UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LAURA MARTINS DA COSTA COURA MARINHO
Orientador: Saul Emanuel Delabrida Silva
Coorientador: Andrea Gomes Campos Bianchi

# EDUCAAFETO: RECONHECIMENTO DE EMOÇÕES NA AVALIAÇÃO DE SISTEMAS INTERATIVOS DE APRENDIZAGEM

Ouro Preto, MG
2023

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

LAURA MARTINS DA COSTA COURA MARINHO

# EDUCAAFETO:
# RECONHECIMENTO DE EMOÇÕES NA AVALIAÇÃO DE SISTEMAS INTERATIVOS DE APRENDIZAGEM

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

**Orientador:** Saul Emanuel Delabrida Silva
**Coorientador:** Andrea Gomes Campos Bianchi

Ouro Preto, MG
2023

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

M338e Marinho, Laura Martins Da Costa Coura.
EducaAfeto [manuscrito]: Reconhecimento de emoções na avaliação de sistemas interativos de aprendizagem. / Laura Martins Da Costa Coura Marinho. - 2023.
42 f.: il.: color., tab..

Orientador: Prof. Dr. Saul Emanuel Delabrida Silva.
Coorientadora: Profa. Dra. Andrea Gomes Campos Bianchi.
Monografia (Bacharelado). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Biológicas. Graduação em Ciência da Computação .

1. Redes Neurais de Computação. 2. Jogos educativos. 3. Reconhecimento de emoções. I. Silva, Saul Emanuel Delabrida. II. Bianchi, Andrea Gomes Campos. III. Universidade Federal de Ouro Preto. IV. Título.

CDU 004

Bibliotecário(a) Responsável: Luciana De Oliveira - SIAPE: 1.937.800

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
DEPARTAMENTO DE COMPUTACAO

# FOLHA DE APROVAÇÃO

**Laura Martins da Costa Coura Marinho**

**EducaAfeto: Reconhecimento de emoções na avaliação de sistemas interativos de aprendizagem**

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 20 de Março de 2023.

Membros da banca

Saul Emanuel Delabrida Silva (Orientador) - Doutor - Universidade Federal de Ouro Preto
Andrea Gomes Campos Bianchi (Coorientadora) - Doutora - Universidade Federal de Ouro Preto
Reinaldo Silva Fortes (Examinador) - Doutor - Universidade Federal de Ouro Preto
Jadson Castro Gertrudes (Examinador) - Doutor - Universidade Federal de Ouro Preto

Saul Emanuel Delabrida Silva, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 20/03/2023.

Documento assinado eletronicamente por **Saul Emanuel Delabrida Silva**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 21/03/2023, às 11:16, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0493032** e o código CRC **B656EBE3**.



R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591692 - www.ufop.br

*Dedico este trabalho ao meu pai Fabrício, cujo suporte e carinho ainda carrego no coração.*

# Agradecimentos

Em primeiro lugar, agradeço à minha família, os principais responsáveis por eu ter chegado até aqui. Sem o amor e apoio deles, assim como o seu exemplo, jamais teria ido tão longe e me tornado a pessoa que sou hoje.

Agradeço também a todos os meus amigos, tantos que não sou capaz de citar aqui, mas principalmente Lauanda, Gabriel Licorice e Matheus, pelos tantos momentos em que me ofereceram ajuda, tornando essa jornada mais fácil.

A meus antigos professores do IFMG, Silvia, Osvaldo e Adolfo, pelos ensinamentos e incentivos que me levaram a iniciar esta etapa da minha vida que agora se conclui, sendo inspirações na minha vida acadêmica e profissional.

Por fim, agradeço aos meus orientadores, professores Dr. Saul e Dra. Andrea, por me guiarem impecavelmente durante este momento tão importante (e difícil) da graduação, à Universidade Federal de Ouro Preto, pelo ambiente e qualidade de ensino, e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

Ninguém ignora tudo. Ninguém sabe tudo. Todos nós sabemos alguma coisa. Todos nós ignoramos alguma coisa. Por isso aprendemos sempre. (FREIRE et al., 2003)

# Resumo

A utilização de sistemas interativos no contexto escolar, tal como jogos educativos e ambientes virtuais de aprendizagem, tem se tornado cada vez mais comum, em especial devido ao seu caráter lúdico, o que leva a um processo de aprendizado mais atrativo e interessante ao aluno. Todavia, o desenvolvimento destes sistemas necessita de métricas adequadas para sua avaliação, de tal modo que seus problemas sejam explicitados de maneira clara e objetiva. Para tanto, destaca-se a observação dos estados emocionais do usuário durante o uso de uma ferramenta de caráter interativo como uma importante metodologia, utilizando da classificação de expressões faciais. Diante do exposto, o presente trabalho visa elaborar um modelo de rede neural convolucional capaz de realizar a classificação de emoções para, em trabalhos futuros, desenvolver uma aplicação capaz de reconhecer as emoções de um usuário por meio de webcam. Assim, desenvolveu-se um modelo que, através de testes, apresentou como melhor acurácia 94,69%, valor que, quando comparado com o melhor valor obtido em trabalhos relacionados (97%), indica a qualidade da arquitetura do modelo e a relevância de seu uso em possíveis trabalhos futuros.

**Palavras-chave**: Rede Neural Convolucional. Reconhecimento de emoções. Sistemas interativos de aprendizagem.

# Abstract

The use of interactive systems in the school context, such as educational games and virtual learning environments, has become increasingly common, especially due to their playful nature, which leads to a more attractive and interesting learning process to the student. However, the development of these systems needs adequate metrics for their evaluation, in such a way that their problems are explicit in a clear and objective way. Therefore, the observation of the user's emotional states during the use of an interactive tool is highlighted as an important methodology, using the classification of facial expressions. In view of the above, the present work aims to develop a convolutional neural network model capable of classifying emotions in order to, in future works, create an application capable of recognizing the emotions of a user through a webcam. Thus, a model was developed which, through tests, presented the best accuracy of 94.69%, a value that, when compared with the best value obtained in related works (97%), indicates the quality of the model's architecture the relevance of its use in possible future work.

