

DESENVOLVIMENTO DE UM MÉTODO SEMI-SUPERVISIONADO DE AGRUPAMENTO DE FACES

Leandro Teixeira Martins¹; Marcos Roberto Ribeiro ²;

¹ Bolsista (CNPq), Engenharia de Computação, IFMG Campus Bambuí, Bambuí - MG; lleandroltm@hotmail.com

² Orientador: Pesquisador do IFMG, Campus Bambuí; marcos.ribeiro@ifmg.edu.br

RESUMO

O agrupamento de faces é uma tarefa que envolve as áreas de Mineração de Dados e Visão Computacional. Apesar de haver pesquisas consideráveis sobre o agrupamento de faces, ainda existem melhorias interessantes que podem ser realizadas. O presente trabalho visa desenvolver um método de agrupamento de faces semi-supervisionado de forma que as intervenções dos usuários possam ser aprendidas e usadas de forma automática no futuro. Além disso, o método será implementado e disponibilizado através de um software livre, permitindo que usuários possam usar, aprimorar e modificar a ferramenta no futuro. Para desenvolver o trabalho foi feito um estudo dos principais algoritmos de detecção de faces e agrupamento de dados. Com a pesquisa dos algoritmos foram encontrados quatro métodos de detecção facial, o *Hass Cascade* e o aprendizado profundo da OpenCV, além dos métodos HOG e CNN da *Face Recognition*. Para os algoritmos de agrupamento também foram selecionados quatro algoritmos, sendo eles o DBSCAN, *k-means*, *MeanShift* e OPTICS, todos pertencentes a biblioteca *Scikit-learn*. Esses algoritmos passaram por testes comparativos para a escolha daqueles mais adequados e que possuem melhor desempenho. Os resultados dos experimentos apontaram uma ótima performance do algoritmo de aprendizado profundo, possuindo o melhor resultado para base de dados Fddb, que possui imagens com alta resolução se aproximando das fotos tiradas por dispositivos atuais. Para os experimentos com os algoritmos de agrupamento de dados, o DBSCAN chamou atenção pela velocidade de execução e um ótimo resultado comparado com os outros métodos disponibilizados pela biblioteca. Utilizando os algoritmos dos experimentos está sendo feita a modelagem e modificação do método proposto, para que possa ocorrer o aprendizado semi-supervisionado, adicionando a intervenção humana quando o resultado do agrupamento possuir alguma incerteza. Como alternativa para implantação do método proposto o sistema PhotoFaces se mostrou bem promissor, possuindo licença livre e construção utilizando a linguagem Python. O trabalho se encontra na etapa final restando somente a modificação e implantação do método proposto.

Palavras Chave: Agrupamento de Dados; Detecção Facial; Aprendizado Semi-Supervisionado.

INTRODUÇÃO

A área de Visão Computacional, em especial a detecção de faces, tem despertado o interesse de vários pesquisadores. Os principais objetivos dos pesquisadores são aumentar o desempenho dos algoritmos existentes ou desenvolver outros métodos mais eficientes e eficazes (VIOLA; JONES, 2001; LI *et al.* 2015; ZHANG *et al.* 2016; RAKSHIT *et al.* 2020).

Com a grande popularização da Internet e de dispositivos móveis inteligentes, houve um grande aumento na produção de dados. Os métodos tradicionais, muitas vezes, não são adequados para analisar esses grandes volumes de dados, sendo interessante a utilização de técnicas de mineração de dados. Uma dessas técnicas é o agrupamento de dados que visa reunir objetos que tenham alguma semelhança entre si, e manter separados os objetos diferentes (XU; WUNSCH, 2005; TAN; STEINBACH; KUMAR, 2009).

No desenvolvimento de pesquisas, pode ser interessante unir mais de um campo da ciência. Esse é o caso de uma ferramenta capaz de agrupar fotos usando métodos de agrupamentos de dados com base em faces obtidas por meio de técnicas de detecção facial. Tal ferramenta apresenta grande utilidade, tendo em vista que os usuários tendem a produzir quantidades cada vez maiores de fotos. Sendo assim, organizar esse grande volume de fotos pode se tornar uma tarefa que consome muito tempo.

