SEVEN
publicações acadêmicas

# Uma abordagem computacional em tempo real para reconhecimento de expressão facial humana baseada na extração de recursos de referência

https://doi.org/10.56238/sevened2024.004-006

**Dennis P. Lopez**
Departamento de Computação – DEC, Universidade Federal de Santa Catarina, Brasil.
E-mail: dppazlopez@gmail.com

**Felipe Z. Canal**
Departamento de Computação – DEC, Universidade Federal de Santa Catarina, Brasil.
Programa de Pós-Graduação em Tecnologias da Informação e Comunicação, Universidade Federal de Santa Catarina, Brasil.
E-mail: felipe.canal@grad.ufsc.br

**Gustavo G. Scotton**
Departamento de Computação – DEC, Universidade Federal de Santa Catarina, Brasil.
E-mail: gustavo.gino@outlook.com

**Eliane Pozzebon**
Departamento de Computação – DEC, Universidade Federal de Santa Catarina, Brasil.
Programa de Pós-Graduação em Tecnologias da Informação e Comunicação, Universidade Federal de Santa Catarina, Brasil.
E-mail: eliane.pozzebon@ufsc.br

**Antonio C. Sobieranski**
Departamento de Computação – DEC, Universidade Federal de Santa Catarina, Brasil.
Programa de Pós-Graduação em Tecnologias da Informação e Comunicação, Universidade Federal de Santa Catarina, Brasil.
E-mail: a.sobieranski@ufsc.br

**RESUMO**
O reconhecimento de expressão facial humana em tempo real desempenha um papel significativo em muitas áreas de aplicação, incluindo interação humano-computador, inteligência de negócios, vigilância por vídeo e robótica. Com base em expressões faciais, os computadores podem interpretar sentimentos humanos e estágios psicológicos para promover aproximações mais realistas em aplicações do mundo real. Este artigo propõe uma solução simples, mas eficaz, para o Reconhecimento de Emoções Faciais (FER) em tempo real, usando uma máscara das características faciais mais relevantes como dados de entrada para uma abordagem de aprendizado de máquina. Para isso, um classificador compacto de Rede Neural Convolucional (CNN) associado a uma camada de extração de feição foi usado para fornecer uma solução de ponta a ponta que pode detectar expressões faciais de vídeos com boas taxas de precisão. A abordagem proposta foi validada usando uma combinação de diferentes conjuntos de dados de emoções faciais disponíveis na literatura, cujos índices de precisão são consideravelmente melhores do que aqueles fornecidos pelos métodos de última geração. Taxas de pontuação de 96,83%, 98,58% e 98,57% foram obtidas para os conjuntos de dados JAFFE, RaFD e CK+, respectivamente, indicando que a abordagem apresentada é uma solução promissora para o RAF em aplicações em tempo real.

**Palavras-chave:** Reconhecimento de expressão facial em tempo real, Classificação das emoções, Extração de padrões faciais, Rede Neural Convolucional.

## 1 INTRODUÇÃO

O reconhecimento de emoções a partir de expressões faciais é uma tarefa relativamente simples para os seres humanos, fazendo parte da comunicação não-verbal diária regular, mas complexa para ser expressa analiticamente [1]. Os seres humanos são capazes de inferir emoções através de outros meios, como linguagem corporal, tom de voz e palavras faladas. No entanto, as características faciais contribuem significativamente para a expressão de emo- ções, uma vez que o rosto é o principal foco de atenção durante a comunicação. Muitos estudos apoiam a premissa de que alguns comportamentos faciais estão universalmente ligados às emoções, independentemente de diferenças culturais e étnicas [2], e podem ser reproduzidos computacionalmente.

Embora natural para os seres humanos, o Reconhecimento de Emoções Faciais (FER) representa um desafio significativo para as máquinas, uma vez que os rostos podem ter muitos atributos, como envelhecimento, forma, cor da pele, barbas e cicatrizes [3]. Além disso, as faces podem incluir alguma variabilidade, como óculos, iluminação irregular e oclusões, e podem ser visualizadas de diferentes perspectivas. Todos esses desafios podem ser considerados complexos para um método computacional para discriminar informações das faces e usar apenas os recursos relevantes necessários para classificar padrões emocionais.

Uma tendência computacional geral para o problema FER é extrair algumas façanhas da imagem facial e passá-las através de um classificador, abstraindo assim sua complexidade e possibilitando previsões. Algumas das técnicas de extração de características mais utilizadas são as abordagens clássicas, também conhecidas como métodos artesanais . Alguns exemplos são (i) as características baseadas em geometria [4, 5], que medem a posição e a forma de diferentes componentes da face, e (ii) as características de aparência [5], que envolvem detalhes mais complexos, como bordas e texturas. Existem também outros métodos usados para extração de características específicas, como o Histograma de Gradientes em Pirâmide (PHOG) [6], Quantização de Fase Local (LPQ) [6, 7] e Características de Wavelets de Gabor [8]. No entanto, com o recente surgimento de técnicas de Deep Learning, muitos métodos baseados em Redes Neurais Convolucionais (CNNs) estão se tornando solucionadores de problemas gerais em aplicações do mundo real. Redes bem conhecidas como AlexNet, GoogLeNet, VGG e ResNet [9–11] já foram utilizadas em diversas aplicações e na área de FER .

Com as CNNs, uma solução de ponta a ponta pode ser fornecida diretamente, pois a rede pode aprender os recursos mais relevantes a partir dos dados. No entanto, muitas amostras e considerável esforço computacional são necessários para treinar o modelo corretamente, o que leva a melhores resultados de classificação e generalizabilidade. Esta característica é a principal

diferença em comparação com os métodos artesanais tradicionais, que são adaptados explicitamente com antecedência e projetados por um especialista.

No entanto, muitas das abordagens de RAF encontradas na literatura dependem principalmente do conjunto de dados utilizado para treinar os modelos, que na maioria das vezes é um único. Conjuntos de dados contendo imagens de rosto ou vídeos desenvolvidos explicitamente para o FER não são novidade. Os primeiros estudos datam de 1969 [12], e muitos conjuntos de dados de expressão facial surgiram desde então, variando seu ambiente de aquisição, o número de expressões reconhecíveis, a faixa etária do sujeito, entre outras características. Alguns conjuntos de dados populares incluem JAFFE, CK+, MMI, RaFD e AffectNet [13–15]. Um conjunto de dados muito homogêneo ou pequeno pode causar overfit- ting e tem pouca representação para algumas classes de dados, e pode levar a algum viés indesejado. Pode até ter dados rotulados incorretamente e tornar o processo de aprendizagem mais complicado – foi relatado que a porcentagem de dados rotulados corretamente para o conjunto de dados FER foi de cerca de 65% [16].

Com essas limitações em mente, este artigo apresenta uma solução CNN híbrida simples, mas eficaz, explicitamente projetada para FER, capaz de processamento de vídeo em tempo real e desenvolvida considerando um conjunto de dados misto. Para isso, o método proposto utiliza uma técnica simples de extração de características geométricas, combinando descritores de pontos de referência de faces e usando-os para treinar e testar modelos de CNNs. O objetivo de usar recursos geométricos é sua invariância a limitações do mundo real, como tons de pele, iluminação e ângulos. Além disso, durante a etapa de treinamento, os dados de entrada (misturados de três bancos de dados) combinaram sete tipos de emoções: (i) neutra, (ii) felicidade, (iii) tristeza, (iv) surpresa, (v) raiva, (vi) medo e (vii) nojo. Os resultados mostraram taxas de precisão de 96,83%, 98,58% e 98,57% para os conjuntos de dados JAFFE, RaFD e CK+, respectivamente, mostrando taxas de acurácia marginalmente melhores quando comparados aos métodos encontrados na literatura.

