

Universidade Federal do ABC  
Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas  
Trabalho de Graduação em Engenharia de Informação

# **Aplicação de Aprendizado de Máquina para Detecção Facial (REVER)**

**Gabriel Pitalli de Carvalho**

**Santo André  
Maio de 2020**



# Sumário

	<b>Introdução</b>	<b>3</b>
<b>0.1</b>	<b>Objetivos e Motivação</b>	<b>3</b>
<b>1</b>	<b>MATERIAIS E MÉTODOS</b>	<b>5</b>
<b>1.1</b>	<b>OpenCV</b>	<b>5</b>
<b>1.2</b>	<b>Algoritmo Viola-Jones</b>	<b>5</b>
1.2.1	Imagem Integral	6
1.2.2	Algoritmo de Impulsão AdaBoost	7
1.2.3	Classificador em Cascata	8
<b>2</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>9</b>
	<b>REFERÊNCIAS</b>	<b>11</b>



# Introdução

Reconhecimento facial é uma tarefa trivial para humanos e a décadas tem sido um desafio para visão computacional e aprendizado de máquina, segundo a referência 1, desde os anos 90 o tema emerge em diferentes conferencias e com o aumento do poder computacional dos dias atuais, sua capacidade se expande muito, fazendo com que tal assunto receba enorme atenção, principalmente devido ao seu grande valor comercial e as mais diversas aplicações possíveis, como verificação de identidade, controle de acesso, segurança, investigação de imagens em bancos de dados, vigilância, entretenimento ou realidade virtual. (2) (1)

O processo de reconhecimento facial de forma automatizada é separado em 4 principais etapas, conforme detalhado no livro 3, primeiramente deve ser feita a *detecção facial*, que consiste em validar e localizar a existência de alguma face na imagem ou video, a segunda etapa consiste no *alinhamento facial*, para que todas faces da base de dados sigam o mesmo padrão, a terceira etapa é a *extração de características* que permite a obtenção de informação efetiva que será útil na distinção das diferentes faces, a quarta e última etapa consiste na *correspondência de características*, onde as características extraídas anteriormente são comparadas com outras já conhecidas para que sejam identificadas.

Aprofundando o estudo da primeira etapa, de *detecção facial*, a referência 4 indica duas diferentes metodologias, a primeira baseada em características e a segunda baseada em imagens, ambas posteriormente podem ser separadas em diversas técnicas mais específicas, como por exemplo a análise de características por constelação ou a análise de imagens com redes neurais, onde cada técnica específica possui seus prós e contras em relação as demais.

## 0.1 Objetivos e Motivação

Este trabalho tem como objetivo encontrar a melhor forma de atuar sobre a primeira etapa (*detecção facial*) e a segunda etapa (*alinhamento facial*) do processo de reconhecimento facial, avaliando o desempenho qualitativo e quantitativo de diferentes metodologias e ferramentas disponíveis.

Atualmente empresas e órgãos públicos possuem a necessidade de manter cadastros pessoais mas existe grande demanda para que estes cadastros sejam feitos de forma totalmente virtual pela população, pois isso evita o deslocamento de pessoas até os pontos de cadastro e torna todo o processo muito mais ágil. Certos cadastros incluem fotos de identificação e isto traz a necessidade de uma verificação feita por humanos para validar

se a mesma consiste em uma foto de face frontal, conforme é necessário para o cadastro. A validação citada já ocorre e é feita de forma totalmente manual, onde funcionários tem que verificar cada uma das imagens recebidas e muitas vezes se deparam com fotos sem nenhuma face frontal ou sem condições de serem identificadas (desfocadas, por exemplo), que são rejeitadas para que uma nova imagem seja solicitada. Estas imagens claramente inválidas por estarem em desacordo com padrão esperado (foto de face frontal), poderiam facilmente ser eliminadas por uma filtragem anterior, reduzindo grande parte do trabalho que é feito hoje manualmente.

# 1 Materiais e Métodos

Com o objetivo de obter melhor entendimento sobre possíveis ferramenta de visão computacional e detecção facial, este trabalho descreve a execução de testes utilizando a linguagem de programação Python e a implementação da ferramenta OpenCV na mesma linguagem.

