidades do trabalho proposto.

Esse trabalho se justifica, pois, no cenário atual, existe uma carência de tecnologia de visão computacional aplicada em prol da busca por pessoas desaparecidas. Por ser um projeto acadêmico sem investimentos financeiros, os autores utilizaram de recursos tecnológicos de software livre, como por exemplo a biblioteca *OpenCV*, que é liberada sobe uma licença BSD e é uma biblioteca muito eficiente, atendendo perfeitamente as necessidades do trabalho proposto. Esse trabalho tem como motivação principal o intuito de auxiliar na busca de pessoas que se perderam por algum motivo, como: doença mental, perda de memória dentre outros a retornarem para

seus lares. A ideia é aumentar de forma significativa a estatística de pessoas desaparecidas que retornam ao seu âmbito familiar.

O presente documento está organizado da seguinte forma: A Seção 2 apresenta um apanhado de trabalhos relacionados com o tema, mas precisamente na área de visão computacional. A Seção 3 apresenta a teoria envolvida na área de visão computacional usada pelos autores. A Seção 4 apresenta os recursos utilizados para o desenvolvimento do trabalho proposto. A Seção 5 foi dedicada a experimentos e testes visando a demonstrar o funcionamento do sistema diante das necessidades reais. E por fim a Seção 6 apresenta a conclusão retirada de todo o trabalho, apresentando as impressões sobre os métodos utilizados e propondo outros caminhos, além de apresentar trabalhos futuros.

## 2 TRABALHOS RELACIONADOS

Com o objetivo de entender os problemas e as soluções das diversas abordagens existentes na área de visão computacional, mais especificamente no que se relaciona com reconhecimento facial, foram analisados alguns dos principais trabalhos da área a fim de retirar lições e experiências, bem como auxiliar na escolha da solução que mais se adequaria dentro das necessidades e limitações.

Entre os trabalhos analisados destacam-se, Gilvan (2010), Machado (2009), Diniz *et al.* (2013), Cárdenas, Vidal, Cavalcante e Vieira (2012) que promoveram avanços significativos na proposição de modelos para reconhecimento facial e de técnicas para buscas de pessoas desaparecidas.

O trabalho desenvolvido por Gilvan (2010), utiliza descritores locais, que são basicamente, de uma visão geral, vetores de características construídos com informações chamadas pontos de interesses ou pontos chaves. O reconhecimento facial utilizando

descritores é uma abordagem muito eficiente. O trabalho em questão, para o reconhecimento facial, utiliza-se do método proposto por Viola e Jones (2004) para detectar face que é representada por meio de descritores locais. Um ponto importante é que apenas pontos-chaves obtidos a partir do círculo que contém a face devem ser armazenados nos vetores (descritores locais), pois assim, além de diminuir o custo de processamento na obtenção de pontos-chaves, reduz a influência do fundo da imagem, o que diminui a possibilidade de falso positivo.

Cárdenas *et al.* (2012) apresentam um trabalho muito interessante, onde desenvolvem um sistema capaz de prever virtualmente como uma pessoa desaparecida estará após um tempo. Essa abordagem pode facilitar o trabalho de busca de pessoas que desapareceram por exemplo, quando criança. Conforme Cárdenas *et al.* (2012) apresenta em seu artigo, a técnica foi desenvolvida em nove passos:

- 1 - Seleção de fotografias da pessoa desaparecida e de seus pais na idade de referência;
- 2 - Construção dos modelos tridimensionais a partir das fotografias;
- 3 - Construção das Estruturas de Dados Genômicas (EDG) dos pais;
- 4 - Geração dos bancos de gametas (meiose);
- 5 - Geração da prole simulando o processo de fecundação entre gametas;
- 6 - Seleção do modelo da prole mais próximo ao modelo construído a partir de fotografias;
- 7 - Construção dos modelos paternos na idade objetivo;
- 8 - Atualização das EDG paternas;
- 9 - Atualização das EDG do filho selecionado a partir da informação paterna atualizada e construção do modelo facial tridimensional.

Segundo os autores mesmo com as restrições e problemas encontrados, o trabalho se apresenta de forma muito promissora.

Machado (2009), em seu trabalho desenvolve um sistema utilizando a linguagem C++ que efetua o cadastramento e o reconhecimento de faces. Seu sistema se utiliza do método *Eigenfaces* para efetuar o reconhecimento das faces nas imagens cadastradas. Esse método está contido na biblioteca OpenCV a qual Machado (2009) utilizou em seu trabalho. De acordo com Machado (2009), foram identificados problemas para o reconhecimento facial usando o *Eigenfaces*, pois o algoritmo em questão, quando utilizado, retornou muitos falsos positivos. Isso significa dizer que ao enviar o comando de reconhecimento facial para seu sistema, o algoritmo retornava positivo mesmo para faces não cadastradas na base de dados. Porém de acordo com o autor, o algoritmo se mostrou muito eficiente, atendendo às suas necessidades.