Mesmo os melhores algoritmos de agrupamento de dados podem cometer erros (XU; WUNSCH, 2005; BIJL, 2018). No caso do agrupamento de faces, as mesmas são traduzidas para valores numéricos e os algoritmos comparam esses valores para decidir se duas faces são de uma mesma pessoa. Pequenas variações nas faces podem acarretar certas mudanças em suas representações numéricas de forma que o agrupamento seja feito de forma errada. Por outro lado, os seres humanos possuem boa capacidade para lidar com padrões visuais (FUKUSHIMA, 1988). Dessa maneira, em muitas situações, um ser humano pode avaliar melhor se duas faces são de uma mesma pessoa do que um algoritmo de agrupamento. Uma abordagem eficiente seria ter uma intervenção humana nos casos em que o algoritmo errar, melhorando

assim o processo de agrupamento das faces. Além disso, as correções feitas pelo humano podem ser utilizadas por técnicas de aprendizado de máquina para evitar a repetição de um mesmo erro.

Já existem ferramentas comerciais capazes de fazer o agrupamento de faces, inclusive com supervisão humana, como, por exemplo, o Google Photos. Entretanto, as ferramentas comerciais possuem limitações como redução da resolução das fotos e problemas de segurança dos dados (ROHR, 2020). Além disso, tais ferramentas podem ser descontinuadas ou passarem a ser monetizadas. No que diz respeito à ciência, é importante que existam ferramentas abertas similares. Isso permite o aprimoramento de técnicas existentes e o desenvolvimento de novos métodos.

Uma alternativa às ferramentas comerciais são os softwares livres. O grande diferencial desse tipo de software é a liberdade de acesso, utilização, compartilhamento e alteração do código (FSF, 2020). Sendo assim, o desenvolvimento de softwares livres pode trazer benefícios para a ciência, principalmente no que diz respeito à disponibilidade de informação.

O processo de agrupar fotos de acordo com as pessoas presentes nas mesmas, não é uma tarefa fácil, principalmente, por causa da quantidade de fotos cada vez maior produzida pelos usuários. Até onde foi pesquisado, não existe um software livre capaz de fazer o agrupamento de fotos com base em faces de forma semi-supervisionada. Dessa maneira, a pesquisa para o desenvolvimento de tal software pode trazer benefícios para a ciência, uma vez que métodos já desenvolvidos podem ser aprimorados e novos métodos podem ser descobertos. O objetivo do presente trabalho é melhorar um método de agrupamento existente, adicionando a intervenção humana e disponibilizando o mesmo através de um software livre.

METODOLOGIA

A primeira etapa do presente trabalho envolveu a seleção dos principais algoritmos e métodos de detecção facial e agrupamento de dados. Essa seleção foi realizada com base em uma pesquisa de literatura. Após essa etapa, foram realizados os experimentos comparativos entre os algoritmos.

Os experimentos com os algoritmos de detecção facial fizeram uso de duas bibliotecas. Uma delas foi a *Open Source Computer Vision Library* (OpenCV), uma biblioteca de código aberto com foco em visão computacional e aprendizado de máquina. A OpenCV disponibiliza o método *Haar Cascade* de Viola e Jones (2001) e o método de aprendizado profundo que usa redes neurais artificiais. A outra biblioteca utilizada foi a *Face Recognition* que visa facilitar o reconhecimento e a detecção de faces. Tal biblioteca disponibiliza os métodos de detecção *Histogram of Oriented Gradients* (HOG) e *Convolutional Neural Network* (CNN).

Para os experimentos de detecção facial foi feita uma pesquisa para encontrar as bases de dados que possuem anotações da localização das faces na imagem, para que seja possível realizar a avaliação dos métodos. A Tabela 1 possui as bases de dados encontradas e suas características. A base de dados WIDER possui uma grande variedade de tipos de imagens, contendo também imagens de multidões. As informações disponibilizadas são as anotações da caixa delimitadora retangular e alguns atributos, como por exemplo, grau de borrão das faces. Já a base FDDB disponibiliza a região da face em formato de uma elipse, contendo as coordenadas do centro, o valor do maior raio, o valor do menor raio e o ângulo de inclinação do rosto. Tendo em vista que a base WIDER possui uma grande variedade de imagens e a FDDB possui imagens com boa resolução, as duas bases de dados são suficientes para os testes.