O restante deste trabalho está organizado da seguinte forma: a seção 2 discute alguns trabalhos de destaque na área de ERF. A seção 3 descreve detalhadamente o método proposto e seus aspectos de implementação, seguida da seção 4, demonstrando os resultados obtidos. Finalmente, as conclusões, discussão e outros trabalhos são apresentados na Seção 5.

## 2 TRABALHOS RELACIONADOS

Na literatura, é possível identificar duas tendências claras em relação à área de RAF : (i) as abordagens clássicas, também conhecidas como métodos artesanais de seleção de características, e (ii) a contraparte da Rede Neural Convolucional.

As abordagens clássicas geralmente envolvem alguma técnica para obter um conjunto de características e alguma métrica ou classificador para discriminar vetores de entrada para a tomada de decisão. Exemplos típicos de abordagens clássicas são as conhecidas Support Vector Machines (SVMs), árvores de decisão, florestas aleatórias ou outras técnicas de aprendizado de máquina não conexionistas para classificar os dados de entrada. Esses métodos são ditos artesanais, uma vez que o especialista/pesquisador geralmente projeta o método de extração de características de acordo com a particularidade do problema. Embora esse projeto de abordagem sob medida possa produzir um método específico muito limitado a um subconjunto de dados de entrada (usados para projetar o modelo do algoritmo), ele geralmente requer menos amostras de entrada. Alguns exemplos das abordagens clássicas são:

- O método proposto por Happy e Routray [17] realiza a reorganização da expressão extraindo características de manchas faciais selecionadas ao redor da face enquanto utiliza um SVM como classificador. Os autores alcançaram 94,09% e 91,8% de acertos com esse sistema nos conjuntos de dados CK+ e JAFFE.
- Usando uma combinação dos recursos FAST orientado e BRIEF rotacionado (ORB) e os Padrões Binários Locais (LBP) para extrair partes de expressões faciais , Ben Niu e colaboradores [18] propuseram uma abordagem sem CNN, alcançando uma precisão de 88,5%, 93,2% e 79,8% para os conjuntos de dados JAFEE, CK+ e MMI, respectivamente.
- Abdulrahman e Eleyan [19] também propuseram uma abordagem FER baseada em os algoritmos LBP e Análise de Componentes Principais (PCA) usando um classificador SVM. Eles realizaram vários testes experimentais com o JAFFE e seu recém-introduzido banco de dados MUFE (Mevlana University Facial Expression), obtendo resultados médios de 87% e 77%, respectivamente.
- [20], um novo método de extração é proposta com base em uma abordagem geométrica, onde seis distâncias são calculadas para medir as diferentes partes da face que melhor descrevem uma expressão facial, e uma árvore de decisão é aplicada às bases de dados JAFEE e COHEN. Obtiveram 89,20% e 90,61% de reconhecimento para cada base de dados.
- [21] propuseram uma nova estrutura para o FER ao reconhecer AUs de sequências de imagens usando classificadores de floresta aleatórios duplos. Eles alcançaram uma precisão de 96,38% para o banco de dados CK+.

Com o surgimento das CNNs, várias aplicações foram propostas com esta técnica em mente como um solucionador de problemas gerais. Seu principal aspecto é sua capacidade de generalizar problemas usando um método genérico de extração de recursos, abrindo caminho para

aprender as características mais relevantes diretamente dos dados de entrada e usando-os em um classificador de rede neural. No entanto, o treinamento de uma CNN requer grandes dados de entrada e tempo computacional considerável para fornecer classificadores robustos e precisos. Alguns trabalhos usando a contraparte da CNN são detalhados abaixo:

- [10] calcula o movimento de uma máscara facial para cada emoção do conjunto de dados de treinamento e usa os resultados para treinar um estimador de máscara facial. As imagens a serem reconhecidas são combinadas com sua máscara facial (obtida pela passagem da imagem pelo estimador) e o classificador para transmitir a emoção esperada. Os resultados obtidos mostraram uma acurácia de 98,06%, 82,74% e 61,52% para os conjuntos de dados CK+, MMI e AffectNet.
- [22] introduziram uma rede de menos de 1MB com parâmetros de 65K capaz de executar a 1851 quadros por segundo em uma CPU Intel i7 para leve e velocidade. No entanto, isso veio com uma penalidade de precisão, alcançando 84,8% no conjunto de dados CK+.
- [23] propuseram uma CNN profunda com uma extração paralela de feições block (*FeatEx* ), inspirado na rede GoogleNet, como responsável central. Esse bloco processa os dados de entrada em dois caminhos paralelos com tamanhos de filtro diferentes para capturar melhor as escalas variadas de uma face dentro de um imagem. Concatenando dois *blocos FeatEx* e passando o resultado para um clas-sifier, eles obtiveram 99,6% e 98,63% de precisão nos conjuntos de dados CK+ e MMI, respectivamente.
- [24] criaram o método recente das redes de atenção. um sistema de ponta a ponta capaz de reconhecer expressões faciais a partir de vídeos. O reconhecimento consiste em três etapas distintas: (i) pré-processamento do quadro (alinhamento facial e outros); (ii) extração de características; e (iii) classificação, que é o mesmo procedimento utilizado no método proposto neste trabalho. As acurácias alcançadas por essa rede para os conjuntos de dados CK+ e AFEW 8.0 foram de 99,69% e 51,18%.
- Minaee e Abdolrashidi [25] também usaram a atenção CNNs para criar um fim-sistema final que aprende as características relevantes e se concentra nas partes essenciais de uma expressão facial. O modelo foi treinado utilizando as bases de dados FER2013, CK+, JAFFE e FERG , alcançando acurácias de 70,02%, 99,3%, 92,8% e 98,0%, respectivamente.
- O trabalho apresentado por Kai Wang [11] afirma que a dificuldade em reconhecer As expressões faciais devem-se à ambiguidade, às imagens faciais de baixa qualidade e à subjetividade dos anotadores. A solução proposta pelo autor é baseada na *Self-Cure*

Network (SCN), uma rede simples, mas eficiente, baseada nas CNNs tradicionais. De acordo com o autor, essa abordagem pode suprimir incertezas e evitar que as CNNs se ajustem excessivamente a imagens faciais ambíguas. Eles obtiveram uma acurácia de 88,14% no RAF-DB, 60,23% no AffectNet e 89,35% no FERPlus.

Embora as técnicas de aprendizado profundo possam ser usadas para ter um sistema de ponta a ponta que aprende tudo, desde os recursos até a classificação, essa nem sempre pode ser a melhor solução. O uso de recursos tradicionais combinados com redes neu- rais pode produzir resultados ainda melhores em alguns casos. O recente ressurgimento no campo é dono disso, dados os avanços nas capacidades de computação e a quebra repetida de recordes usando métodos antigos (por exemplo, CNNs) e novos (por exemplo, transformadores). Essas técnicas podem ser usadas não apenas para criar o classificador, mas também para aprender as características mais relevantes.

Com base nos trabalhos encontrados na literatura, também resumidos na Tabela 1, é possível perceber que os problemas devem girar em torno da dificuldade de reconhecer a diversidade e ambiguidade da face nas emoções faciais. Em nossa abordagem, demonstramos que o problema não está relacionado à complexidade da rede neural em si, mas à qualidade dos dados de entrada passados a ela. Desenvolvemos uma abordagem híbrida cujas características são baseadas em uma técnica de extração geométrica robusta que resolve as limitações apresentadas pelos autores e fornece uma abordagem rápida e prática no contexto do REF, independentemente de variações étnicas, raciais, rugas, cicatrizes, barbas ou óculos. Além disso, o método proposto é eficaz contra emoções ambíguas ou indefinidas, como será discutido na seção 4.