Assim como o trabalho de Machado (2009), o trabalho dos autores Diniz *et al.* (2013) utilizou-se do algoritmo *Eigenfaces* para efetuar o reconhecimento facial. Diniz *et al.* (2013) desenvolveram um sistema chamado *RedeFace* que efetua reconhecimento facial para validar o sistema, verificando seu desempenho e sua capacidade de reconhecimento. Os autores utilizaram uma base contendo 1280 imagens e identificaram que o desempenho do método, utilizando Análise de Componentes Principais (PCA, do inglês *Principal Component Analysis*), foi bastante satisfatório. Ainda de acordo com os autores, o sistema obteve uma taxa de reconhecimento acima de 90% de precisão.

Pela análise feita desses trabalhos e de outros não citados, optou-se por utilizar o método *Eigenface* neste trabalho por ser um método eficiente e que atenderia as necessidades do projeto proposto. O trabalho de Gilvan (2010) se mostrou bastante interessante e pelo fato de utilizar descritores locais, é

um sistema eficiente e flexível, porém apresenta um grau maior de dificuldade para sua implementação. O trabalho desenvolvido por Machado (2009) e Diniz *et al.* (2013) utilizam-se do mesmo método escolhido pelos autores. Apesar de o método apresentar algumas deficiências, como uma ocorrência elevada de falsos positivos, atende bem às necessidades do sistema proposto neste trabalho. Na seção de trabalhos futuros são abordadas algumas melhorias que os autores desejam implementar no projeto. Como diferencial o trabalho em questão propõe um sistema que ajude a resolver um problema de cunho social, que afeta diversas famílias. No que tange tecnologia o sistema traz um sistema inclusivo e que utiliza de recursos tecnológicos atuais, como mobile para facilitar a vida do usuário moderno.

### 3 FUNDAMENTAÇÃO TEÓRICA

Ao contrário dos seres humanos, que tem certa facilidade de perceber através do sentido da visão, características tais como cores, forma, emoções, o ambiente computacional, como um todo, não consegue reproduzir esse feito com tanta facilidade. Para alcançar esse objetivo diversos cientistas pelo mundo, trabalham com o intuito de aproximar o máximo possível o conceito de visão computacional do



Figura 1 – Principais etapas de um sistema de visão computacional.  
Fonte – MIRANDA, 2008.

sistema de visão humano. Na FIG. 1, tem-se uma visão geral das principais fases de um sistema típico de visão computacional.

#### 3.1 PROCESSAMENTO DE IMAGENS

De acordo com Szeliski (2010), a primeira etapa da visão computacional é o processamento de imagens, pois é possível através desse processo converter a imagem em uma forma adequada para que o meio computacional disponível, capaz de interpreta-las. Alguns exemplos de operações que acontecem nesse processo são: correção de exposição balanceamento de cores, redução de ruídos, aumento de nitidez e até mesmo reposicionamento da imagem, entre outros recursos que são de responsabilidade do processamento de imagem.

O processamento de imagens ocorre, geralmente, nos níveis baixo, médio e alto. No nível baixo, acontecem operações primitivas tais como redução de ruídos, melhorias de contrastes, dentre outras. No nível médio, acontecem operações tais como segmentação ou classificação. Já em alto nível ocorrem operações mais voltadas para a visão humana.

#### 3.2 MÉTODO VIOLA JONES

O método de Viola Jones é muito utilizado para detecção de faces em vídeos e imagens. De acordo com Ligneul (2011) o método utiliza como unidade básica *features* retangulares. Essas *features* apresentam formatos específicos e apresentam quatro padrões possíveis de configuração, que estão ilustrados na FIG. 2.

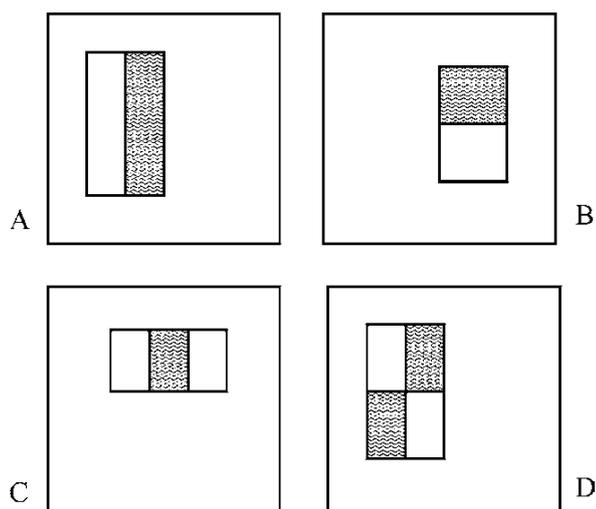


Figura 2 – Possíveis configuração de feature.  
Fonte – VIOLA; JONES, 2004, p.139.

A fórmula para o cálculo de uma *feature* é dada pela Eq. 1:

$$f(w) = \sum^w P_{\text{preto}} - \sum^w P_{\text{branco}}; \quad (1)$$

Em que:

$f(w)$ : valor do feature definido na janela  $w$ ;

$\sum^w P_{\text{preto}}$ : Somatório de pixels na região preta;

$\sum^w P_{\text{branco}}$ : Somatório de pixels na região branca.