**Keywords**: Convolutional Neural Network. Emotion recognition. Interactive learning systems.

# Lista de Ilustrações

# Lista de Tabelas

# Lista de Abreviaturas e Siglas

| | |
|---|---|
| DECOM | Departamento de Computação |
| UFOP | Universidade Federal de Ouro Preto |
| DAN | Deep Alignment Network |
| CK+ | Extended Cohn–Kanade |
| CNN | Convolutional Neural Networks |
| EV | Expressional Vector |
| LSTM | Long Short-Term Memory |
| MTCNN | MultiTask Convolutional Neural Network |
| EEG | Eletroencefalografia |
| RNN | Recurrent Neural Network |
| CLAHE | Contrast-limited Adaptive Histogram Equalization |

# Sumário

# 1 Introdução

A educação é um dos fundamentos da sociedade atual, sendo responsável, no âmbito escolar, pela construção do ser humano através da obtenção de conhecimentos e habilidades. Para tanto, diversos métodos e ferramentas são empregados, muitas vezes definidos pelo contexto em que são utilizados e pela informação que se visa disponibilizar aos alunos. Logo, tendo em vista a conjuntura atual, marcada pelo constante desenvolvimento de novas tecnologias, destaca-se as ferramentas baseadas em sistemas interativos.

Os sistemas interativos de aprendizagem são definidos por métodos pedagógicos que utilizam de páginas da internet e softwares, tais como jogos educativos e ambientes virtuais de aprendizagem, de modo a complementar uma aula ou disciplina. No geral, o uso destas ferramentas provém de uma necessidade e interesse de aplicação em sala de aula de atividades lúdicas, caracterizadas pelo seu objetivo de desenvolver o aprendizado de forma mais atrativa para o aluno (SANT'ANNA; NASCIMENTO, 2011).

Isto significa que estas atividades visam cumprir um papel de auxiliar na construção do conhecimento, permitindo também o desenvolvimento da criatividade, capacidades de comunicação e outras características positivas do estudante. Entretanto, a elaboração de sistemas adequados, de tal modo que eles sejam engajantes e atraiam o interesse de quem está os usando, assim como também sejam capazes de criar um ambiente propício ao aprendizado, é muitas vezes uma tarefa difícil.

Portanto, faz-se necessário a concepção e execução de meios de avaliação destes sistemas interativos. Para tanto, observa-se que a análise do estado emocional dos alunos é uma métrica eficiente para realizar a qualificação de uma determinada aplicação. A título de exemplo, um estudante que apresenta emoções negativas durante o uso revela dificuldades no tema abordado e/ou falhas na construção do sistema.

Deste modo, o reconhecimento de emoções surge como um dos principais métodos avaliativos de páginas da internet, jogos e demais recursos educativos de caráter interativo. São várias as abordagens envolvendo a recognição de sentimentos, como, por exemplo, a utilização de sensores físicos, capazes de extrair informações psicofisiológicas de um usuário (SILVA, 2022), entretanto suas implementações apresentam problemas, sendo, em alguns casos, até mesmo inviáveis. Isto é especialmente visível nos sensores físicos, que contém um custo mais elevado de obtenção e apresentam certo grau de desconforto quando utilizados.

Tendo isto em vista, nota-se que a utilização de métodos menos intrusivos, tal como o reconhecimento feito por *webcams*, é mais recomendada. Através de técnicas de análise facial fundamentadas em redes neurais artificiais, é possível o desenvolvimento de ferramentas que sejam capazes de reconhecer estados emocionais de seres humanos com uma alta taxa de acurácia

e velocidade. Logo, neste contexto, o presente trabalho visa a elaboração de modelo de rede neural convolucional capaz de realizar a classificação de emoções, de modo a aplicá-lo, futuramente, em trabalhos que envolvam o desenvolvimento do sistema completo de reconhecimento de emoções por vídeo.

## 1.1 Justificativa

Ferramentas educativas, em especial aquelas definidas pelas suas características de interatividade com os estudantes, devem ser desenvolvidas atenciosamente, tendo em vista sua utilização em locais de ensino e o impacto positivo que podem gerar quando empregadas adequadamente. Um importante componente deste desenvolvimento é o uso de testes e demais verificações que retornem um feedback claro e objetivo. Entretanto, somente o feedback feito pelos próprios alunos pode ser tendencioso, não revelando de maneira explícita os possíveis problemas da ferramenta avaliada.

Deste modo, faz-se uso de métricas de avaliação que não dependam da resposta do estudante, como a classificação de suas emoções durante o uso da ferramenta. Nota-se, porém, que a análise feita por meio de sensores físicos limitam o comportamento natural do usuário, gerando desconfortos, e são difíceis de adquirir. Por conseguinte, o presente projeto foca no reconhecimento de emoções por meio de expressões faciais observadas em vídeo.

Além de permitir a verificação de estados emocionais dos alunos, este trabalho visa gerar um método de avaliação de sistemas através da geração de relatórios com as emoções observadas, permitindo, assim, que sejam desenvolvidas ferramentas educativas mais engajantes e interessantes. Com tais características mais evidenciadas, as ferramentas usadas em sala de aula permitirão que alunos se envolvam mais com os conteúdos ensinados, aprendendo de maneira lúdica e atrativa.

## 1.2 Objetivos

Os objetivos do presente trabalho estão descritos abaixo, sendo divididos em Objetivo Principal e Objetivos Específicos:

### 1.2.1 Objetivo Principal

Desenvolver um modelo de rede neural convolucional capaz de realizar o reconhecimento de emoções por meio de expressões faciais em imagens, otimizando-o através de uma análise dos melhores hiperparâmetros do modelo e diferentes configurações de bases de dados para treinamento, validação e teste.

### 1.2.2 Objetivos Específicos

Especificamente este trabalho tem como objetivos:

- Desenvolver um modelo de rede neural convolucional (CNN) de modo que ele possa classificar emoções em sete classes: felicidade, tristeza, medo, nojo, raiva, surpresa e desprezo;
- Otimizar os hiperparâmetros do modelo por meio da biblioteca *Keras Tuner*.
- Realizar testes com diferentes configurações de bases de dados.
- Avaliar o desempenho do modelo implementado, comparando a acurácia com os resultados obtidos em trabalhos relacionados;

## 1.3 Organização do Trabalho

Este trabalho é organizado em cinco capítulos distintos, sendo o presente capítulo referente à introdução. Já os demais capítulos são definidos através de componentes fundamentais para a estrutura de uma monografia: no capítulo 2, realiza-se uma revisão bibliográfica acerca do tema abordado, sendo feita a apresentação de trabalhos relacionados e de uma fundamentação teórica.