Tabela 1 Comparação dos métodos de RAF encontrados na literatura.

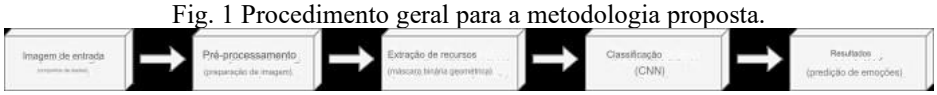
| Autor | Ano | Método | Base de dados | Geral |
|---|---|---|---|---|
| Feliz e Routray | 2014 | SVM | CK+<br>JAFFE | 94.09%<br>91.80% |
| Burkert et al. | 2015 | CNN | CK+<br>MMI | 99.60%<br>98.63% |
| Cugu et al. | 2017 | Viola-Jones e CNN | Oulu-CASIA<br>CK+ | 62.69%<br>84.80% |
| Meng et al. | 2019 | CNN | CK+<br>AFEW 8.0 | 99.69%<br>51.18% |
| Chen et al. | 2019 | CNN | CK+<br>MMI AffectNet | 98.06%<br>82.74%<br>61.52% |
| Minace e Abdolrashidi | 2019 | CNN | FER2013<br>CK+<br>JAFFE<br>FERG | 70.02%<br>99.30%<br>92.80%<br>98.00% |
| Kai Wang e outros | 2020 | SCN | RAF-DB<br>AffectNet<br>FERPlus | 88.14%<br>60.23%<br>89.35% |
| Ben Niu e outros | 2021 | LBP e ORBE | JAFFE<br>CK+<br>MMI | 88.50%<br>93.20%<br>79.80% |

## 3 METODOLOGIA

A metodologia descrita neste trabalho pode ser considerada como uma abordagem híbrida, uma vez que uma técnica de Extração de Traços Geométricos é usada para obter as características mais relevantes a partir de imagens faciais. Essas características são então passadas por um classificador da CNN projetado especificamente para classificar as expressões faciais humanas em categorias predeterminadas de emoções. O objetivo de usar uma solução híbrida é aproveitar sua capacidade de prever rapidamente os resultados de um feed de vídeo em tempo real e continuamente.

Um procedimento geral para a metodologia proposta é ilustrado na Figura 1º. Primeiro, uma técnica de aumento de dados é aplicada ao conjunto de dados de entrada para aumentar o número de amostras e fornecer um conjunto de dados robusto para treinar o modelo da CNN, evitando o efeito de sobreajuste. Em seguida, uma etapa de pré-processamento é introduzida para padronizar o tamanho e o alinhamento das faces apresentadas nas imagens. A partir daí, uma técnica de Extração de Características é empregada para obter um conjunto de características geométricas da imagem facial e convertê-las em uma máscara binária. Finalmente, um algoritmo é usado no modelo de rede neural convolucional para prever expressões faciais. Essas etapas são descritas em detalhes nas subseções a seguir.

Fig. 1 Procedimento geral para a metodologia proposta.




Além disso, a implementação do código é baseada na linguagem Python3 [26] devido à sua robustez para prototipagem e testes, incluindo bibliotecas para suportar o método apresentado, como *OpenCV* [27], *Keras* [28], *TensorFlow* (versão GPU) [29] e *Dlib* [30].

## 3.1 IMAGEM DE ENTRADA

A técnica de aumento de dados aplicada a cada imagem de entrada consistiu em girar uma imagem em um ângulo aleatório (entre 10 e 30 graus) nos sentidos horário e anti-horário, produzindo três imagens espelhadas posteriormente e aumentando em seis vezes o tamanho original do conjunto de dados. Os conjuntos de dados de emoções faciais utilizados para validar a metodologia proposta são descritos a seguir:

- JAFFE – O *conjunto de dados* de Expressão Facial Feminina Japonesa (JAFFE) é uma fonte de expressão facial clássica de 1998 [13]. São 253 imagens de 10 modelos japonesas que expressam sete emoções comuns: neu- tral, raiva, nojo, medo, felicidade, tristeza e surpresa. As imagens são em tons de cinza e têm uma resolução de 256x256 pixels. Embora seja um conjunto de dados sim- ple caracterizado por apenas algumas imagens, tem sido amplamente utilizado academicamente em todo o mundo.
- O RaFD – The Radbound Faces Database (RaFD), apresentado em 2010, é composto por um conjunto de imagens de 67 modelos instruídos a mostrar oito expressões diferentes (as sete mostradas no JAFFE mais outra por desprezo) e a olhar em três direções (direita, à frente e à esquerda) [15]. Para cada exposição, as fotografias dos sujeitos correspondem a 5 ângulos de câmera (-90°, -45°, 0°, 45° e 90°), totalizando 120 imagens por modelo. Este artigo utilizou apenas as imagens frontais (0°) para as fases de treinamento e teste.
- CK+ – O *conjunto de dados Extended Cohn-Kanade* (CK+) é uma extensão apresentada em 2010 como um substituto para o já popular na época Cohn-Kanade dataset (CK) lançado em 2000 [14]. É composto por 593 imagens de 123 sujeitos mostrando oito emoções (o mesmo que para RaFD) com uma resolução de imagem espacial de 640x490 pixels e rotulado com o Facial Action Coding System (FACS), além da classificação da expressão facial.

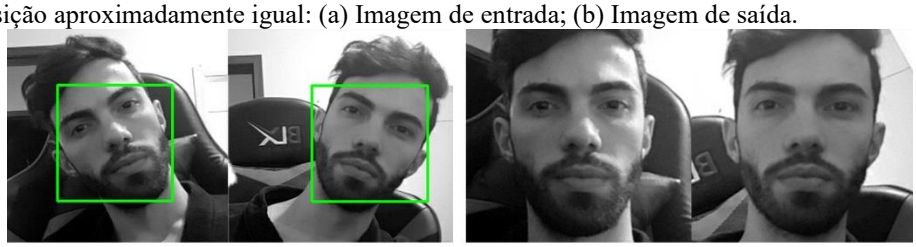
3.2 PRÉ-PROCESSAMENTO

Um estágio de pré-processamento simples é introduzido para diminuir a dimensionalidade e fornecer um conjunto de imagens regulares para a fase de extração de recursos. O resultado desse procedimento é demonstrado na Figura 2, composta principalmente por:

Simplificação de cores – antes de qualquer extração, a imagem de entrada é convertida em escala de cinza, pois essa escala de cores tem pouca significância na expressão facial, e a variação de brilho é suficiente para detectar as características;

- Regularização facial – em seguida, é realizado um processo de centralização da face utilizando as coordenadas do centro do nariz como ponto de referência de alinhamento;
- Reposicionamento facial – uma rotação rígida é então realizada para apresentar os olhos em uma linha horizontal, deixando ambos os olhos no mesmo nível do eixo y e garantindo imagens alinhadas idênticas para o conjunto de dados;
- Redução da resolução espacial – finalmente, a imagem de entrada é redimensionada para normalizar a distância da câmera (que faz com que o tamanho do rosto na imagem aumente ou diminua) e para garantir que os tamanhos dos rostos sejam equivalentes.