Segundo Viola e Jones (2004), existem muitos motivos para utilizar *features* retangulares ao invés de pixel propriamente dito. O mais importante é que um sistema baseado em *features* retangulares é mais rápido que um sistema baseado em pixels, pois a quantidade de *features* é muito menor que a quantidade de pixels.

Ainda de acordo Viola e Jones (2004), as características de um retângulo podem ser computadas mais rapidamente usando uma representação intermediária para a imagem. Essa representação é dada pela integral da imagem, que é

definida pela Eq. 2, cuja representação visual está ilustrada na FIG. 3.

$$ii(x, y) = \sum_{x \leq x', y \leq y'} i(x', y'); \quad (2)$$

Em que:

$ii(x, y)$ : é a integral da imagem;

$i(x', y')$ : é a imagem original.

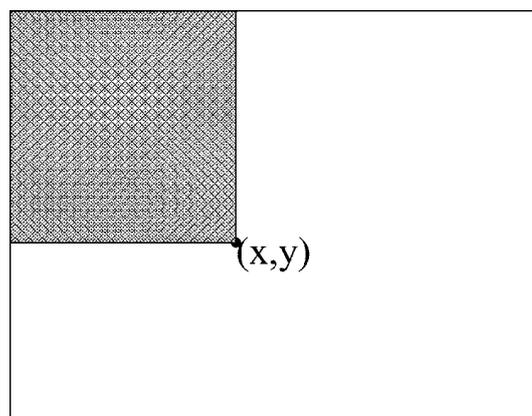


Figura 3 – Valor da integral da imagem no ponto  $(x, y)$  é a soma de todos os pixels acima e à esquerda.  
Fonte – VIOLA; JONES, 2004, p.139.

Usa-se para tanto, duas estruturas de recorrência. São elas:

$$s(x, y) = s(x, y - 1) + i(x, y); \quad (3)$$

$$ii(x, y) = ii(x - 1) + s(x, y). \quad (4)$$

Em que:

$s(x, y)$ : é a soma acumulada das linhas;

$s(x, -1)$  é igual a 0;

$ii(-1, y)$  é igual a 0;

Esta razão permite o cálculo da imagem integral apenas com uma passagem sobre a imagem original.

Feito a extração da integral da imagem a partir da imagem original, de acordo Viola e Jones (2004) é possível calcular o valor de qualquer área retangular, usando os quatro vértices da área que se deseja obter o valor. Tendo-se o valor total da área até os quatro vértices, basta realizar uma simples subtração de retângulos. Para exemplificar, a FIG. 4 ilustra as relações da área que se deseja calcular com as *features* retangulares, mostrando que para realizar o cálculo da região D, bastaria calcular  $4+1-(2+3)$ .

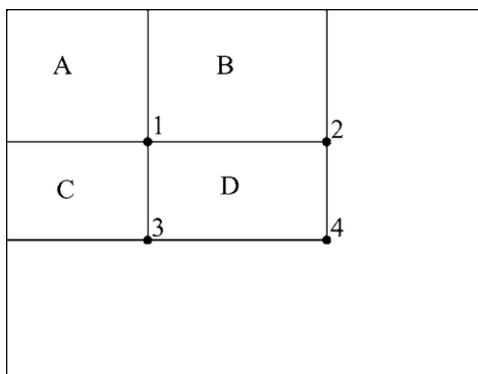


Figura 4 – Cálculo de uma região retangular de uma imagem integral.  
Fonte – VIOLA; JONES, 2004, p.140.

O método de Viola Jones utiliza uma variante do algoritmo AdaBoost para classificar um objeto em uma imagem. A ideia é que o classificador tenha um conjunto de características bem definidas, afim de avaliar regiões da imagem de forma mais precisa, reduzindo a margem de erro.

A FIG. 5 exemplifica o processo de treinamento do algoritmo. O classificador percorrer a imagem em busca de áreas com as mesmas características definidas no objeto desejado. A parte superior da imagem apresenta duas *features* retangulares, essas *features* são as características que o algoritmo AdaBoost selecionou durante o treinamento para identificar faces. Observa-se que o algoritmo identificou que a região dos olhos é mais escura que a região do nariz e bochecha.

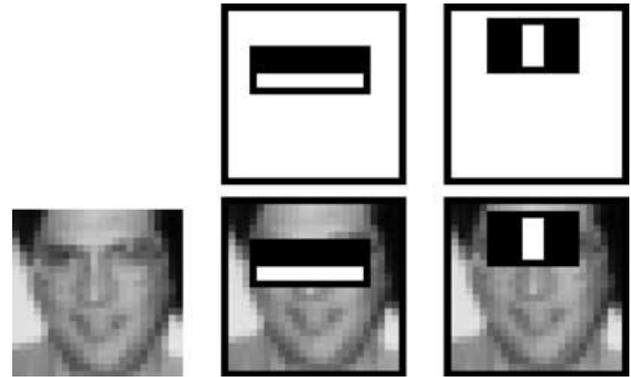


Figura 5 – Treinamento de reconhecimento de objeto AdaBoost.  
Fonte – VIOLA; JONES, 2004, p.144.