No capítulo 3, tem-se o desenvolvimento do trabalho, no qual define-se o tipo de pesquisa e como ela foi feita. Portanto, detalha-se os modelos e base de dados que compõem a metodologia, exibindo como eles são utilizados para alcançar os objetivos propostos de reconhecer emoções e otimizar o modelo desenvolvido para realizar este reconhecimento.

Ademais, o capítulo 4 também apresenta uma seção com os resultados obtidos a partir da implementação do modelo elaborado e dos testes realizados. Por fim, o capítulo 5 possui as considerações finais do trabalho, explicitando os objetivos que foram alcançados e as possibilidades de propostas para trabalhos futuros.

# 2 Revisão Bibliográfica

Neste capítulo, apresenta-se uma contextualização sobre o tema de reconhecimento de emoções e visão computacional, iniciando-a através da exposição de trabalhos relacionados que abordaram estes assuntos de maneira geral até aqueles que os abordaram em contextos específicos, tais como no uso de uma investigação acerca a imitação de emoções por pessoas que tem autismo. Em seguida, tem-se uma fundamentação teórica sobre os conceitos que serão discutidos ao decorrer deste trabalho, visando possibilitar ao leitor um melhor entendimento da metodologia aplicada.

## 2.1 Trabalhos Relacionados

Primeiramente, analisa-se os trabalhos de visão computacional voltados ao reconhecimento de emoções que são aplicados a contextos gerais de uso. Em sua maioria, os trabalhos com esta temática se baseiam exclusivamente no reconhecimento por meio de expressões faciais, havendo variação apenas nos principais algoritmos utilizados em sua concepção, estes sendo normalmente fundamentados em redes neurais artificiais (*artificial neural networks*) e aprendizagem em profundidade (*deep learning*).

Tautkute, Trzcinski e Bielski (2018), por exemplo, apresentam um sistema desenvolvido com base na arquitetura *Deep Alignment Network* (DAN), definida como um método de detecção de pontos de referências (*landmarks*) do rosto que utiliza de redes neurais profundas. Realizando o treinamento e teste do modelo em diferentes bases de dados (*datasets*) e comparando a acurácia do modelo desenvolvido com outros, os autores puderam observar que seu modelo apresentou uma acurácia melhor em duas das três bases de dados de teste. A título de exemplo, para a base de dados CK+ (*Extended Cohn–Kanade*), foi constatado uma acurácia de 0,736 para o modelo dos autores, enquanto os outros modelos apresentaram os valores 0,628, 0,728, 0,304 e 0,.204.

Outro trabalho que também utiliza de redes neurais profundas para o reconhecimento de rostos é o de Jain, Shamsolmoali e Sehdev (2019). Visando classificar emoções em seis classes diferentes, os autores fizeram testes em duas bases de dados diferentes e definiram seu modelo através de *deep residual blocks* e camadas convolucionais. Já o trabalho de Mehendale (2020) utiliza de redes neurais convolucionais (*Convolutional Neural Networks* - CNN) de dois níveis, sendo o primeiro voltado à eliminação do fundo (*background*) e geração de um vetor de expressão facial (EV) e o segundo voltado à classificação das emoções.

Ribeiro (2018) apresentou o desenvolvimento de um método de meta-aprendizado em aprendizagem em profundidade para a seleção automática dos melhores hiperparâmetros das Redes Neurais Convolucionais, ou seja, através da utilização da biblioteca de otimização *hyperopt*,

o autor otimiza os hiperparâmetros das CNNs responsáveis pelo treinamento e classificação de expressões faciais. Ko (2018) expõe um modelo híbrido de reconhecimento, combinando CNNs para características fixas (*spatial features*) de *frames* individuais de uma determinada imagem e *Long Short-Term Memory* (LSTM) para características que mudam ao longo do tempo (*temporal features*) de *frames* consecutivos.

Por outro lado, trabalhando não apenas com o rosto, mas também com a voz e linguagem corporal, tem-se o artigo de Ranganathan, Chakraborty e Panchanathan (2016). Os autores apresentam um modelo baseado em *Convolutional Deep Belief Network* (CDBN) para elaborar uma abordagem multimodal de reconhecimento de emoções. Também utilizando de CNNs, Ghofrani, Toroghi e Ghanbari (2019) exibem um modelo que utiliza *MultiTask Convolutional Neural Network* (MTCNN) para a identificação de rostos e corte dos limites corretos deles na primeira etapa e arquitetura ShuffleNet V2 para a identificação de emoções na segunda etapa.

Visando lidar com o problema de reconhecimento de emoções em vídeo, a pesquisa de Zheng (2018) propõe um modelo fundamentado em LSTM, este que, por sua vez, é baseado em mecanismo de atenção (*attention mechanism*), sendo feito a extração das características do rosto por meio de MTCNN. Já Liu, Su e Liu (2017) apresentou o desenvolvimento de um modelo baseado em LSTM para lidar com reconhecimento de emoções multimodal, trabalhando com expressões extraídas de vídeo e emoções neurais provindas de sinais de eletroencefalograma (EEG). Ademais, destaca-se que o modelo elaborado pelos autores possui várias camadas e utiliza de dois mecanismos de atenção.

Em relação aos trabalhos que utilizam de reconhecimento de emoções em contextos específicos, tem-se, primeiramente, o artigo de Silva (2022). Com um foco em detectar estados emocionais de tédio e estresse em pessoas jogando, o autor utiliza de rede neural recorrente (*Recurrent Neural Network* - RNN) com LSTM. Sob outra perspectiva, o trabalho de Drimalla et al. (2021) utiliza dos conceitos de Visão Computacional e de identificação de sentimentos para investigar a capacidade de imitação de expressões faciais de indivíduos com autismo. Para tanto, emprega-se da ferramenta *OpenFace*, baseada em redes neurais profundas, de modo a obter os movimentos e ações dos rostos dos participantes envolvidos nos testes.