Fig. 2 Independentemente do ângulo da face em relação à câmera, o algoritmo pode alinhar, centralizar e redimensionar a face para uma posição aproximadamente igual: (a) Imagem de entrada; (b) Imagem de saída.



(a) (b)

Essas etapas foram implementadas com a ajuda da biblioteca *Dlib*, um kit de ferramentas projetado explicitamente para aprendizado de máquina, processamento de imagens e aplicações de álgebra linear. Inicialmente, utiliza-se a função de detector facial [31], que detecta uma face baseada no método HOG e em uma SVM linear, seguida de um algoritmo proprietário para calcular os ajustes e alinhamentos necessários da imagem. O funcionamento desse algoritmo é detalhado na subseção a seguir.

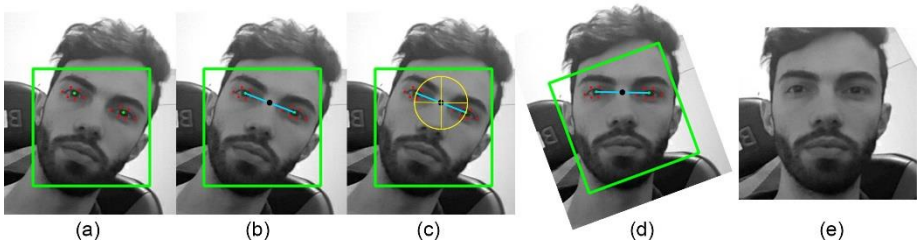
### 3.2.1 Algoritmo de alinhamento

A forma mais adequada de alinhar a face na imagem corresponde ao posicionamento dos olhos em um registro no plano horizontal. Para isso, os centroides de ambos os olhos são inicialmente calculados considerando os pontos médios de cada olho ($x$ e $y$) e dividindo por sua cardinalidade ($n$). Essa etapa é demonstrada na Figura 3a. Em seguida, a diferença entre

os olhos esquerdo e direito (Δ) é calculada nas direções *x* e *y*. A partir daí, obtém-se o ponto central dos olhos, como mostra a Figura 3b. Com base nessas informações, o ângulo necessário para girar a imagem e garantir que o eixo *y* de ambos os olhos esteja localizado na mesma posição é obtido calculando-se a tangente de arco para os valores de Δ nas direções *x* e *y*. O resultado dessa etapa está demonstrado na Figura 3c. Finalmente, para manter um conjunto de dados padronizado, a imagem facial é redimensionada, garantindo que todas as imagens compartilhem o mesmo tamanho e, portanto, o mesmo tamanho de máscara de entrada necessário para a rede neural. Esse estágio é alcançado utilizando-se a porção da distância entre os olhos (Δx e Δy) e calculando-se um fator de escala.

Além disso, a rotação da imagem é realizada através da obtenção da matriz de transformação (*M*) com a ajuda do getRotationMatrix2D e warpAffine funções da biblioteca *OpenCV*. O resultado e a saída final do algoritmo de alinhamento são mostrados nas Figuras 3d e 3e, respectivamente.

Fig. 3 Fluxo de trabalho do algoritmo de alinhamento de pré-processamento: (a) Localização do centroide de cada olho mostrado em pontos verdes; (b) Distância entre os centroides dos dois olhos mostrada em uma linha azul;

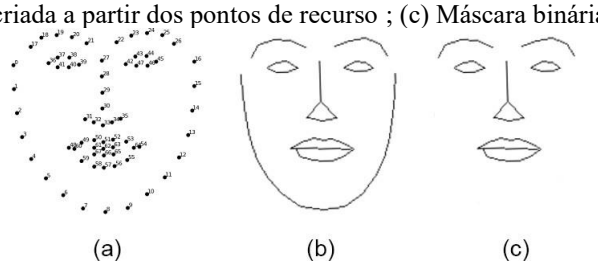



(c) Ângulo de rotação da imagem mostrado em um círculo amarelo; (d) Imagem da face resultante obtida pelo algoritmo de alinhamento (girada para o ângulo computado); (e) Saída final do algoritmo de alinhamento (alinhado e dimensionado em um tamanho padrão).

## 3.3 EXTRAÇÃO DE FEIÇÕES GEOMÉTRICAS

A técnica de Extração de Características Geométricas descrita nesta seção considera a linha do maxilar, boca, olhos, sobrancelhas e nariz como as características mais relevantes durante a extração de expressões faciais. Essas características são extraídas da imagem pré-processada com o auxílio da função preditora de forma, uma *função de biblioteca Dlib* baseada em um conjunto de árvores de regressão pré-treinadas capazes de estimar a localização de 68 pontos de coordenadas necessários para o mapeamento de estruturas faciais [32]. Uma visão geral dessa técnica é apresentada na Figura 4.

Fig. 4 Visão geral da técnica de Extração de Feço Geométrico: (a) Pontos de coordenadas capturados pelo extrator de feição geométrica; (b) Máscara binária criada a partir dos pontos de recurso ; (c) Máscara binária sem mandíbula.

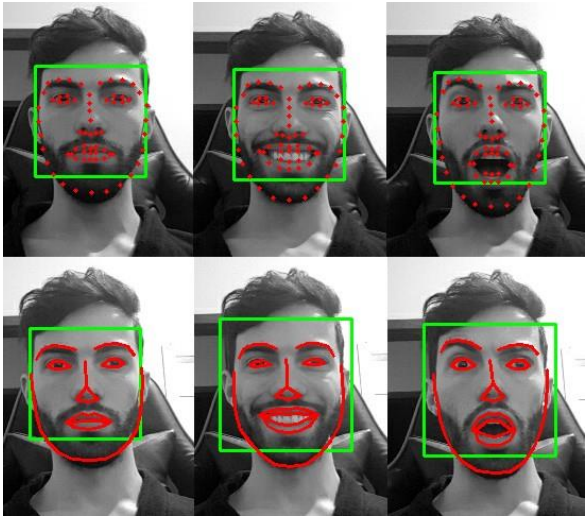



A Figura 4a ilustra os 68 pontos mais representativos utilizados por essa abordagem. Esses pontos são então conectados e transformados em uma máscara binária, Figura 4b, posteriormente usada pelo classificador CNN. Este trabalho também tem como objetivo com- pare a importância e o impacto da linha do maxilar durante o processo de classificação, avaliando uma máscara binária sem linha do maxilar, como ilustrado na Figura 4c.

A partir da imagem com as características faciais, obtém-se um novo quadro com uma máscara binária correspondente à coloração absoluta branca e preta, subtraindo-se a imagem pré-desenhada. Como resultado, para uma imagem com um canal e profundidade de cor de 8 bits, apenas valores de 0 (preto) e 255 (branco) sem valores intermediários de cinza podem ser obtidos, simplificando o conjunto de recursos adquiridos.

Uma aplicação da técnica de Extração de Traços Geométricos descrita é demonstrada na Figura 5, onde são destacados os 68 pontos coordenados detectados e os contornos dos olhos, nariz, boca, sobrancelhas e linha do maxilar. As características delimitadas pelas linhas vermelhas, desenhadas sobre a imagem original, são posteriormente processadas com a função de subtração da imagem para criar a máscara binária. Além disso, as marcas quadradas verdes são usadas apenas para reconhecimento facial e são ignoradas ao aplicar a função de subtração.