Para que o treinamento do algoritmo seja possível são necessários dois grupos de imagens denominados de acordo com Viola e Jones (2004) como imagens positivas e imagens negativas. As imagens positivas são as que representam o objeto de interesse e as imagens negativas são aquelas que não contêm o objeto de interesse.

Em suma o AdaBoost é um algoritmo que utiliza um classificador fraco, melhorando com isso o desempenho do algoritmo, que se mostrou muito eficiente, utilizado com intuito de treinar o classificador ou até mesmo recuperar características relevantes no objeto comparando as imagens do conjunto positivo com as do conjunto negativo.

### 3.3 EIGENFACES

Enquanto que para a detecção de face a biblioteca OpenCV implementa o algoritmo de Viola Jones, para o reconhecimento de faces a mesma utiliza o método *Eingefaces* desenvolvido por Turk e Pentland (1991).

De acordo com Turk e Pentland (1991), o método *Eingefaces* é baseado em uma abordagem da teoria da informação que tem como princípio a decomposição de imagens em pequenos conjuntos de sub imagens características. Para esse conjunto se dá o nome de *Eigenfaces* que são basicamente os

componentes que o algoritmo irá utilizar para fazer o treinamento das faces. Através dos *Eigenfaces* coletado, é gerada uma nova imagem (com as principais características obtidas na coleta de dados) e assim é comparada a imagem formada pelos *Eigenfaces* com a do indivíduo a ser reconhecido.

Conforme explica Turk e Pentland (1991) a abordagem utilizada pelo *Eigenfaces* tem como princípio extrair informações relevantes (não necessariamente olhos, nariz, lábios) de um rosto codificá-lo de forma mais eficiente (aplicando técnicas de processamento de imagem). Assim obtém-se um melhor resultado ao comparar a imagem em questão com uma outra, de uma base de dados que recebeu o mesmo tratamento. Ainda de acordo com publicação dos autores, o método funciona primeiramente com operações de inicialização. Os passos desta fase estão enumerados abaixo:

1. O primeiro passo é obter o conjunto de imagens para treinamento.
2. Segundo é extrair os *Eigenfaces* do *data set*, mantendo apenas as imagens com maior valor próprio ou seja que estejam no *Eigenvector* (vetor obtido a partir da matriz de covariância que por sua vez é obtida a partir da extração dos *Eigenfaces* obtido do *data set*).
3. O terceiro passo é o cálculo da imagem resultante dos passos acima de cada indivíduo, sendo que esse passo pode ser atualizado frequentemente a medida que novos dados (imagens) forem inseridos.

Após as imagens do *data set* terem sido processadas, executam-se os passos seguintes para efetuar o reconhecimento das faces:

- a. Calcula os *Eigenfaces* das imagens de entradas;
- b. Após o cálculo dos *Eigenfaces* verifica se a imagem resultante projetada é uma face;

- c. Se for uma face, efetua a classificação como pessoa conhecida (é uma face que está contida no *data set*) ou desconhecida.

### 3.4 RECUPERAÇÃO DE IMAGENS BASEADA EM CONTEÚDO

Com o crescimento da tecnologia nos últimos anos, principalmente com o desenvolvimento da internet e câmeras digitais, o armazenamento de imagens de forma digital está cada vez mais comum, com isso cada vez mais, existe uma necessidade de recuperar essas imagens de forma mais eficiente. Uma das áreas que estuda esse conceito é CBIR (*Content-Based Image Retrieval*) ou no português recuperação de imagens baseada em conteúdo.

De acordo Müller *et al.* (2009) a CBIR se tornou uma das áreas mais estudadas no campo de visão computacional nos últimos 10 anos. Em CBIR as imagens de uma base são associadas a outras informações. Essas imagens podem ser representadas por diversas forma como por exemplo: característica de pixel, objetos, *features* e etc. A similaridade de duas imagens por intermédio da recuperação baseada em conteúdo é calculado por uma função que calcula a distância entre as respectivas características de cada imagem. À essa função, que extrai características e calcula similaridades, pode-se chamar descritores. De acordo com Torres e Falcão (2006),

“um descritor visa agrupar imagens com características similares em uma dada região do espaço de medida (características), e grupos de imagens com características distintas em regiões diferentes deste espaço. Este aspecto tem sido explorado em estruturas de indexação para armazenar imagens similares em regiões próximas no disco do computador, e assim tornar mais eficiente a consulta.” (TORRES; FALCÃO, 2006).

O sistema proposto utiliza reconhecimento de padrões para consultar se uma determinada imagem pertence a um determinado grupo de dados imagéticos. Caso

essa imagem pertença a algum grupo, o mesmo é retornado para o banco de dados, onde é feita uma consulta e exibido na tela o indivíduo encontrado a partir da consulta feita pelo usuário.