## 1.1 OpenCV

A ferramenta OpenCV, que pode ser encontrada no endereço [5](#), é uma biblioteca de código aberto focada na solução de problemas utilizando visão computacional em tempo real, desenvolvida pela intel e posteriormente pela Itseez, com suporte a múltiplas plataformas e uso gratuito sobre a licença de código aberto BSD. A ferramenta apresenta suporte a frameworks de aprendizado profundo, como TensorFlow, Pytorch e Caffe e contempla tanto funções básicas, para aplicações como processamento de imagem, alteração de cor ou resolução, até aplicações avançadas, como detecção facial, identificação de características e biometria. [\(6\)](#)

Neste trabalho, será utilizada a função de detecção de faces da ferramenta OpenCV, que utiliza um classificador em cascata baseado características, este é um método eficiente para reconhecimento de faces em imagens proposto por Paul Viola and Michael Jones, amplamente conhecido como método Viola-Jones, onde uma função é treinada com muitos exemplos positivos (imagens que contém o objeto a ser detectado) e negativos (imagens que não contém o objeto a ser detectado) e então utilizada para detectar as mesmas características em outras imagens. [\(7\)](#)

A detecção de faces utilizando OpenCV consiste em duas etapas principais, o treinamento do modelo, onde são apresentadas diversas imagens já identificadas para que o modelo identifique padrões positivos e negativos. Após o treinamento, o modelo obtido pode ser utilizado para identificar, em novas imagens, características semelhantes as vistas nas imagens do treinamento. Neste projeto, será utilizado um modelo fornecido em conjunto com a ferramenta OpenCV, já treinado com diversos exemplos de faces frontais.

## 1.2 Algoritmo Viola-Jones

O algoritmo Viola-Jones foi publicado em 2001, no paper "Rapid object detection using a boosted cascade of simple features" [\(8\)](#) e é famoso por sua capacidade de detecção de faces com muita velocidade, isso ocorre devido a 3 principais técnicas utilizadas: o

cálculo da imagem integral, o algoritmo de impulsão *AdaBoost* e o classificador em cascata.

### 1.2.1 Imagem Integral

A primeira etapa do algoritmo Viola-Jones consiste em transformar a imagem original em uma imagem integral, isto é feito calculando o valor de cada pixel como a soma de todos os pixels que estão acima ou a esquerda do mesmo, como ilustrado na figura 1.

Figura 1 – Imagem original (esquerda) e imagem integral (direita).

1	1	1
1	1	1
1	1	1

Input image

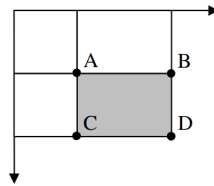
1	2	3
2	4	6
3	6	9

Integral image

Fonte: Jensen (2008)

A utilização desta técnica permite calcular facilmente o tamanho de qualquer retângulo formado entre quatro pixels da imagem, conhecendo apenas o valor dos seus cantos, possibilitando assim a análise rápida de diversas partes da imagem. Tal calculo é feito definindo o retângulo a ser analisado e então aplicando a equação 1.1.

Figura 2 – Representação da área da imagem a ser analisada.



Fonte: Jensen (2008)

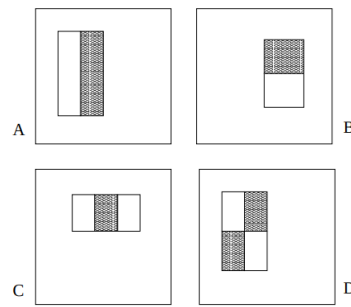
$$\text{Soma do retângulo cinza} = D - (B + C) + A \quad (1.1)$$

Com a possibilidade de calcular facilmente a soma dos pixels de um retângulo arbitrário de forma rápida, o algoritmo para detecção pode analisar diversos trechos da imagem, chamados aqui de características, fazendo a comparação de duas ou mais áreas retangulares predefinidas, como os exemplos ilustrado na figura 3.

O valor final de cada característica é definido pela soma do valor dos pixels sob o retângulo cinza menos a soma do valor dos pixels sob o retângulo branco.



Figura 3 – Alguns exemplos de características retangulares analisadas.