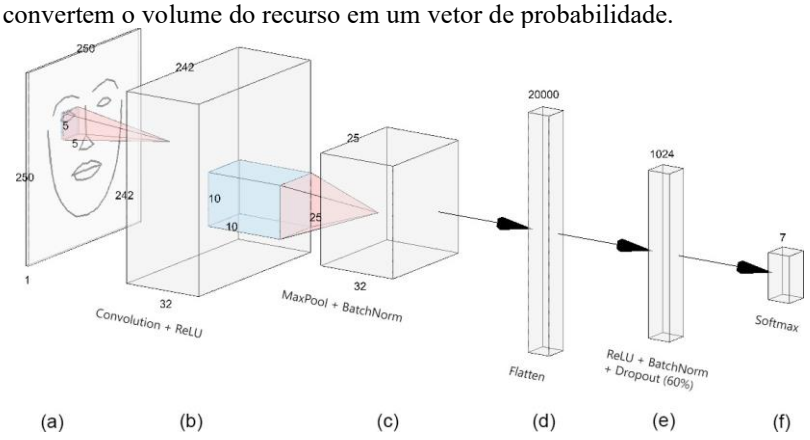
Fig. 5 Aplicação da técnica de Extração de Características Geométricas para três exposições distintas: neutro, felicidade e surpresa, respectivamente.



## 3.4 REDE NEURAL CONVOLUCIONAL

A configuração e os parâmetros da arquitetura CNN proposta estão ilustrados na Figura 6. A camada de entrada, como mostrado na Figura 6a, compreende uma imagem binária das características faciais mais representativas e cobre um domínio planar de 250x250 pixels. A Figura 6b mostra a camada de função de ativação da Unidade Linear Retificada (ReLU) com dimensões de rede de 32x25x25 e ativação de 10x10x25. A Figura 6c exibe a camada MaxPool+BatchNorm com dimensões de rede de 25x25x32. A Figura 6d apresenta a camada Flatten, que remodela o tensor para uma única dimensão e produz 20000 unidades. Por fim, as Figuras 6e 6f mostram as camadas 60% de abandono e saída de decisão (*Softmax* ), que são discutidas em detalhes na subseção subsequente.

Figo. 6 Configuração da arquitetura CCN proposta – A máscara binária é inserida em uma camada convolucional para extrair as características obtidas, seguida por uma camada max-pooling para reduzir a dimensionalidade. As camadas totalmente conectadas convertem o volume do recurso em um vetor de probabilidade.




Além disso, detalhes de implementação sobre a CNN usada na abordagem apresentada também são demonstrados pelo código-fonte na Figura 7, que descreve a sequência usada para construir a CNN e os valores dos parâmetros usados como configuração de entrada.

Fig. 7 Código fonte para o modelo CNN proposto, incluindo seu pipelining.

```
def cnn_modelo():
    modelo = Sequential()
    modelo.add(Conv2D(32, (5,5), input_shape=(250, 250, 1), activation='relu'))
    modelo.add(BatchNormalization())
    modelo.add(MaxPooling2D(pool_size=(10, 10), strides=(10, 10), padding='same'))
    modelo.add(Flatten())
    modelo.add(Dense(1024, activation='relu'))
    modelo.add(BatchNormalization())
    modelo.add(Dropout(0.6))
    modelo.add(Dense(7, activation='softmax'))
    sgd = optimizers.SGD(lr=1e-3)
    modelo.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
    callbacks_list = ModelCheckpoint("modelo/expressao.h5", monitor='val_accuracy',
                                     verbose=1, save_best_only=True, mode='auto')
    return modelo, callbacks_list
```

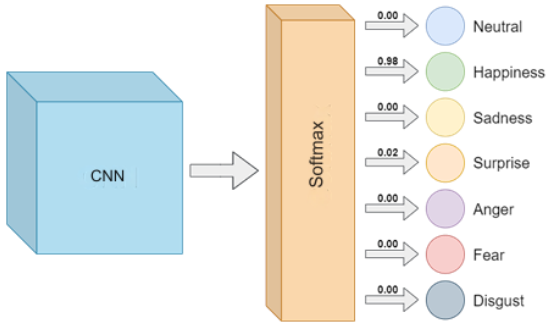
## 3.5 CLASSIFICADOR DE EXPRESSÃO

A etapa final da metodologia proposta está relacionada à expressão clas- sifier layer. Essa camada segue a arquitetura introduzida na Figura 6f, onde a função ReLU ativa a rede e a primeira camada densa, seguida por um procedimento de 60% de abandono e normalização em lote (BatchNorm). A camada final totalmente conectada emprega então a ativação *Softmax* para construir uma distribuição de probabilidade associada a uma das sete emoções do conjunto de dados descritas na Seção 3.1.

A razão para o uso da ativação da ReLU é baseada em evidências empíricas autoadquiridas. Esta função de ativação produziu os melhores resultados para o classificador CNN proposto em comparação com outras funções de ativação. [33], a ReLU é a recomendação padrão em relação às funções de ativação nas abordagens modernas de redes neurais, pois é adequadamente projetada para trabalhar com imagens e não satura na região positiva, tendendo a convergir muito mais rápido do que a tangente sigmoide ou hiperbólica nas imagens, além de não ser centrada em zero.

O estágio de abandono é usado para evitar o overfitting, uma vez que os nós individuais são removidos da rede com uma probabilidade de 60%, mantendo uma rede reduzida e forçando os nós de uma camada (neurônio) a adaptar e corrigir erros das camadas anteriores, tornando o modelo mais robusto e confiável. Além disso, o procedimento BatchNorm normaliza a ativação da camada anterior, estabilizando e acelerando a rede neural enquanto melhora o erro de generalização. Finalmente, a função Softmax calcula a probabilidade de cada expressão de entrada, e a soma das probabilidades para cada uma das sete expressões resulta em um valor de 1. Conforme ilustrado na Figura 8, a expressão com maior valor de probabilidade é a emoção resultante.

Fig. 8 A função Softmax resulta na probabilidade de uma dada expressão facial. Neste exemplo, a CNN informa que a probabilidade de a expressão ser felicidade é de 98%, embora ainda haja uma pequena chance (2%) de ser a expressão de surpresa.



## 4 RESULTADOS EXPERIMENTAIS

### 4.1 VALIDAÇÃO

Validar uma CNN com os mesmos dados do conjunto de treinamento pode ser considerado um erro de procedimento lógico. Afinal, um modelo que verifica as imagens com base em informações prévias obteria um bom feedback de classificação, tornando a rede extremamente precisa. No entanto, se a eficiência do modelo for medida com base em dados aos quais a rede nunca teve acesso, então, nesse caso, o modelo será completamente limitado (ou inútil no pior dos casos) na previsão de novos dados em um conjunto mais amplo ou durante aplicações do mundo real.

Esse problema pode ser resolvido conduzindo uma fase supervisionada de aprendizado de máquina onde o conjunto de dados é dividido em dois conjuntos: (i) treinamento e (ii) validação. Dessa forma, a rede pode usar o primeiro conjunto para treinar suas previsões e o segundo para validar sua precisão, evitando que os dados usados para calcular a precisão sejam vistos por ambos os lados (chamamos isso de dados invisíveis).

Nossa abordagem dividiu cada conjunto de dados em 80% para treinamento e 20% para avaliação durante o estágio inicial de treinamento supervisionado. As etapas de treinamento subsequentes ocorreram por 300 épocas usando um otimizador SGD com uma taxa de aprendizado de 10−3, uma entropia cruzada categórica como uma função de perda e um tamanho de lote de 100. A Tabela 2 mostra os resultados experimentais obtidos pela abordagem apresentada em termos de acurácia em comparação com alguns trabalhos relacionados encontrados na literatura.