## 4 METODOLOGIA

A pesquisa realizada neste trabalho pode ser classificada como descritiva. Isto porque a pesquisa desenvolvida irá descrever o desenvolvimento de uma aplicação que trabalha com o conceito de visão computacional. Esse conceito tem o propósito de ajudar a sociedade com um problema que vem crescendo com o passar dos anos, - o desaparecimento de pessoas -, bem como abordar os fatores sociológico e tecnológico.

Quanto à metodologia, o trabalho faz a opção pelo método dedutivo. Esta opção se justifica porque o método escolhido permite que durante o trabalho sejam apresentadas ideias correlatas, provando as premissas e as indagações feitas na pesquisa.

Quanto ao procedimento, este trabalho realizar-se-á por meio de observação indireta, tendo em vista que foram coletadas informações de livros e trabalhos relacionados. A pesquisa utilizar-se-á de revisões bibliográficas como ferramenta de elaboração a fim de acrescentar base de conhecimento suficiente, para abordar com propriedade o tema sugerido, proporcionando obter informações e comparativos das bibliografias analisadas e o trabalho em mãos.

### 4.1 FERRAMENTAS UTILIZADAS

O desenvolvimento do trabalho proposto só é possível graças ao estudo da visão computacional que tem como princípio tentar simular o máximo possível o sistema de visão humano. Com o intuito de facilitar o processo de desenvolvimento de aplicação que

utilizasse a tecnologia de visão computacional, a Intel Pesquisas desenvolveu a biblioteca *OpenCV*.

Para a detecção de faces a *OpenCV* implementa o método criado por Viola e Jones em 2004. Esse método faz uso de funções matemáticas combinadas com visão computacional e possibilita o reconhecimento facial em tempo real de forma eficiente. Esse algoritmo proposto por Viola e Jones reescreve o algoritmo de inteligência artificial que aplica os conceitos de aprendizagem de máquina, Adaboost, a fim de classificar as imagens e conseguir efetuar o reconhecimento.

Para o reconhecimento de faces, a *OpenCV* implementa o método *Eigenfaces* desenvolvido por Turk e Pentland. Esse método obtém, a partir da imagem, outras sub imagens que são chamadas de *Eigenfaces*, que são compostas pelas principais características da imagem original a exemplo da boca, nariz e lábios.

O trabalho desenvolvido utilizará a linguagem de programação *Java Standard Edition (SE)*, sendo empregada para desenvolvimento de uma interface com usuário *desktop* e *mobile*. Para armazenamento de dados será utilizado o *MySQL*, sistema de gerenciamento de banco de dados (SGBD) gratuito e que apresenta total integração com as ferramentas que serão utilizadas no trabalho proposto.

### 4.2 DESENVOLVIMENTO DO SISTEMA

O sistema, como falado anteriormente, foi desenvolvido na linguagem *Java*. Trata-se uma linguagem de programação e plataforma computacional lançada pela primeira vez pela Sun Microsystems em 1995, mas que atualmente é mantida pela Oracle. De acordo com informações retiradas do site da linguagem, [https://java.com/pt\\_BR](https://java.com/pt_BR), a linguagem *Java* tem atualmente mais de 9 milhões

de desenvolvedores por todo mundo. Seguem algumas estatísticas retiradas do site da linguagem:

- 97% dos Desktops Corporativos executam o Java;
- 89% dos Desktops (ou Computadores) nos EUA Executam Java;
- 9 Milhões de Desenvolvedores de Java em Todo o Mundo;
- A Escolha Nº 1 para os Desenvolvedores;
- Plataforma de Desenvolvimento Nº 1;
- 3 Bilhões de Telefones Celulares Executam o Java;
- 100% dos Blu-ray Disc Players Vêm Equipados com o Java;
- 5 bilhões de Placas Java em uso;
- 125 milhões de aparelhos de TV executam o Java;
- 5 dos 5 Principais Fabricantes de Equipamento Original Utilizam o Java ME.

Para armazenamento das informações utilizou-se o banco de dados MySQL, ferramenta gratuita, que também é mantido pela Oracle.

De acordo com o site <http://www.oracle.com>, o MySQL é o banco de dados de código aberto mais popular do mundo.

Abaixo será explicado a respeito das funcionalidades do sistema.

As buscas poderão ser feitas por nome e por região, mas o grande diferencial da aplicação será no uso do conceito de visão computacional para recuperar informações sobre as pessoas desaparecidas cadastradas no banco de dados. Além de efetuar pesquisas pelas opções já mencionadas, o usuário que estiver realizando as buscas das pessoas na base de dados, poderá também realizar o *upload* de uma

foto. Essa será lida pelo *software* e as características das imagens serão comparadas com as características das imagens cadastradas no banco de dados. Caso encontre alguma pessoa com o crivo feito no código da aplicação, o mesmo retornará o individuo desaparecido que está cadastrado.

Para desenvolver esta aplicação utilizando técnicas de visão computacional será utilizada a biblioteca OpenCV. Esta é uma biblioteca *open source* que é escrita na linguagem C e atualmente roda em diversas linguagens como C++, Python, Java entre outras. De acordo com Bradski e Kaebler (2008), a *OpenCV* foi projetada para ter uma grande eficiência computacional e com foco em tempo real, por isso é uma ferramenta bastante utilizada para processamento de imagem.