Portanto, constata-se que o trabalho exposto neste documento assemelha-se aos trabalhos apresentados acima em seu uso de redes neurais para o desenvolvimento de um modelo de extração de características faciais e reconhecimento de emoções. Entretanto, difere-se ao apresentar uma metodologia que foca na otimização do modelo por meio de alterações de seus hiperparâmetros e alterações nas configurações de bases de dados utilizadas para treino, validação e teste. Além disso, distingue-se também por fornecer um contexto propício para a elaboração de futuros trabalhos envolvendo a aplicação do modelo em um contexto educacional, em especial para produtos de caráter interativo, tal como jogos educacionais e ambientes virtuais de aprendizagem.

Por fim, visando auxiliar na comparação entre este trabalho e os discutidos anteriormente nesta seção, tem-se a Tabela 2.1.

| **TRABALHO** | **MODELO** | **BASE DE DADOS** |
|---|---|---|
| TAUTKUTE; TRZCINSKI; BIELSKI, 2018 | CNN (DAN) | CK+, ISED, JAFFE, AffectNet |
| JAIN; SHAMSOLMOALI; SEHDEV, 2019 | DNN | CK+, JAFFE |
| MEHENDALE, 2020 | CNN (FERC) | CK+, Caltech faces, CMU |
| RIBEIRO, 2018 | CNN | JAFFE, CK+, FERPlus |
| KO, 2018 | CNN, LSTM | - |
| RANGANATHAN; CHAKRABORTY; PANCHANATHAN, 2016 | CDBN | emoFBVP, CK |
| GHOFRANI; TOROGHI; GHANBARI, 2019 | MTCNN, CNN (ShuffleNet V2) | FER2013 |
| ZHENG, 2018 | CNN, LSTM | CHEAVD |
| LIU; SU; LIU, 2017 | LSTM, RNN | Mahnob-HCI |
| SILVA, 2022 | RNN, LSTM | - |
| DRIMALLA et al., 2021 | DNN (OpenFace) | - |
| **ESTE TRABALHO** | **CNN (SimpleNet otimizado)** | **Extended and Augmented Google FER, CK+, Natural Human Face Images for Emotion Recognition** |

Tabela 2.1 – Comparação entre os trabalhos relacionados e o trabalho proposto. Fonte: elaborado pela autora.

## 2.2 Fundamentação Teórica

Nesta seção, tem-se como objetivo permitir ao leitor um entendimento sobre os conceitos e termos abordados neste trabalho, em especial no capítulo de metodologia. Deste modo, explica-se as noções principais de redes neurais e aprendizagem em profundidade, necessárias para uma compreensão sobre CNN, e de reconhecimento de emoções.

### 2.2.1 Reconhecimento de emoções

A determinação de estados emocionais de um indivíduo, quando realizada por meio de imagens e vídeos, necessita da análise de expressões faciais. Tal análise é, comumente, fundamentada na observação das principais características que compõem um rosto humano, sendo estas sobrancelhas, olhos, boca, testa, queixo, nariz e bochechas. Entretanto, tendo em vista a complexidade emocional possuída por seres humanos e as sutilezas em movimentos faciais, a classificação de emoções pode ser uma tarefa de extrema dificuldade.

Tendo isto em mente Ekman e Friesen (2003) dividem as emoções em seis grandes "famílias": felicidade, tristeza, medo, nojo, raiva e surpresa (representadas graficamente na Figura 2.1, em ordem). Assim, os autores afirmam que cada emoção pode ser também classificada internamente, tal como observa-se para a surpresa, que pode ser acompanhada por sentimentos

positivos (*dazed surprise*) ou negativos (*dumbfounded surprise*).

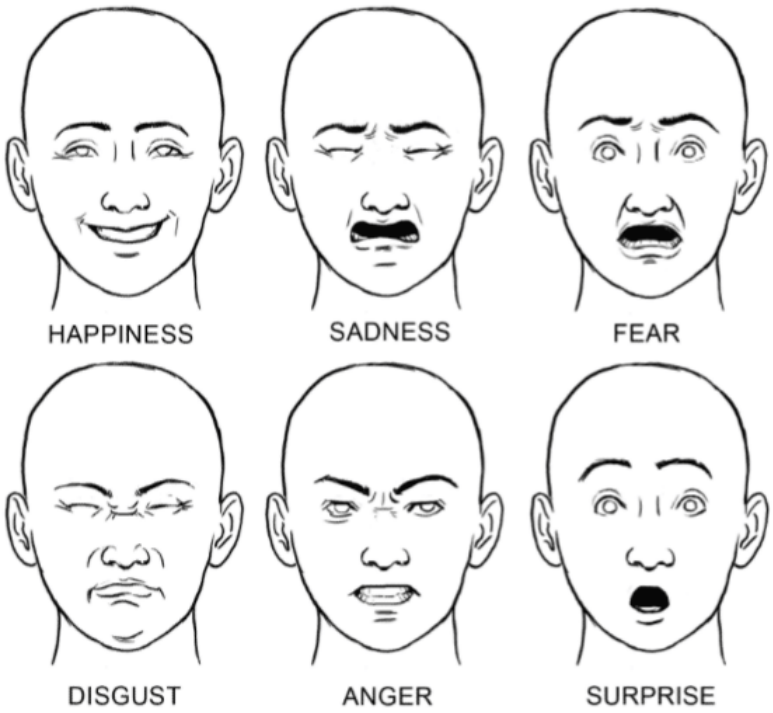


Figura 2.1 – Representação visual das 6 emoções básicas definidas por Ekman e Friesen. Fonte: Ekman's 6 Basic Emotions and How They Affect Our Behavior. Disponível em: <https://themindsjournal.com/basic-emotions-and-how-they-affect-us/> . Acesso em 14 ago. 2022.

Todavia, outros autores identificam diferentes classificações de emoções. Matsumoto (1992), por exemplo, parte do descobrimento feito por Ekman e Friesen em 1986 de que desprezo (*contempt*) seria outra emoção universal, e apresenta mais evidências para tal afirmação através de uma pesquisa com pessoas de diferentes culturas. Logo, considerando as bases de dados utilizados neste trabalho, foca-se no reconhecimento das setes principais emoções definidas por Ekman e Friesen: felicidade (*happiness*), tristeza (*sadness*), medo (*fear*), nojo (*disgust*), raiva (*anger*), surpresa (*surprise*) e desprezo (*contempt*).