Tabela 1 – Base de Dados para Detecção Facial

Base de Dados	Quantidade de Imagens	Quantidades de Faces
WIDER FACE	32.203	393.703
FDDB	2.845	5.171

Fonte: Os Autores, 2021

Os algoritmos de detecção facial foram submetidos aos experimentos utilizando bases de dados diferentes, com o objetivo de identificar qual deles possui melhor desempenho, avaliando a qualidade do resultado e quantas faces cada algoritmo consegue encontrar. A avaliação é composta pela precisão, *recall* e acurácia, onde a acurácia é a medida de quantas faces foram encontradas em relação a quantidade de faces existentes na base de dados. A qualidade da detecção determina se a face encontrada está correta ou se é um erro, podendo assim contabilizar os falsos positivos. Para análise da avaliação foi feito um *script* que verifica cada saída do algoritmo e compara com as anotações fornecidas pela base de dados, contabilizando as faces encontradas para o cálculo da acurácia e avaliando a qualidade da detecção para contabilizar os falsos positivos.

Quanto aos métodos de agrupamento foi utilizada a biblioteca *Scikit-learn*, que possui diversos algoritmos de aprendizado de máquina implementados. Os algoritmos de agrupamento de dados utilizados foram: DBSCAN, *k-means*, *MeanShift* e OPTICS.

Para os experimentos com os algoritmos de agrupamento foram utilizadas outras bases de dados, que possuem imagens com identificação das faces e não da localização. A Tabela 2, possui a relação de quantidade de imagens e quantidade de pessoas das bases de dados encontradas. Em concordância com Campos (2019), as 3 bases usadas possuem variações diferentes. Com a LFW é possível avaliar o desempenho dos métodos quando se tem muitas imagens e muitos indivíduos, em contra partida a *14-celebrity-faces-dataset* permite analisar o mesmo aspecto com poucas imagens e poucos indivíduos. E por fim, a base IMPA-FACE3D que possui variações de poses e expressões.

Tabela 2 – Base de Dados para o agrupamento

Base de Dados	Quantidade de Imagens	Quantidades de Pessoas
LFW	13.233	5.749
<i>14-celebrity-faces-dataset</i>	70	14
IMPA-FACE3D	535	38

Fonte: Os Autores, 2021

Dentre as métricas de avaliação encontradas foi utilizada a *F-measure* também chamada de *F1-score*. Abualigah, Khader e Hanandeh (2018) e Bijl (2018), utilizam a *F-measure* para avaliar o resultado de suas pesquisas, pelo fato de que é uma métrica interessante por combinar duas medidas clássicas na mineração de dados, que são a precisão e o *recall*. De posse dos resultados comparativos entre os algoritmos de agrupamento, o próximo passo é adaptar o melhor algoritmo, acrescentando a intervenção do usuário para os dados duvidosos. O resultado de tal intervenção é armazenado para que sejam utilizados no futuro, caso ocorra o mesmo problema. O desenvolvimento do trabalho se encontra na etapa final, onde está sendo feita a modificação do método para que possa ser implementado no sistema PhotoFaces.

RESULTADOS E DISCUSSÕES

Após os primeiros experimentos de detecção facial, foi possível observar que a base de dados WIDER possui algumas imagens com grande número de faces, causando um alto grau de borrão e uma escala reduzida. Essas faces possuem um nível de dificuldade de detecção elevado. A partir disso, foi definido um filtro para desconsiderar esse tipo de face e não prejudicar a avaliação dos algoritmos. O filtro elege para contagem as faces com tamanho maior que 25 pixels, que não possuem borrão e que sejam faces válidas.

Além da contagem de faces encontradas, foi criado um script que realiza a contagem dos falsos positivos, que são as faces que o algoritmo retornou que não possui correspondente na base de dados. Para cada face encontrada é calculado a diferença entre a coordenada X do ponto central detectado, com todas as coordenadas X dos pontos centrais das faces da imagem. A face da imagem que tiver a menor diferença, corresponde a detecção da face analisada. As detecções com uma diferença maior que 50 pixels são contabilizados como falsos positivos, esse valor foi encontrado no decorrer dos testes. A coordenada Y não é considerada, porque alguns algoritmos incluem na caixa delimitadora a testa e o queixo, já outros desconsideram essas regiões, contendo assim somente boca, olhos e nariz. Dessa forma, a coordenada Y pode possuir uma grande variação de um algoritmo para o outro.