## 5 RESULTADOS EXPERIMENTAIS

Conforme o objetivo inicial e demonstrado nas seções anteriores, o trabalho teve como fruto um sistema desenvolvido na linguagem de programação Java, o mesmo segue o fluxo de funcionamento de acordo com a FIG. 6.

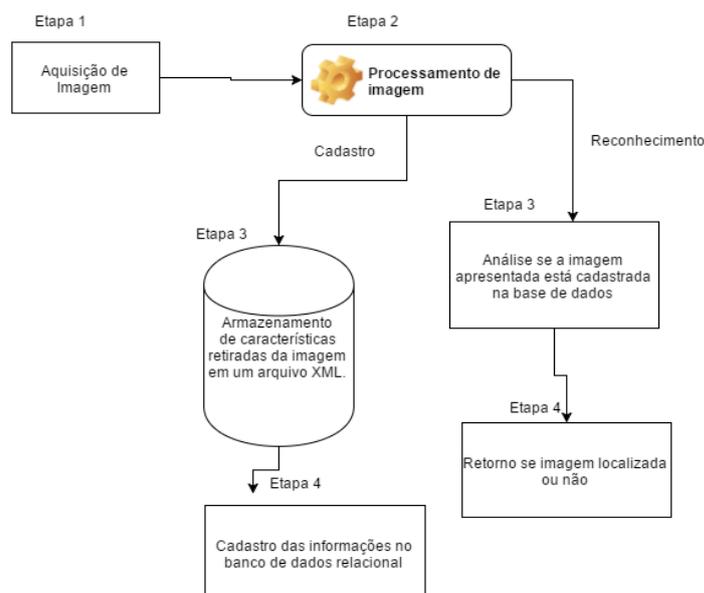


Figura 6 – Fluxo de funcionamento do sistema

Para validação do sistema quanto ao reconhecimento facial, foram feitos alguns testes para medir informações como: tempo de processamento e precisão de reconhecimento. Os testes foram realizados utilizando o seguinte cenário tecnológico: Notebook HP com processador Intel I5, 4GB de memória RAM, 500GB de HD e sistema operacional Windows 8.1. Já para o treino utilizando o método *Eigenfaces*, utilizou-se um *data set* composto por 15 pessoas, cada uma sendo cadastrada com 7 imagens. Para o reconhecimento facial, utilizaram-se imagens coletadas de forma aleatória retiradas de uma rede social, com poses aleatórias e iluminações quaisquer, acreditando-se que essa pequena amostra de 15 pessoas é independente. A TAB. 1 a seguir demonstra o tempo gasto pelo sistema para treinar cada pessoa cadastrada:

Tabela 1

Tempo gasto para treinar as imagens de cada pessoa

Treinamento	
Pessoa	Tempo (s)
Ana Caroline	3
Ananias	4
Carolina	5
Claudio	5
Jacira	5
Larissa	5
Maria Clara	4
Maria Consolação	4
Maria Paula	4
Maria Socorro	4
Mario	3
Messias	5
Mike	5
Mikel	4
Wallace	4
Média em segundos	4,266667

O sistema levou em média 4,3 segundos para efetuar o treinamento de cada pessoa. Já para o

reconhecimento, a TAB. 2 apresenta o tempo gasto para o sistema reconhecer ou não a face de entrada.

Tabela 2

Tempo de treinamento das imagens

Reconhecimento	
Pessoa	Tempo (s)
Ana Caroline	40
Ananias	60
Carolina	46
Claudio	49
Jacira	34
Larissa	70
Maria Clara	53
Maria Consolação	46
Maria Paula	58
Maria Socorro	50
Mario	35
Messias	54
Mike	35
Mikel	34
Wallace	33
Média em segundos	46,66666667

A média de tempo para reconhecer ou não a face está em torno de 40,7 segundo, um valor considerado alto pelos autores. Em função dos resultados, pretende-se trabalhar com outras técnicas posteriormente com intuito de otimizar o processo de treinamento e reconhecimento. Outro aspecto importante é que tanto para treinamento quanto para o reconhecimento o sistema executa outros algoritmos quando prepara a imagem para obter um melhor resultado, o que conta bastante no desempenho da aplicação. No quesito precisão no reconhecimento, obteve-se uma porcentagem de 86,7% de faces reconhecidas, um valor considerado razoavelmente bom, pois o *data set* utilizado continha imagens aleatórias, com intuito de simular ao máximo um ambiente de produção. A fim de acurar o sistema, utilizaram-se as métricas de precisão e revocação aplicadas a partir de uma base de dados conhecida previamente. São utilizados 2

(dois) parâmetros: os conjuntos de dados existentes que são relevantes e o conjunto de dados relevantes recuperados pelo sistema.