Tabela 2 Acurácia obtida pelo método apresentado em comparação com outros trabalhos relacionados utilizando o procedimento de validação simples (sem dobras cruzadas).

| **Método** | **Ano** | **JAFFE** | **RaFD** | **CK+** |
|---|---|---|---|---|
| BDBN | 2014 | 93.00% | | 96.70% |
| Feliz e Routray | 2014 | 91.80% | | 94.09% |
| Hamester et al. | 2015 | 95.80% | | |
| DeXpression | 2015 | | | 99.60% |
| Zavares et al. | 2017 | | 85.79% | 88.58% |
| Mavani et al. | 2017 | | 95.71% | |
| MicroExpNet | 2017 | | | 84.80% |
| TeacherExpNet | 2017 | | | 97.60% |
| VENTILADOR | 2019 | | | 99.69% |
| FMPN | 2019 | | | 98.06% |
| Minaee e Abdolrashidi | 2019 | 92.80% | | 98.00% |
| Kai Wang e outros. | 2020 | | 88.14% | |
| Ben Niu e outros. | 2021 | 88.50% | | 93.20% |
| Nosso método | | 96.83% | 98.58% | 98.46% |
| Nosso método (sem mandíbula) | | 96.03% | 97.28% | 98.57% |

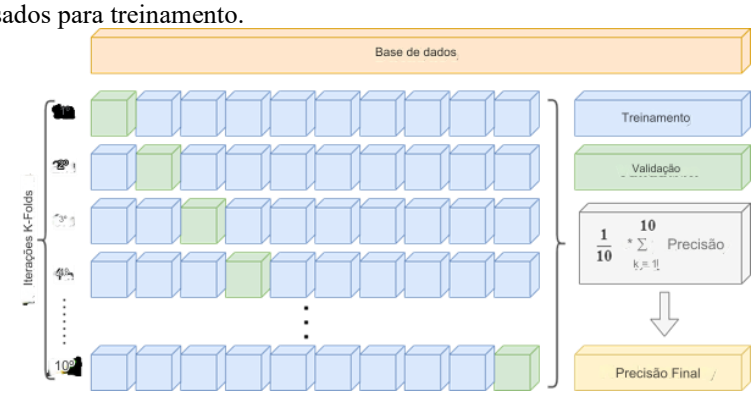
Embora eficaz, esse procedimento pode causar uma falsa sensação de correção de sobreajuste que resulta da volatilidade da seleção do conjunto de validação e da suposição de que o banco de dados não é homogêneo. Além disso, sua efetividade está relacionada à invisibilidade do conjunto de validação para a rede treinada pelo conjunto de capacitação. Portanto, a aleatoriedade aplicada ao conjunto de dados não garante um aumento na precisão, e a realização desse procedimento apenas uma vez não garantiu precisão suficiente para afirmar se a rede analisa dados invisíveis de forma eficiente.

## 4.2 VALIDAÇÃO CRUZADA

Para medir a acurácia da capacidade de predição da rede, foi utilizado o método de validação por dobras cruzadas [34], uma técnica de validação simples e popular baseada no número de grupos ($k$) que um conjunto de dados precisa dividir para evitar resultados enviesados. Assim, cada conjunto de dados foi dividido em dez subconjuntos ($k = 10$), conforme este valor geralmente resulta em previsões com variações modestas e baixa polarização, obtendo subconjuntos de tamanho aproximadamente igual.

Um exemplo do método de validação cruzada é ilustrado na Figura 9, onde os dados são embaralhados aleatoriamente e divididos em dez subconjuntos, e a precisão do modelo é obtida pela média dos valores de precisão de cada iteração, garantindo que o resultado final seja uma estimativa mais precisa na previsão do desempenho de aplicativos do mundo real ou com dados invisíveis.

Fig. 9 Método de validação por dobras cruzadas para k = 10, resultando em uma separação aleatória de 10 subconjuntos de tamanhos aproximadamente iguais. O primeiro subconjunto é usado para validação de rede, enquanto os 9 conjuntos de dados restantes são usados para treinamento.




Os resultados individuais de cada iteração e a acurácia geral da rede para os dois métodos propostos (máscara binária com e sem linha do maxilar) são apresentados na Tabela 3. Estes resultados mostram que a variação entre os dois métodos propostos é insuficiente para afirmar que a linha do maxilar fornece informações valiosas para a rede proposta no reconhecimento da expressão facial.

Tabela 3 Comparação dos resultados da validação cruzada para os dois métodos propostos.

| | **JAFFE** | | **RaFD** | | **CK+** | |
|---|---|---|---|---|---|---|
| Método | Normal | No-Jawline | Normal | No-Jawline | Normal | No-Jawline |
| Dobra 1 | 96.54% | 95.90% | 97.88% | 96.41% | 97.93% | 98.19% |
| Dobra 2 | 97.06% | 96.11% | 98.21% | 97.02% | 97.74% | 97.59% |
| Dobra 3 | 96.92% | 96.17% | 98.07% | 96.87% | 96.91% | 97.02% |
| Dobra 4 | 96.14% | 95.83% | 98.20% | 97.08% | 98.49% | 98.66% |
| Dobrável 5 | 95.36% | 94.75% | 98.36% | 97.27% | 98.58% | 98.71% |
| Dobra 6 | 97.05% | 96.38% | 97.93% | 96.85% | 98.07% | 98.23% |
| Dobra 7 | 97.22% | 96.61% | 98.27% | 97.21% | 96.95% | 97.36% |
| Dobra 8 | 95.49% | 95.74% | 97.97% | 96.86% | 98.39% | 98.54% |
| Dobra 9 | 95.17% | 94.99% | 98.14% | 97.04% | 97.43% | 97.87% |
| Dobra 10 | 96.61% | 96.06% | 97.93% | 96.63% | 97.82% | 97.62% |
| **Média** | **96.36%** | **95.85%** | **98.10%** | **96.92%** | **97.83%** | **97.98%** |

Da mesma forma, a comparação dos resultados finais obtidos entre as técnicas de validação simples e cruzada é apresentada na Tabela 4. Como esperado, o método de validação por dobra cruzada reduziu a acurácia da rede, por ser uma técnica mais conservadora e computacionalmente mais cara.

Tabela 4 Comparação dos resultados obtidos pelas técnicas de validação simples e cruzada.

| **Validação Método** | **JAFFE** | **RaFD** | **CK+** |
|---|---|---|---|
| Simples Normal | 96.83% | 98.58% | 98.46% |
| No-Jawline | 96.03% | 97.28% | 98.57% |
| Dobra cruzada Normal | 96.36% | 98.10% | 97.83% |
| No-Jawline | 95.85% | 96.92% | 97.98% |

## 4.3 TEMPO DE EXECUÇÃO E DESEMPENHO

Nesta subseção, discutimos o desempenho do método em termos de tempo de execução para fornecer uma solução de ponta a ponta em relação à previsão de emoções faciais. Usamos o conjunto de dados JAFFE como entrada para o algoritmo desenvolvido enquanto executamos uma técnica de aumento de dados para alcançar os dados necessários em alguns testes de desempenho, aumentando exponencialmente as entradas de 1 para 10000 imagens.

Uma comparação do tempo total necessário para o algoritmo de ponta a ponta (incluindo a etapa de pré-processamento da imagem) e o algoritmo de previsão (ignorando a etapa de pré-processamento) é mostrada na Tabela 5. Além disso, para fins de visualização, o tempo total de cada iteração também é mostrado em FPS (quadros por segundo) e PPS (previsão por segundo).