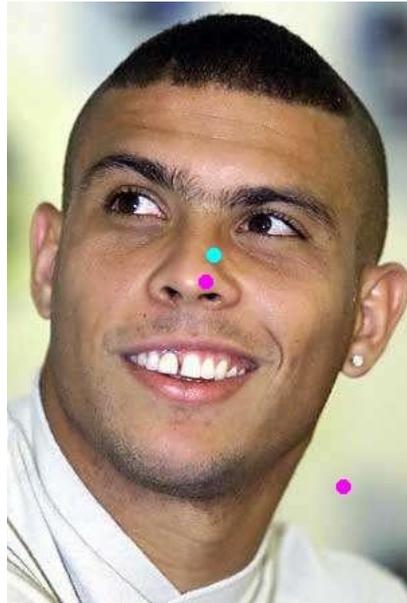
A Figura 1 apresenta dois casos, um onde a detecção está correta e outro onde o algoritmo errou. Os pontos na imagem foram marcados com os dados das anotações da base de dados e com o resultado da detecção do algoritmo de aprendizado profundo da OpenCV. O ponto em azul claro corresponde ao ponto central da caixa delimitadora fornecida pela base de dados, enquanto os pontos em rosa correspondem aos pontos centrais de duas caixas delimitadoras que o algoritmo retornou como resultado. No caso apresentado, será contabilizado um falso positivo e um verdadeiro positivo.

É possível observar que, quando o algoritmo acerta, há uma pequena diferença do ponto central, porém ela não é muito grande quando comparada à uma detecção errada. A distância também pode ser influenciada pela escala da face, quando a face é pequena em comparação ao tamanho da imagem a diferença entre a detecção e a face real também pode ser pequena. Para resolver esse problema pode ser usado o tamanho da face como referência, logo, se a diferença dos pontos centrais for maior que uma porcentagem do tamanho da face, quer dizer que a detecção está errada e não corresponde a face em questão. Com essa verificação de erros foi possível contabilizar os falsos positivos, que fazem parte do cálculo da precisão apresentados na Tabela 3.

Os resultados contidos na Tabela 3 representam a precisão, *recall* e acurácia dos algoritmos, utilizando como referência as anotações das faces pertencentes a base de dados WIDER e Fddb. Observando os resultados obtidos com o uso da base WIDER, a mesma possui um grau de dificuldade maior no processo de encontrar a região da face. Se atentando também ao fato de que algumas faces não foram

consideradas. É nítido que os métodos disponibilizados pela *Face Recognition* não possuem diferenças, quando comparado a quantidade de faces encontradas. Outra questão é que o algoritmo *Haar Cascade* obteve o pior resultado nas duas bases de dados, com uma grande diferença quando comparado com os outros métodos. Esse resultado era esperado devido ao *Haar Cascade* ter dificuldade com faces que não sejam frontais. Já o método de Aprendizado profundo teve um bom resultado na base WIDER e obteve a melhor precisão quando aplicado nas imagens da FDDB.

Figura 1 – Face detectada com método de aprendizado profundo da OpenCV



Fonte: Adaptado de (JAIN; LEARNED-MILLER, 2010)

Tabela 3 – Resultados dos algoritmos de detecção facial

Base de Dados	Algoritmos de Detecção Facial	Precisão	Recall	Acurácia
WIDER	OpenCV - Aprendizado profundo	0,95	0,56	0,55
	OpenCV - <i>Haar Cascade</i>	0,99	0,21	0,21
	<i>Face Recognition</i> - HOG	0,97	0,67	0,66
	<i>Face Recognition</i> - CNN	0,97	0,67	0,66
FDDB	OpenCV - Aprendizado profundo	0,86	0,98	0,85
	OpenCV - <i>Haar Cascade</i>	1,00	0,28	0,28
	<i>Face Recognition</i> - HOG	0,99	0,77	0,77
	<i>Face Recognition</i> - CNN	0,99	0,77	0,77

Fonte: Os Autores, 2021

Utilizando os resultados encontrados e calculando a média das acurácias dos algoritmos nas duas bases de dados, o método de aprendizado profundo obtém uma média de 0,70 ficando atrás do HOG e CNN com 0,71, possuindo assim uma pequena diferença que dificulta a escolha do melhor algoritmo. Porém, analisando a acurácia dos algoritmos na base FDDB que possui melhor resolução se aproximando das imagens atuais, o melhor desempenho é do algoritmo de aprendizado profundo. Com base nessas análises o método de aprendizado profundo foi o escolhido para os testes com os algoritmos de agrupamento de dados.