Conforme Cardoso (2000), a fórmula para o cálculo de precisão é dada pela razão entre os dados relevantes recuperados pelo sistema e o total de dados recuperados no momento da consulta, como é demonstrado na Eq. 5:

$$P = \frac{T}{N} \quad (5)$$

Em que:

*P*: Precisão;

*T*: Itens(faces)relevantes recuperados pelo sistema;

*N*: Total de itens(faces) recuperados pelo sistema;

Para a métrica de precisão o sistema apresentou o desempenho conforme TAB. 3:

Tabela 3  
Cálculo da métrica de precisão

Precisão			
Pessoa	Itens recuperados	Itens relevantes	Precisão
Ana Caroline	12	7	0,583333
Ananias	13	6	0,461538
Carolina	13	7	0,538462
Claudio	12	7	0,583333
Jacira	14	7	0,5
Larissa	34	0	0
Maria Clara	9	5	0,555556
Maria			
Consolação	8	5	0,625
Maria Paula	8	5	0,625
Maria Socorro	12	6	0,5
Mario	12	5	0,416667
Messias	11	5	0,454545
Mike	23	4	0,173913
Mikel	40	0	0
Wallace	21	4	0,190476
Precisão média			0,413855

Já o cálculo para a métrica de revocação é dada pela razão entre os itens relevantes para consulta devidamente cadastrados e os itens relevantes retornados pelo serviço de busca de faces do sistema implementado. A Eq. 6 descreve essa razão:

$$R = \frac{I}{T} \quad (6)$$

Em que:

*R*: Revocação;

*I*: Itens(faces)relevantes do sistema para a busca;

*T*: Itens(faces)relevantes recuperados pelo sistema;

A TAB. 4 demonstra os resultados apresentados pelo sistema quando calculada a métrica de revocação sobre os resultados de busca.

Tabela 4  
Cálculo da métrica de revocação

Revocação			
Pessoa	Itens	Itens relevantes	Revocação
Ana Caroline	7	7	1
Ananias	7	6	0,857142857
Carolina	7	7	1
Claudio	7	7	1
Jacira	7	7	1
Larissa	7	0	0
Maria Clara	7	5	0,714285714
Maria			
Consolação	7	5	0,714285714
Maria Paula	7	5	0,714285714
Maria Socorro	7	6	0,857142857
Mario	7	5	0,714285714
Messias	7	5	0,714285714
Mike	7	4	0,571428571
Mikel	7	0	0
Wallace	7	4	0,571428571
Revocação média			0,695238095

## 6 CONSIDERAÇÕES FINAIS

A biblioteca OpenCV mostrou-se bastante versátil e eficiente para resolver diversos problemas da área de visão computacional. As ferramentas disponíveis na *Application Programming Interface* (API) para processamento de imagens, detecção de face e reconhecimento facial atenderam às necessidades iniciais do sistema proposto pelos autores. Outro ponto positivo no uso da OpenCV foi a disponibilidade de documentação robusta e com boa qualidade, abordando as diversas funcionalidades que a mesma disponibiliza.

Para o reconhecimento facial utilizou-se uma das abordagens da OpenCV, chamada *Eigenfaces*. Como demonstrado na seção anterior apresentou resultados aceitáveis para um primeiro momento, no entanto os autores acreditam ser necessário trabalhar o quesito tempo gasto pelo sistema para treinamento e para reconhecimento. A taxa de reconhecimento do sistema proposto girou em torno 86,7% como pode-se observar na seção de resultados e experimentos. Esse valor é considerando bom pelos autores, pois, foram utilizadas imagens aleatórias e muitas se apresentavam com filtros ou com posicionamento da face que dificultaram o processo de reconhecimento.

Apesar das limitações e problemas apresentados, os autores consideraram promissor o projeto. Os valores apresentados pelo sistema quando calculadas as métricas de revocação e precisão são considerados muito bons apesar de serem aplicado em uma base de imagem relativamente pequena.

Acredita-se que uma abordagem utilizando descritores locais seja um caminho interessante a seguir. Nas bibliografias analisadas, os autores que citam esse tipo de abordagem definem como uma solução muito eficiente e precisa para o reconhecimento de face. Outra alternativa é a utilização de redes neurais para efetuar o reconhecimento facial. Parece ser uma alternativa mais precisa e mais complexa, porém com

um resultado final possivelmente melhor que as outras abordagens.

O processamento de imagens é algo muito custoso, exigindo cada vez mais um sistema tecnológico (*hardware* e *software*) mais robusto. Com intuito de resolver problemas de processamento pesados como o de imagem estão sendo desenvolvidas algumas técnicas como a de processamento paralelo. Segundo Hwang e Briggs (1984) o Processamento Paralelo é uma forma de processar informações explorando eventos concorrentes no processo computacional. O Processamento Paralelo permite dividir um grande processamento em vários outros menores que podem assim serem processados de forma simultânea pelo processador. A ideia é utilizar desse conceito para processar imagens de forma mais rápida e eficiente, obtendo um ganho em performance para aplicar o reconhecimento facial em câmeras instaladas em locais públicos a fim de identificar transeuntes desaparecidos cadastrados na base de dados do sistema.