Tabela 5 Teste de desempenho do algoritmo baseado no tempo de processamento usando uma variação de dados de entrada.

| | **Tempo total de processamento** | | **Tempo de processamento por imagem** | | **Métricas** | **por segundo** |
|---|---|---|---|---|---|---|
| Imagens | De ponta a ponta | Predição | De ponta a ponta | Predição | FPS | PPS |
| e0 | 1s 077ms | 1s 065ms | 1s 77ms | 1s 65ms | 0.92 | 0.93 |
| E1 | 1s 309ms | 1s 189ms | 130ms | 118 milímetros | 7.63 | 8.40 |
| e2 | 3s 737ms | 2s 505ms | 37ms | 25ms | 26.75 | 39.91 |
| e3 | 27s 374ms | 15s 291ms | 27ms | 15ms | 36.53 | 65.39 |
| e4 | 268s 808ms | 143s 789ms | 26ms | 14ms | 37.20 | 69.54 |

A configuração de instalação utilizada para os testes de desempenho foi composta por um computador desktop com sistema operacional Windows 10 Pro (versão 20H2), um processador AMD Ryzen 7 3700X (8 núcleos com 3,59 GHz por núcleo), 32 GB de RAM (3200 MHz), uma placa de vídeo NVIDIA GeForce RTX 2070 (8 GB de memória de vídeo dedicada) e um SSD Samsung 970 EVO Plus M.2 NVMe.

O uso da CPU durante as visualizações de imagem variou de 7,8% a 9,1%, enquanto o uso da GPU variou de 0,3% a 1,1%. Por outro lado, o uso de RAM foi de no máximo 2GB, uma vez que a memória de vídeo dedicada foi totalmente utilizada. É importante considerar que esses resultados foram observados apenas para o processo em que as emoções foram previstas, enquanto o sistema e os processos secundários foram ignorados.

Esses resultados mostram que um gargalo é criado durante o carregamento da imagem e que os processos da GPU no lote são executados mais rapidamente do que os da CPU. Uma possível solução para esse problema (não adotada em nossa abordagem) é baseada na distribuição do carregamento da imagem para um novo thread para realizar o pré-processamento, fazendo com que a GPU aguarde mais dados durante o tempo ocioso e mitigando o maior uso da CPU em comparação com o uso da GPU para a configuração apresentada.

Como a CPU carrega mais imagens de uma só vez (evitando redundância de operações), ela faz com que a GPU acesse uma quantidade maior de dados, reduzindo a ociosidade e, consequentemente, aumentando o desempenho do algoritmo. No entanto, ainda há um gargalo de memória para a GPU, sendo necessário aguardar o carregamento e descarregamento de novos dados da memória dedicada da GPU.

Os resultados mostram que a solução proposta pode ser executada em tempo real, proporcionando continuamente a tomada de decisão da imagem facial com sua emoção correspondente. Em um feed de vídeo contínuo fornecido por uma webcam limitada a 30 fps, o algoritmo foi capaz de processar 27 fps, onde a perda de fps poderia estar relacionada à latência entre a entrada da webcam, dados de carregamento e separação de quadros. Este passo adicional é necessário para obter a imagem de entrada para a rede end-to-end enquanto se realiza uma

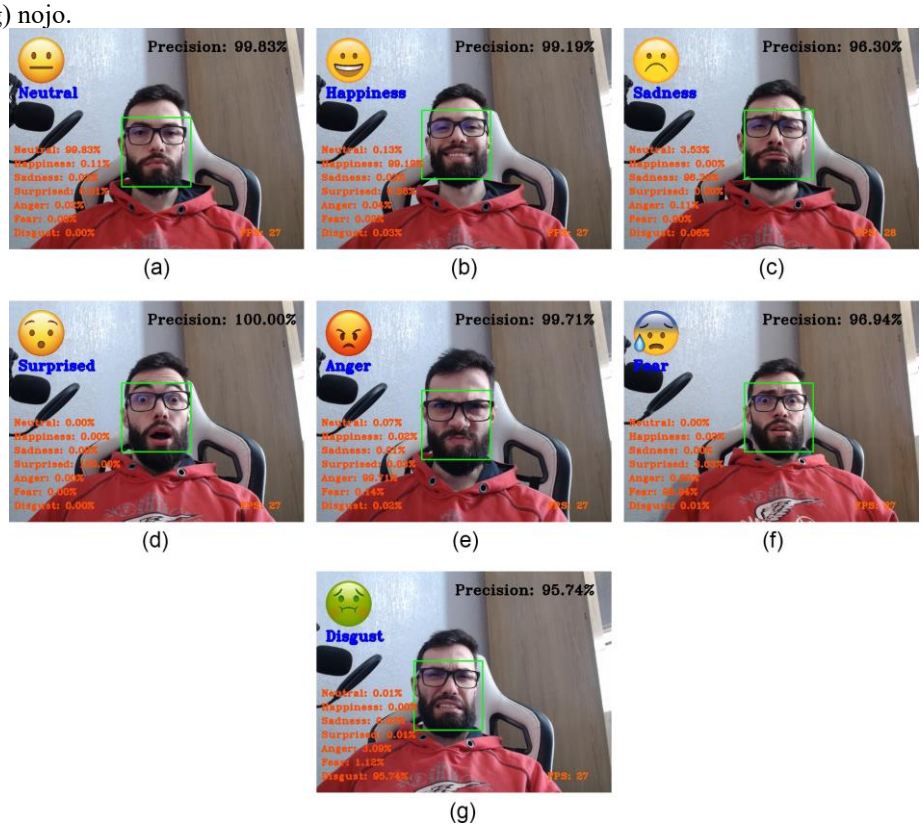
operação extra que não foi necessária para o teste realizado com o banco de dados JAFFE, pois a imagem já estava capturada, sendo necessário apenas carregar a imagem na memória. A aplicação de a abordagem proposta para uma sequência de vídeo é demonstrada na Figura 10.

## 5 CONSIDERAÇÕES FINAIS

Este trabalho apresentou uma abordagem computacional em tempo real para reconhecimento de imagens faciais e emoções de vídeo baseada em uma técnica de extração de características geométricas associada a uma CNN. Os resultados experimentais foram comparados com métodos de última geração, fornecendo resultados marginalmente melhores para um conjunto de dados misto. As taxas de acurácia utilizando validação simples foram de 96,83%, 98,58% e 98,57% para os conjuntos de dados JAFFE, RaFD e CK+. Por outro lado, por meio de validação cruzada, obtivemos taxas de acurácia de 96,36%, 98,10% e 97,98% para os mesmos conjuntos de dados, indicando que a abordagem apresentada pode ser considerada uma solução promissora para FER, especialmente para aplicações em tempo real.

Nosso método oferece resultados competitivos nos três conjuntos de dados testados, obtendo maiores precisões para os conjuntos de dados JAFFE e RaFD. Usando validação de dobras cruzadas, podemos comparar a precisão entre o método normal e não-maxilar, cujas conclusões são que o uso da linha do maxilar na máscara binária criada pode ajudar a rede a ter um ponto de referência adicional com os outros pontos de referência faciais. No entanto, não influencia diretamente na expressão realizada. Escolhendo usar ou não esse ponto de referência como entrada para a rede

Fig. 10 Resultados das emoções faciais para a metodologia proposta: a) neutro, b) felicidade, c) tristeza, d) surpresa, e) raiva, f) medo e g) nojo.




É uma decisão relativamente insignificante em termos de precisão. Ao compararmos os resultados obtidos com a validação simples e cruzada, observamos uma diminuição da acurácia em uma faixa de aproximadamente 1%, uma vez que o processo de validação por dobra cruzada tende a ser mais conservador com seus percentuais. Assim, reforça a tese de que os resultados obtidos com esse método não são consequência do efeito overfitting.