### 2.2.2 Aprendizagem em profundidade

No contexto tecnológico atual, observa-se diversas áreas que podem ser contempladas pela Inteligência Artificial, campo da Ciência de Computação que estuda a utilização de agentes que realizam ações baseadas no comportamento inteligente de humanos. Tais áreas variam desde trabalhos específicos normalmente desempenhados apenas por humanos até a resolução de problemas matemáticos considerados difíceis, tal como, por exemplo, problemas de solução de equações diferenciais parciais (EDP), utilizadas no estudo de eventos físicos.

Uma das áreas de maior destaque, porém, é a de aprendizagem em profundidade (*deep*

*learning*). Sendo um subconjunto da área de aprendizado de máquina (*machine learning*), é definida como *deep neural network architectures with improved learning capabilities* (JANIESCH; ZSCHECH; HEINRICH, 2021). Ou seja, trata-se de uma área que permite que um software treine a si mesmo através de redes neurais de várias camadas, estas com grandes quantidades de dados pertinentes à tarefa que ele visa executar.

Tal treinamento é feito através de uma hierarquia de conceitos em que os conceitos considerados complicados são construídos a partir dos mais simples (GOODFELLOW; BENGIO; COURVILLE, 2016). Para tanto, emprega-se o uso de camadas (*layers*) não lineares para a extração de características e transformação destas, sendo as camadas mais baixas mais próximas à entrada de dados responsáveis por aprender as características simples enquanto as camadas mais altas aprendem as características complexas derivadas das características das camadas abaixo (SHINDE; SHAH, 2018).

Portanto, tem-se diferentes tipos de camadas, normalmente classificadas em visíveis (*visible*) e escondidas/invisíveis (*hidden*), sendo as visíveis divididas entre camadas de entrada (*input*) e de saída (*output*). As de entrada lidam com os dados que vão ser processados enquanto as de saída lidam com a predição ou classificação final feita pelo modelo. Além disso, os dados das camadas visíveis são aqueles que somos capazes de observar, enquanto os dados das camadas invisíveis são abstratos, sendo valores apenas usados pelo modelo para definir conceitos importantes dos dados de entradas. Na Figura 2.2, é possível observar melhor as camadas e seu funcionamento.

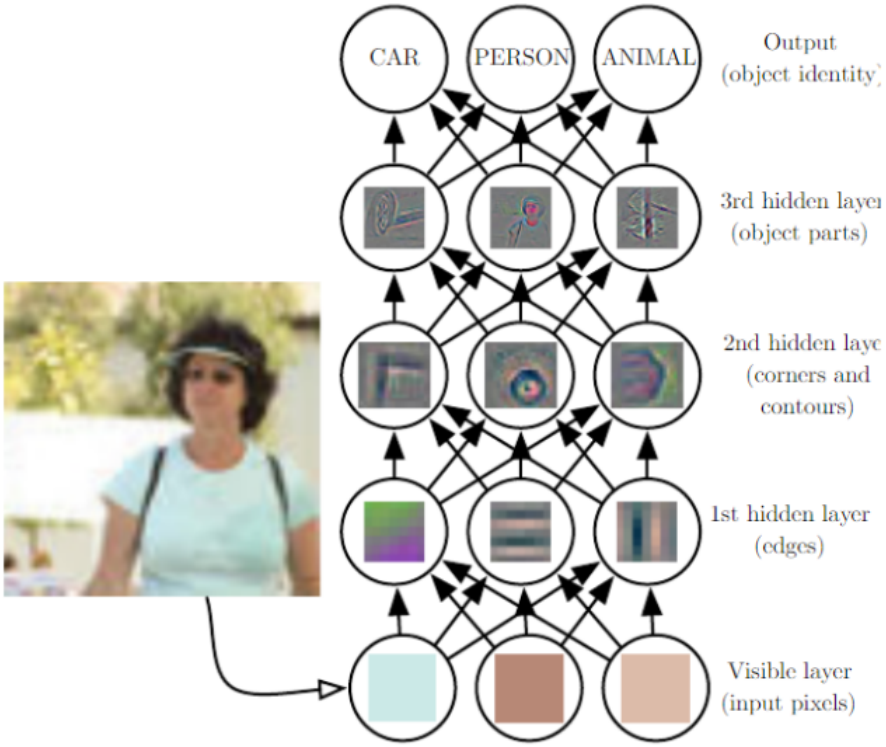


Figura 2.2 – Diagrama demonstrando como seria o funcionamento de uma rede neural e cada camada dela para uma determinada imagem. Fonte: Deep learning, 2016.

### 2.2.3 Redes neurais artificiais

Tendo em vista, portanto, as noções definidas em aprendizagem de profundidade, há o desenvolvimento de diversas arquiteturas e modelos que utilizam do conceito de camadas, estas denominadas redes neurais artificiais. De acordo com Haykin (2001), uma rede neural é uma máquina que é projetada para modelar a maneira como o cérebro realiza uma tarefa particular ou uma função de interesse. Logo, as redes neurais caracterizam-se como modelos compostos por camadas, estas sendo compostas por neurônios (*neurons*), que se inspiram no funcionamento e estrutura dos cérebros humanos.

Na Figura 2.3, tem-se uma representação de uma rede neural genérica e simples. É possível observar que os círculos são os neurônios da rede e que, quanto mais camadas existem, mais profunda é a rede. Também é notável que há outros componentes que definem o funcionamento da rede: conexões (*connections*), pesos (*weights*), polarização externa ou viés (*biases*), função de propagação (*propagation function*) e uma regra de aprendizado (*learning rule*).