Na execução dos algoritmos de agrupamento alguns parâmetros possuem influência direta no resultado. Para avaliar melhor o desempenho dos algoritmos foi realizado vários testes para encontrar o melhor valor para os parâmetros. No caso do algoritmo *k-means* o parâmetro passado é o número de grupos que ele vai gerar, como esse número já é conhecido não é necessário encontrar o valor ideal. Para o algoritmo DBSCAN, o parâmetro que possui maior influência e que é considerado o mais importante segundo Pedregosa et al. (2011) é o *eps*, em todos os testes o parâmetro *minSamples* foi definido como 1, considerando que pode existir grupos com uma única face. O parâmetro analisado no algoritmo OPTICS foi o *minSamples*, e para o *MeanShift* foi o *bandwidth* que possui uma função que o define automaticamente,

porém não alcança bons resultados. Os melhores valores para os parâmetros citados estão apresentados na Tabela 4, relacionando o algoritmo a cada base de dados.

Tabela 4 – Valores dos Parâmetros usados nos Algoritmos de Agrupamento

Algoritmos (Parâmetro)	14-celebrity-faces-dataset	IMPA-FACE3D	LFW
DBSCAN (<i>eps</i>)	0,57	0,43	0,43
OPTICS (<i>minSamples</i>)	3	6	2
<i>MeanShift</i> (<i>bandwidth</i>)	0,52	0,43	0,43

Fonte: Os Autores, 2021

Os resultados considerando os parâmetros da Tabela 4, são apresentados na Tabela 5 que possui os valores de desempenho usando todos os algoritmos em cada base de dados apresentada. Para a análise do melhor algoritmo, foi apresentado o *F-measure* e a quantidade de grupos formados pelos algoritmos. De forma geral, os algoritmos DBSCAN e *MeanShift* possuem uma grande semelhança, obtendo a mesma métrica para todas as bases de dados, diferenciando somente na quantidade de grupos formados.

Para a LFW e *14-celebrity*, os melhores resultados são do DBSCAN juntamente com *MeanShift*, tendo uma diferença na quantidade de grupos formados para base LFW, onde o *MeanShift* obteve uma quantidade menor de grupos. Porém, a diferença é pouca olhando para o tamanho da base de dados. Em concordância com Campos (2019), considerando o tempo de execução, o DBSCAN possui uma enorme vantagem conseguindo um tempo melhor, sendo assim uma ótima opção. E por fim, na base IMPA-FACE3D os algoritmos obtiveram um desempenho bem próximo, com melhor resultado do algoritmo *K-means*.

Tabela 5 – Resultado dos Algoritmos de Agrupamento

Base de Dados	Algoritmos	Grupos Formados	F-measure
<i>14-celebrity-faces-dataset</i>	DBSCAN	18	0,88
	OPTICS	15	0,85
	<i>MeanShift</i>	18	0,88
	<i>K-means</i>	14	0,84
LFW	DBSCAN	7674	0,77
	OPTICS	2114	0,30
	<i>MeanShift</i>	7078	0,77
	<i>K-means</i>	5749	0,56
IMPA-FACE3D	DBSCAN	62	0,83
	OPTICS	36	0,84
	<i>MeanShift</i>	55	0,83
	<i>K-means</i>	38	0,87

Fonte: Os Autores, 2021

A solução proposta pelo presente trabalho atua no final do processo do agrupamento de faces. Antes do resultado final, é feita uma verificação para constatar se há uma incerteza no agrupamento da face. Essa incerteza acontece quando a imagem em questão possui características pertencentes a dois grupos diferentes. Após essa verificação, se houver um grau de incerteza considerável serão formuladas perguntas para o usuário. A pergunta irá relacionar a imagem com os possíveis grupos que ela pode pertencer. Essa modelagem está sendo realizada para que possa melhorar o resultado do agrupamento das faces. O sistema de Campos (2019) nomeado como PhotoFaces, se mostrou bem promissor como alternativa para implantação do método proposto, possuindo licença livre e construção utilizando a linguagem Python.