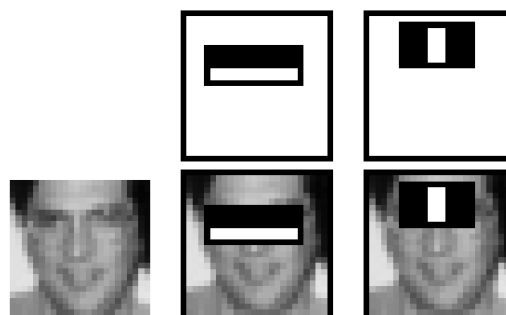
Fonte: Viola e Jones (2001)

### 1.2.2 Algoritmo de Impulsão AdaBoost

As características demonstradas anteriormente, são definidas basicamente como duas ou mais áreas retangulares de qualquer tamanho, tal simplicidade implica na possibilidade da criação de uma enorme variação das mesmas que precisariam ser calculadas diversas vezes, para cada parte de imagem e com diferentes tamanhos, isso implica em um alto custo de processamento, para evitar tal problema, durante a etapa de treinamento do modelo, é utilizado o algoritmo de impulsão *AdaBoost*, que identifica quais são as características com maior probabilidade de acerto.

O *AdaBoost*, que tem seu nome derivado de *adaptive boosting* (traduzido como impulsão adaptativa), é um método de aprendizado de máquina que utiliza a combinação de vários classificadores fracos para obter uma classificação forte, no caso da detecção facial, o algoritmo é utilizado tanto para selecionar um conjunto de características mais eficientes como para treinar o classificador. (10)

Figura 4 – Características retangulares mais eficientes para detecção facial.



Fonte: Viola e Jones (2001)

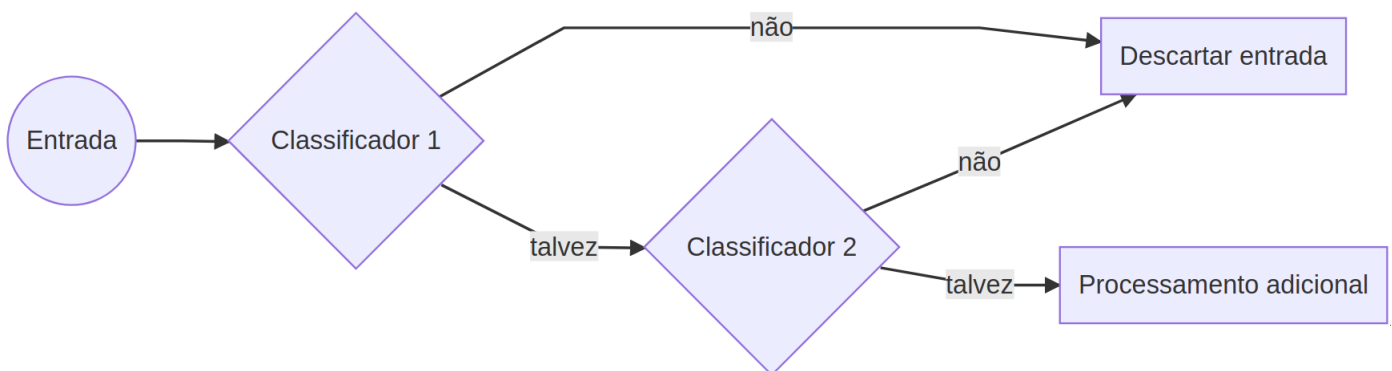
A figura 4 retrata as melhores características registradas por Viola e Jones (2001), fica claro que as mesmas se destacam por evidenciar as regiões dos olhos e do nariz.

### 1.2.3 Classificador em Cascata

Pensando que, na maioria dos casos, uma face não ocupa a maior parte de uma imagem a ser identificada, é necessário encontrar uma forma rápida de descartar os elementos do fundo da mesma e concentrar o poder de processamento nos elementos que tem maior probabilidade de serem reconhecidos como uma face, isso leva a uma formulação para o problema onde ao contrário de encontrar faces, é necessário um algoritmo que descarte as "não faces".

Para tal problema, o classificador em cascata apresenta uma ótima solução, esta consiste na utilização de uma série de classificadores que são aplicados de forma sequencial, conforme ilustrado na imagem 5, permitindo que imagens que certamente não possuem faces sejam rapidamente descartadas logo nas primeiras iterações, enquanto imagens com possíveis faces são classificadas por toda cascata, trazendo um elevado nível de confiança ao resultado.

Figura 5 – Diagrama do funcionamento do classificador em cascata.



Fonte: Viola e Jones (2001)

Um classificador comum, com um único estágio normalmente aceitaria muitos casos de falso negativo, para reduzir a taxa de falsos positivos e de descarte de imagens relevantes, mas no classificador em cascata, falsos positivos nos primeiros estágios não são um problema, pois serão analisados em outros diversos estágios e provavelmente eliminados.

A utilização desse modelo combinada com o algoritmo *AdaBoost*, possibilita a análise das característica mais eficientes logo no início e consequentemente o descarte muito mais rápido dos casos negativos nos primeiros estágios.