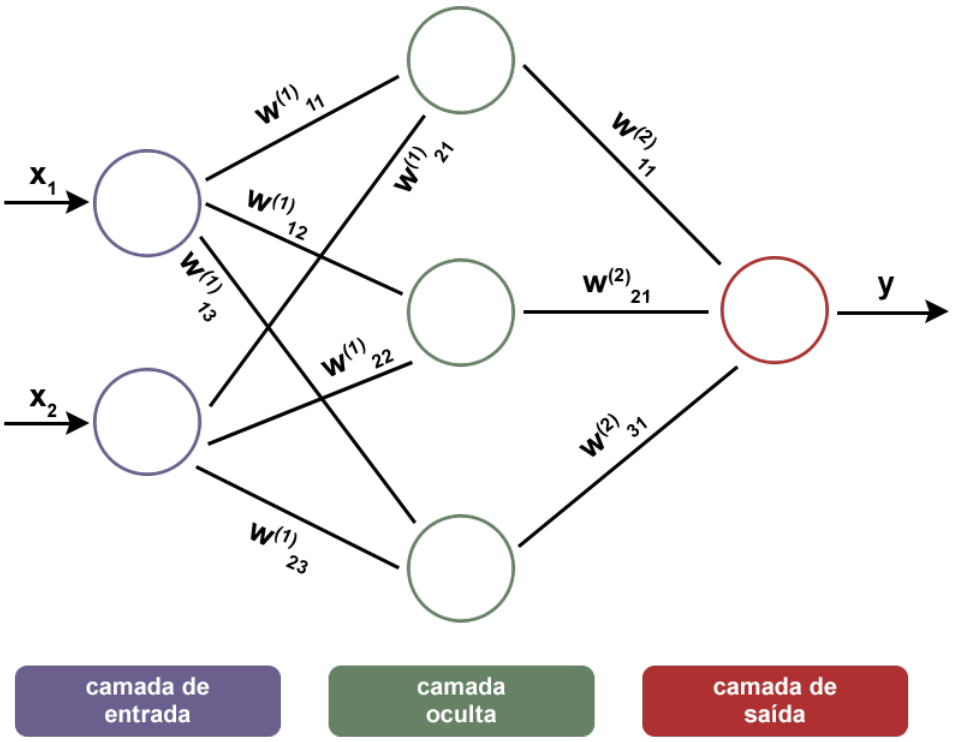


Figura 2.3 – Diagrama representando uma rede neural genérica com seus componentes. Fonte: elaborado pela autora.

Já os neurônios possuem uma entrada proveniente dos neurônios anteriores (normalmente referida como sinapse) que é multiplicada pelo peso e, em seguida, tem as entradas e pesos que foram multiplicados somados no combinador linear. Após, tem-se a adição do viés a soma, resultando na função de ativação o que, por fim, leva a saída do neurônio. Tais etapas são, assim, demonstradas na Figura 2.4.

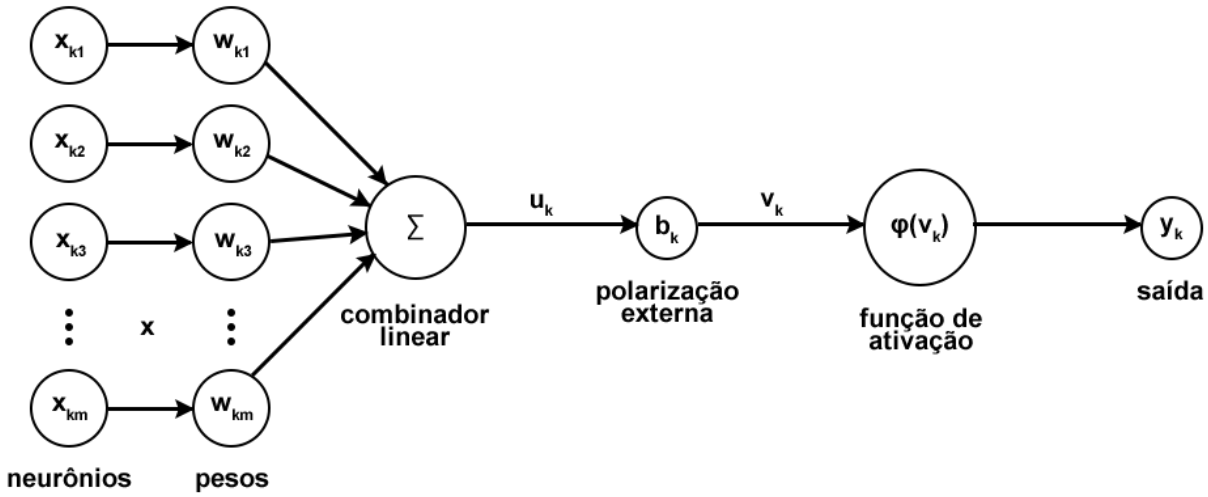


Figura 2.4 – Diagrama representando um neurônio com todas as suas etapas. Fonte: elaborado pela autora.

Ademais, as redes neurais podem ser categorizadas em diferentes tipos, cada uma adequada a diferentes problemas e fontes de dados. Por exemplo, tal como afirmam Hao, Zhang e Ma (2016), as redes neurais convolucionais são as arquiteturas mais populares para reconhecimento de imagens, e as redes neurais recorrentes são mais aplicadas para tarefas sequenciais como reconhecimento de falas ou escrita à mão. Portanto, considerando a temática desta pesquisa, foca-se, na subseção a seguir, nos conceitos principais das CNNs.

#### 2.2.3.1 Redes neurais convolucionais (CNN)

Redes neurais convolucionais são um tipo especializado de rede neural projetadas para processar dados que estão na forma de múltiplos *arrays* (RIBEIRO, 2018). Ou seja, são um tipo de rede neural artificial em que a arquitetura é capaz de lidar com dados cujo formato seja de múltiplos vetores. Além disto, tais redes são também definidas pelo uso da operação matemática de convolução, esta que substitui a multiplicação de matrizes em pelo menos uma das camadas de uma determinada rede.

A convolução é uma operação que ocorre entre duas matrizes (ou entre duas funções) e que realiza a aplicação de uma matriz sobre outra, resultando em uma nova matriz. Em processamento de imagens, e por conseguinte em redes neurais utilizadas para lidar com imagens, a convolução é utilizada com o intuito de aplicar um filtro (*kernel*) em cada pixel da imagem, exceto os da borda, de maneira que o pixel central da imagem resultante seja substituído pelo resultado da operação. Entretanto, em redes neurais convolucionais, a saída obtida não é uma imagem, e sim um mapa de atributos reconhecidos pelo filtro.