CONCLUSÕES

O objetivo deste trabalho foi o desenvolvimento de uma método semi-supervisionado de agrupamento de faces, selecionando os melhores algoritmos de detecção de faces e agrupamento de dados já implementados. Após a seleção e testes com os algoritmos encontrados, o método de detecção de faces e agrupamento que mais se destacaram são o aprendizado profundo da OpenCV e o DBSCAN da *Scikit-learn*. Com os algoritmos selecionados a modelagem e modificação está sendo realizada para inclusão da

intervenção humana proposta. Após a modelagem o método será implementado em um software livre para que possa ser disponibilizado.

REFERENCIAS BIBLIOGRÁFICAS

ABUALIGAH, Laith Mohammad; KHADER, Ahamad Tajudin; HANANDEH, Essam Said. A combination of objective functions and hybrid krill herd algorithm for text document clustering analysis. **Engineering Applications of Artificial Intelligence**, ago. 2018, v. 73, p. 111-125. Disponível em: <http://encurtador.com.br/enCJU>. Acesso em: 20 fev. 2021.

BIJL, E. **A comparison of clustering algorithms for face clustering**. 2018. Tese (Doutorado) – University of Groningen, Groningen, 2018. Disponível em: <http://fse.studenttheses.ub.rug.nl/id/eprint/18064>. Acesso em: 15 fev. 2020.

CAMPOS, M. V. R. **PHOTOFACES: um sistema de agrupamento de fotos com base em rostos**. 2020. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais, Bambuí, 2020.

FSF. **The Free Software Foundation (FSF) is a nonprofit with a worldwide mission to promote computer user freedom**. 2020. Disponível em: <https://www.fsf.org/about/>. Acesso em: 1 mar. 2020.

FUKUSHIMA, Kunihiro. Neocognitron: A hierarchical neural network capable of visual pattern recognition. **Neural networks**, 1988, v. 1, p. 119-130. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/0893608088900147>. Acesso em: 6 mar. 2020.

JAIN, Vidit; LEARNED-MILLER, Erik. **Fddb: A benchmark for face detection in unconstrained settings**. University of Massachusetts Amherst, 2010. Disponível em: <http://vis-www.cs.umass.edu/fddb/>. Acesso em: 23 mar. 2021.

LI, Haoxiang *et al.* A convolutional neural network cascade for face detection. **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2015. p. 5325-5334. Disponível em: http://openaccess.thecvf.com/content_cvpr_2015/papers/Li_A_Convolutional_Neural_2015_CVPR_paper.pdf. Acesso em: 20 fev. 2020.

PEDREGOSA, Fabian *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, 2011, v. 12, p. 2825–2830. Disponível em: <https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html>. Acesso em: 11 set. 2020.

RAKSHIT, Pranati *et al.* Face Detection using Support Vector Machine with PCA. **2nd International Conference on Non-Conventional Energy: Nanotechnology & Nanomaterials for Energy & Environment (ICNNEE)**, Índia, 9 jan. 2019. Disponível em: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3515989. Acesso em: 25 fev. 2020.

ROHR, Altieres. **Google vazou vídeos guardados no Google Fotos para outros usuários do serviço**. 2020. Disponível em: <https://encurtador.com.br/rJTX2>. Acesso em: 5 mar. 2020.

TAN, Pang-ning.; STEINBACH, Michael.; KUMAR, Vipin. **Introdução ao DATA MINING: Mineração de dados**. Rio de Janeiro: Ciência Moderna Ltda, 2009.

VIOLA, Paul; JONES, Michael. Rapid object detection using a boosted cascade of simple features. **Proceedings of the 2001 IEEE computer society conference on computer vision and pattern**

recognition, Kauai, HI, EUA, 15 abr. 2003. <https://ieeexplore.ieee.org/abstract/document/990517>. Acesso em: 10 fev. 2020.

XU, Rui; WUNSCH, Donald. Survey of clustering algorithms. **IEEE Transactions on neural networks**, 09 mai. 2005, v. 16, n. 3, p. 645-678. Disponível em: <https://doi.org/10.1109/TNN.2005.845141>. Acesso em: 22 fev. 2020.

ZHANG, Kaipeng. *et al.* Joint face detection and alignment using multitask cascaded convolutional networks. **IEEE Signal Processing Letters**, 26 ago. 2016, v. 23, n. 10, p. 1499–1503. Disponível em: <https://ieeexplore.ieee.org/abstract/document/7553523/authors#authors>. Acesso em: 15 fev. 2020.