Os autores pretendem também reescrever o sistema com intuito de viabilizar o cadastro de pessoas desaparecidas por intermédio de tecnologias mobile, pois atualmente somente a identificação de pessoas no banco de dados é feita utilizando essa tecnologia.

## AGRADECIMENTOS

Os autores agradecem suas respectivas famílias, pelo apoio incondicional, auxílio, força e paciência demonstrada durante todo o curso. Os Autores agradecem ainda aos professores do curso de Ciência da Computação; ao Walisson Ferreira de Carvalho que coorientou esse trabalho, prestando auxílio de forma efetiva e memorável; à Magali Maria de Araújo Barroso e à Maira Moreira Morais que também coorientou esse trabalho com muita responsabilidade e dedicação.

## REFERÊNCIAS

BRADSKI, G; KAEHLER, A. **Computer Vison with the OpenCV Library**. 1 ed. USA: O'REILLY, 2008. Pg.1-56.

CÁRDENAS-ESGUERRA, M.; *et al.* **Facial Aging Simulation Applied to the Missing Person Problem** - XXXVIII Conferencia Latinoamericana en Informatica (CLEI), IEEE, 2012, p. 1-9.

CARDOSO, O. Recuperação de Informação. INFOCOMP - **Journal of Computer Science**, vol. 2, n. 1, 2000.

DINIZ, F. A. *et al.* RedFace: um sistema de reconhecimento facial baseado em técnicas de análise de componentes principais e autofaces: comparação com diferentes classificadores. **Revista Brasileira de Computação Aplicada**, vol. 5, n. 1, 2013.

GILVAN, J. **Deteção e Reconhecimento de Objetos usando Descritores Locais**. Teste (Doutorado em Ciência da Computação). Universidade Federal do Ceará (UFC): Fortaleza, 2010.

HWANG, K.; BRIGGS, F.A., **Computer Architecture and Parallel Processing**, McGraw-Hill, 1984. 1-9 pg.

JAVA, **Obtenha Informações sobre a Tecnologia Java**. Disponível em <[https://java.com/pt\\_BR/about](https://java.com/pt_BR/about)>. Acessado: 21 Junho 2015.

JUSBRASIL. **Mais de 60% das pessoas que somem reaparecem, mas estatísticas não são atualizadas**. Cidade: Rio de Janeiro, Portal jusbrasil, 2013. Disponível em <<http://oab-rj.jusbrasil.com.br/noticias/100676104/mais-de-60-das-pessoas-que-somem-reaparecem-mas-estatisticas-nao-sao-atualizadas>>. Acessado em: 21 sets 2014.

LIGNEUL, T. S. **Deteção de faces através do algoritmo de Viola-Jones**. Disponível em <[http://www.lcg.ufrj.br/Cursos/cos756/trabalhos-2011/trabalhos-2011/Tulio%20Ligneul%20-](http://www.lcg.ufrj.br/Cursos/cos756/trabalhos-2011/trabalhos-2011/Tulio%20Ligneul%20-%20Detector%20de%20Faces%20Viola-Jones%20-%20Relatorio.pdf)

[%20Detector%20de%20Faces%20Viola-Jones%20-%20Relatorio.pdf](http://www.lcg.ufrj.br/Cursos/cos756/trabalhos-2011/trabalhos-2011/Tulio%20Ligneul%20-%20Detector%20de%20Faces%20Viola-Jones%20-%20Relatorio.pdf)>. Acessado em 20 out 2014.

MACHADO, B. *et al.* **Implementação de um algoritmo de Reconhecimento facial usando Eigenfaces**. Colóquio Científico no Uni-BH: a Computação é Dez! Revista e-xacta, vol. 2, n. 2, 2009.

MIRANDA, J. J. **MOVE-IN: Uma API para Interatividade com a Webcam**. Monografia (Bacharelado em Ciência da Computação). Universidade do Estado de Santa Catarina (UDESC): Joinville, 2008.

MÜLLER, H. *et al.* A review of content-based image retrieval systems in medical applications - clinical benefits and future directions. **International Journal of Medical Informatics**, 2009.

TURK, M; PENTLAND, A. Eigenfaces for Recognition, **Journal of Cognitive Neuroscience**, 1991. Pg.1-9.

ORACLE, **Visão geral**, Disponível em <<http://www.oracle.com/br/products/mysql/overview/index.html>>. Acessado: 21 junho 2015

SZELISKI, R. **Computer Vision: Algorithms and Applications**. , 2010. Pg.1-100

TORRES, R; FALCÃO, A. Content-Based Image Retrieval: Theory and Applications. **Revista de Informática Teórica e Aplicada**, 2006, pg 161-185.

VIOLA, P; JONES, M. Robust Real-Time Face Detection, **International Journal of Computer Vision**, 2004. pg. 1-9.

ZYLBERKAN, M. **57 pessoas desaparecem por dia em São Paulo**. Cidade: São Paulo, Revista Veja, 2013. Disponível em <<http://veja.abril.com.br/noticia/brasil/57-pessoas-desaparecem-por-dia-em-sp>>. Acessado: 21 sets 2014.