Portanto, são estes atributos obtidos pela convolução (como, por exemplo, bordas) que são utilizados para o reconhecimento de características e objetos em imagens pelo modelo. Já

em seguida, após a realização desta operação, tem-se a etapa conhecida como *pooling*. Definida por sua função de *merge semantically similar features into one* (LECUN; BENGIO; HINTON, 2015), ela é responsável por reduzir o tamanho dos mapas gerados anteriormente, extraindo as características mais dominantes dos mesmos.

Esta extração pode ser feita através da obtenção do maior valor (*Max Pooling*) ou da média de todos os valores (*Average Pooling*). Entretanto, considerando que o *Max Pooling* suprime os ruídos de maneira mais eficiente que o *Average Pooling*, ele é mais recomendado para a construção de uma CNN.

Por fim, tem-se a etapa relacionada à classificação da imagem. Como demonstra a Figura 2.5, após a etapa de aprendizado das *features*, há um conjunto de camadas voltadas a "achatar" (*flatten*) a imagem em um vetor de colunas, este que é então usado como a entrada de uma rede neural. Já a classificação é realizada na última parte da rede, utilizando da função *Softmax*.

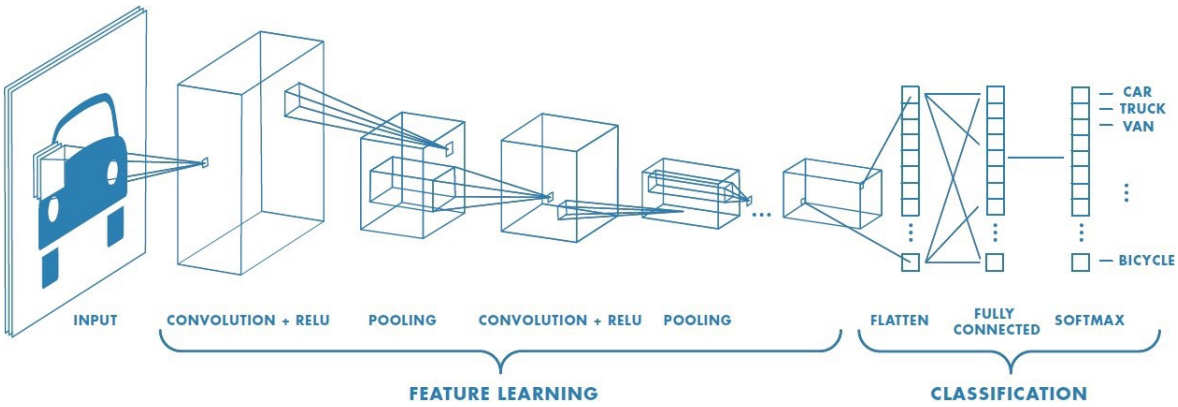


Figura 2.5 – Diagrama com todas as etapas de uma rede neural convolucional simples. Fonte: A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way, 2018. Disponível em: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. Acesso em 15 ago. 2022.

# 3 Desenvolvimento

Neste capítulo, apresenta-se os métodos e técnicas utilizados para a elaboração do trabalho. Inicialmente, tem-se uma etapa voltada à análise das bases de dados escolhidos para o treino e teste do modelo a ser elaborado, descrita na Seção 3.1. Através do uso do site *Kaggle*, uma comunidade online para pessoas que estudam ciência de dados e aprendizagem de máquina, foram obtidos dois *datasets* diferentes de expressões faciais. Já o terceiro *dataset* foi obtido através de uma requisição para os autores.

Em seguida, tem-se a etapa de pré-processamento (Seção 3.2) Nela, o foco foi em ajustar as imagens das bases de dados de tal modo que todas sigam o mesmo padrão, possuindo, portanto, a mesma dimensão (em *pixels*) e também os mesmos rótulos (*labels*). Com os dados prontos, foi feita a definição das arquiteturas dos modelos das redes neurais e a realização de testes dos modelos com os dados, tal como detalhada na Seção 3.3.

Em relação aos testes (Seção 3.4), foram definidas diferentes estratégias visando constatar qual base de dados apresentava melhores resultados quando utilizado na construção do modelo. Assim, os testes foram feitos com cada *dataset* individual, misturando apenas dois *datasets* e, por fim, misturando os três *datasets*. Para cada configuração, observou-se as taxas de acurácia de teste (*accuracy*) e perda (*loss*).

Com os resultados obtidos nos testes, definiu-se a melhor configuração das bases de dados no sistema. Por fim, foi feita uma comparação com trabalhos relacionados que também desenvolveram modelos de classificação de emoções e utilizaram a base de dados CK+, analisando, então, quais mudanças podem ser aplicadas ao modelo apresentado neste trabalho de modo a melhorá-lo (Seção 4.2). Abaixo, tem-se seções que apresentam em detalhes as etapas descritas anteriormente.

## 3.1 Caracterização das bases de dados

A primeira etapa que compõe a metodologia deste trabalho é a aquisição e caracterização das bases de dados. As três bases escolhidas foram a *Extended FER*, *Natural Human Face Images* e a *Extended Cohn-Kanade* (CK+). As duas primeiras foram disponibilizadas, respectivamente, por Prajwal Sood e Sudarshan Vaidya, enquanto a terceira foi obtida por meio de um requerimento aos autores, (LUCEY et al., 2010) (KANADE; COHN; TIAN, 2000). Nas subseções a seguir, descreve-se as bases.

### 3.1.1 Extended and Augmented Google FER

Este *dataset* é uma versão extendida do dataset Google FER 2013, possuindo uma maior quantidade de dados e uma classe nova, *Contempt*. As imagens possuem apenas um canal de cor, sendo, portanto, em escala de cinza (*grayscale*), e tem dimensões 48 x 48 *pixels*. As emoções contidas são felicidade, tristeza, medo, nojo, raiva, surpresa, desprezo e neutro, e as quantidades de imagens para cada são, respectivamente, 7191, 4861, 4096, 2280, 3962, 3201, 2096 e 4958.