Em termos de tempo de execução, a velocidade de desempenho do método apresentado também é encorajadora, executando-o em tempo real com precisão suficiente, abrindo caminho para muitas aplicações do mundo real. Trabalhos futuros consideram a implementação de algum mecanismo capaz de mitigar a invariância temporal do sistema. Tomamos cada quadro como sua entidade separada, sem considerar sua relação com os anteriores. Isso pode causar algum efeito de cintilação durante as previsões, e levando em conta informações temporais relacionadas aos últimos quadros podem ser usadas para fornecer uma classificação e previsão de suavização.

## DECLARAÇÕES

Conflito de interesses Todos os autores certificam que não têm afiliações ou envolvimento em qualquer organização ou entidade com qualquer interesse financeiro ou não financeiro no assunto ou materiais discutidos neste manuscrito.

## DISPONIBILIDADE DE DADOS

O código gerado durante e/ou analisado durante o presente estudo está disponível no repositório do GitHub https://github.com/gustavogino/ Facial-Expression-Recognition. Além disso, um vídeo demonstrativo de um aplicativo em tempo real pode ser encontrado em https://www.youtube.com/watch?v= fFOldbHtHQU.

## REFERÊNCIAS


Mellouk, W., Handouzi, W.: Facial emotion recognition using deep learning: review and insights. Procedia Computer Science 175, 689–694 (2020)

Ekman, P., Friesen, W.V.: Constants across cultures in the face and emotion. Journal of personality and social psychology 17(2), 124 (1971)

Orgeta, V.: Effects of age and task difficulty on recognition of facial affect. The journals of gerontology. Series B, Psychological sciences and social sciences 65B, 323–7 (2010). https://doi.org/10.1093/geronb/gbq007

Jung, H., Lee, S., Yim, J., Park, S., Kim, J.: Joint fine-tuning in deep neu- ral networks for facial expression recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2983–2991 (2015)

Sun, N., Li, Q., Huan, R., Liu, J., Han, G.: Deep spatial-temporal fea-ture fusion for facial expression recognition in static images. Pattern Recognition Letters 119, 49–61 (2019)

Dhall, A., Asthana, A., Goecke, R., Gedeon, T.: Emotion recognition using phog and lpq features. In: Face and Gesture 2011, pp. 878–883 (2011). IEEE

Ojansivu, V., Heikkilä, J.: Blur insensitive texture classification using local phase quantization. In: International Conference on Image and Signal Processing, pp. 236–243 (2008). Springer

Zhang, Z.: Feature-based facial expression recognition: Sensitivity analysis and experiments with a multilayer perceptron. International Journal of Pattern Recognition and Artificial Intelligence 13(06), 893–911 (1999)

Barsoum, E., Zhang, C., Ferrer, C.C., Zhang, Z.: Training deep networks for facial expression recognition with crowd-sourced label distribution. In: Proceedings of the 18th ACM International Conference on Multimodal Interaction, pp. 279–283 (2016)

Chen, Y., Wang, J., Chen, S., Shi, Z., Cai, J.: Facial motion prior networks for facial expression recognition. In: 2019 IEEE Visual Communications and Image Processing (VCIP), pp. 1–4 (2019). IEEE

Wang, K., Peng, X., Yang, J., Lu, S., Qiao, Y.: Suppressing uncertain- ties for large-scale facial expression recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

Sakai, T., Nagao, M., Fujibayashi, S.: Line extraction and pattern detection in a photograph. Pattern recognition 1(3), 233–248 (1969)

Lyons, M., Kamachi, M., Gyoba, J.: The Japanese Female Facial Expres- sion (JAFFE) Dataset (1998)

Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In: 2010 Ieee Computer Society Conference on Computer Vision and Pattern Recognition-workshops, pp. 94–101 (2010). IEEE

Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D.H., Hawk, S.T., Van Knippenberg, A.: Presentation and validation of the radboud faces database. Cognition and emotion 24(8), 1377–1388 (2010)

Goodfellow, I.J., Erhan, D., Carrier, P.L., Courville, A., Mirza, M., Ham- ner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.-H., *et al.*: Challenges in representation learning: A report on three machine learning con- tests. In: International Conference on Neural Information Processing, pp. 117–124 (2013). Springer

Happy, S., Routray, A.: Automatic facial expression recognition using fea- tures of salient facial patches. IEEE transactions on Affective Computing 6(1), 1–12 (2014)

Niu, B., Gao, Z., Guo, B.: Facial expression recognition with lbp and orb features. Computational Intelligence and Neuroscience 2021 (2021)

Abdulrahman, M., Eleyan, A.: Facial expression recognition using sup- port vector machines. 2015 23rd Signal Processing and Communications Applications Conference, SIU 2015 - Proceedings, 276–279 (2015). https: //doi.org/10.1109/SIU.2015.7129813

Salmam, F.Z., Madani, A., Kissi, M.: Facial expression recognition using decision trees. Proceedings - Computer Graphics, Imaging and Visual- ization: New Techniques and Trends, CGiV 2016, 125–130 (2016). https: //doi.org/10.1109/CGIV.2016.33

Pu, X., Fan, K., Chen, X., Ji, L., Zhou, Z.: Facial expression recognition from image sequences using twofold random forest classifier. Neuro- computing 168, 1173–1180 (2015). https://doi.org/10.1016/J.NEUCOM. 2015.05.005

İlke Çuğu, Şener, E., Akbaş, E.: Microexpnet: An extremely small and fast model for expression recognition from face images (2017) arXiv:1711.07011 [cs.CV]

Burkert, P., Trier, F., Afzal, M.Z., Dengel, A., Liwicki, M.: Dexpres- sion: Deep convolutional neural network for expression recognition. arXiv preprint arXiv:1509.05371 (2015)

Meng, D., Peng, X., Wang, K., Qiao, Y.: Frame attention networks for facial expression recognition in videos. In: 2019 IEEE International Conference on Image Processing (ICIP), pp. 3866–3870 (2019). IEEE

Minaee, S., Abdolrashidi, A.: Deep-emotion: Facial expression recognition using attentional convolutional network. arXiv preprint arXiv:1902.01019 (2019)

Van Rossum, G., Drake, F.L.: Python 3 Reference Manual. CreateSpace, Scotts Valley, CA (2009)

Itseez: Open Source Computer Vision Library. GitHub (2015). https:// github.com/itseez/opencv Accessed 2021-06-26

Chollet, F., et al.: Keras. GitHub (2015). https://github.com/fchollet/ keras Accessed 2021-06-26

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin,M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: Tensorflow: A system for large-scale machine learning. In: 12th USENIX Symposium on Operat- ing Systems Design and Implementation (OSDI 16), Savannah, GA, pp. 265–283 (2016)

King, D.E.: Dlib-ml: A Machine Learning Toolkit vol. 10, pp. 1755–1758 (2009)

King, D.: Face Detection with Python using OpenCV. http://dlib.net/ face detector.py.html. Accessed on: March 5, 2023 (2015)

Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensem-ble of regression trees. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 1867–1874 (2014)

Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge, Massachusetts (2016). http://www.deeplearningbook.org

Refaeilzadeh, P., Tang, L., Liu, H.: In: LIU, L., ÖZSU, M.T. (eds.) Cross- Validation, pp. 532–538. Springer, Boston, MA (2009). https://doi.org/ 10.1007/978-0-387-39940-9 565