## 2 Resultados e Discussão

Para analisar a eficiência da implementação do algoritmo *Viola-Jones* na biblioteca *OpenCV*, foi utilizado o dataset de imagens *UTKFace*, disponível em [11](#), que consiste em um conjunto de mais de 20 mil imagens, com uma única face em cada, de diversas pessoas entre 0 e 116 anos de idade, catalogadas de acordo com idade, raça e sexo, alguns exemplos das imagens contidas no dataset podem ser vistos na figura 6.

Figura 6 – Exemplos de imagens do dataset utilizado.



Fonte: *UTKFace dataset* ([11](#))

No primeiro teste, foram filtradas apenas imagens de pessoas entre 18 e 60 anos de idade e executado um teste de detecção de faces sob um grupo aleatório de 5 mil imagens desse grupo, os resultados podem ser vistos na tabela 1.

Tabela 1 – Resultados Teste 1.

Número de faces	Número de imagens	Porcentagem
0	192	5.4763%
1	2659	75.8414%
Mais de 1	655	18.6823%

No segundo teste, foram novamente filtradas apenas imagens de pessoas entre 18 e 60 anos de idade e desta vez executado um teste de detecção de faces sob todo o grupo que se enquadrava no filtro, sendo um total de 17130 imagens, os resultados podem ser vistos na tabela 2.

No terceiro teste, foi novamente executado um teste de detecção de faces sob o mesmo grupo do segundo teste (17130 imagens), mas foram alterados alguns parâmetros do modelo de detecção para que este se tornasse mais sensível, ou seja, detectaria uma

Tabela 2 – Resultados Teste 2.

Número de faces	Número de imagens	Porcentagem
0	891	5.2014%
1	13111	76.5382%
Mais de 3128	655	18.2604%

maior quantidade de faces, mas com mais chance de ocorrência de falsos positivos, os resultados podem ser vistos na tabela [3](#).

Tabela 3 – Resultados Teste 3.

Número de faces	Número de imagens	Porcentagem
0	263	1.5353%
1	8439	49.2645%
Mais de 1	8428	49.2002%

# Referências

- 1 ZHAO, W. et al. Face recognition: A literature survey. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 35, n. 4, p. 399–458, dez. 2003. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145/954339.954342>. Citado na página 3.
- 2 PARMAR, D.; MEHTA, B. Face recognition methods & applications. *International Journal of Computer Technology and Applications*, v. 4, p. 84–86, 01 2014. Citado na página 3.
- 3 LI, S. Z.; JAIN, A. K. *Handbook of Face Recognition*. 2nd. ed. [S.l.]: Springer Publishing Company, Incorporated, 2011. ISBN 085729931X, 9780857299314. Citado na página 3.
- 4 HJELMÅS, E.; LOW, B. K. Face detection: A survey. *Computer Vision and Image Understanding*, v. 83, n. 3, p. 236 – 274, 2001. ISSN 1077-3142. Disponível em: <http://www.sciencedirect.com/science/article/pii/S107731420190921X>. Citado na página 3.
- 5 ITSEEZ. *Open Source Computer Vision Library*. 2015. <https://github.com/itseez/opencv>. Citado na página 5.
- 6 WIKIPEDIA. *OpenCV — Wikipedia, The Free Encyclopedia*. 2019. <http://en.wikipedia.org/w/index.php?title=OpenCV&oldid=906212825>. [Online; accessed 31-July-2019]. Citado na página 5.
- 7 ITSEEZ. *The OpenCV Reference Manual*. 2.4.9.0. ed. [S.l.], 2014. Citado na página 5.
- 8 VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: . [S.l.: s.n.], 2001. v. 1, p. I–511. ISBN 0-7695-1272-0. Citado 3 vezes nas páginas 5, 7 e 8.
- 9 JENSEN, O. H. *Implementing the Viola-Jones face detection algorithm*. Dissertação (Mestrado) — Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark, 2008. Citado na página 6.
- 10 SANTANA, L. M. Queiroz de; ROCHA, F. Processo de detecção facial utilizando viola;jones. *Interfaces Científicas - Exatas e Tecnológicas*, v. 1, 02 2015. Citado na página 7.
- 11 IMAGING, A.; PROCESSING, C. I. *UTK Face Dataset*. [Online; accessed 04-August-2019]. Disponível em: <https://susanqq.github.io/UTKFace/>. Citado na página 9.