Ademais, foi possível observar que além de imagens de pessoas reais, há também desenhos representando expressões faciais e algumas figuras que não possuem rostos ou não correspondem à emoção da classe. Entretanto, tendo em vista a grande quantidade de dados, constata-se que os dados errados, por representarem uma pequena fração dos dados totais, não influenciam negativamente no *dataset*. Abaixo, na Figura 3.1, tem-se exemplos de diferentes expressões faciais armazenadas na base de dados.

Figura 3.1 – Exemplos de imagens do *dataset Extended Google FER*. Fonte: Prajwal Sood. Imagens selecionadas pela autora.

### 3.1.2 Extended Cohn-Kanade (CK+)

A segunda base de dados escolhida é o dataset CK+ (Extended Cohn-Kanade) (LUCEY et al., 2010) (KANADE; COHN; TIAN, 2000). Ela é composta por apenas sete classes, sendo elas felicidade, tristeza, medo, nojo, raiva, surpresa e desprezo. As imagens possuem dimensão 640 x 490 ou 640 x 480 *pixels*, algumas em escala de cinza e outras coloridas. Na Figura 3.2, observa-se alguns exemplos das imagens do dataset.

Figura 3.2 – Exemplos de imagens do *dataset* CK+. Fonte: (LUCEY et al., 2010) (KANADE; COHN; TIAN, 2000). Imagens selecionadas pela autora.

### 3.1.3 Natural Human Face Images for Emotion Recognition

O terceiro e último *dataset* exibe imagens de expressões faciais de pessoas reais e desenhos divididas em 8 classes: felicidade, tristeza, medo, nojo, raiva, surpresa, desprezo e neutro. Respectivamente, tem-se 1406, 746, 570, 439, 890, 775, 208 e 524 imagens para cada classe,

sendo elas em escala de cinza e dimensões 224 x 244 *pixels*. Abaixo, a Figura 3.3 apresenta exemplos destas figuras.

Figura 3.3 – Exemplos de imagens do *dataset Natural Human Face Images*. Fonte: Sudarshan Vaidya. Imagens selecionadas pela autora.

## 3.2 Pré-processamento

A segunda etapa do desenvolvimento é definida pelo conceito de pré-processamento. Como afirmado na tese Batista et al. (2003), esta fase tem como objetivo preparar os dados para que a fase seguinte, a fase de extração de conhecimento, seja mais efetiva. Ou seja, esta fase é definida pelas transformações nos dados que são fundamentais para o funcionamento adequado do modelo de reconhecimento de emoções.

Deste modo, considerando as características das bases de dados apresentadas na seção anterior, nota-se que é necessário fazer algumas alterações nos dados. A primeira alteração feita foi a renomeação manual dos *labels* das imagens, ou seja, do nome das emoções que compõem as bases. Todos os nomes passaram a ser em letra minúscula, e as classes *happiness* e *neutrality* foram modificadas para *happy* e *neutral*, respectivamente. Em seguida, foi definida a função responsável pelas demais alterações essenciais.

Originalmente, esta função era utilizada para realizar todo a etapa de pré-processamento, sendo nomeada como *Data()* e possuindo de entrada o caminho (*path*) do diretório da base de dados desejada e uma variável booleana responsável por indicar se as imagens da base devem ser redimensionadas para 48 x 48 *pixels*. Já dentro da função, as imagens eram transformadas em *numpy arrays* de *datatype float32* e normalizadas, passando a ter seus valores na faixa de 0 e 1. Por último, era gerado um dicionário com as classes (emoções), convertendo-se as classes das imagens da base para o seu respectivo valor número definido no dicionário e retornando este dicionário, as classes convertidas e um *array* com os dados transformados.

Entretanto, devido aos resultados ruins obtidos durante os testes dos modelos, estes apresentados em mais detalhes no Capítulo 4, esta função foi substituída pela utilização da classe *ImageDataGenerator* e da função *image_dataset_from_directory()* da biblioteca *Keras*. Primeiramente, os dados são separados em diretórios de treino, teste e validação, com uma proporção de, respectivamente, 80%, 10% e 10%, por meio da biblioteca *splitfolders*.

Em seguida, gera-se um *Image Data Generator* para o diretório de treino, sendo definidos os parâmetros da classe: rotação (*rotation*), deslocamento horizontal (*width shift range*),

deslocamento vertical (*eight shift range*), níveis de brilho (*brightness range*), virar horizontalmente a imagem (*horizontal flip*), distorção em um eixo (*shear range*), e normalização (*rescale*). Respectivamente, cada parâmetro foi estabelecido com os valores 20, 0.2, 0.2, (0.5, 1.5), *True*, 10 e 1./255.

Tais atributos foram selecionados tendo em vista que o modelo de classificação deve ser capaz de reconhecer emoções em diferentes rostos, estes que podem estar em locais com pouca ou muita luminosidade ou em posições diferentes. Deste modo, os parâmetros focam em transformações geométricas, visando simular rostos que não estejam posicionados totalmente de frente para a câmera, e transformações de brilho. As imagens formadas pelo gerador são salvas no diretório de treino com o prefixo 'aug_' no nome de seu arquivo. Já os outros diretórios não passam por esta etapa de aumento de número de imagens, tendo em vista que não há necessidade.

Em relação ao uso da função *image_dataset_from_directory()*, ela é responsável por, através de acesso ao diretório com as imagens, gerar um objeto *Dataset* da biblioteca *Tensorflow*. Para cada diretório (treino, validação e teste), foi gerado um *Dataset* diferente com os parâmetros de tamanho da imagem (48 x 48), tamanho de *batch* (32), rótulos (obtidos dos nomes dos subdiretórios) e tipo dos rótulos (*categorical*).

As três últimas funções desenvolvidas são responsáveis pelo pré-processamento apenas da base de dados CK+. Considerando que a base original possui diretórios separados para as imagens e seus *labels*, estes informados dentro de arquivos texto, foi necessário a criação de uma função responsável por percorrer os diretórios, salvando os caminhos (*path*) até as imagens e seus respectivos *labels*. Já em seguida, foi criada uma função responsável pela nova estrutura da base de dados, com cada classe tendo seu próprio diretório, e, finalmente, foi criada uma função dedicada à escolha de imagens consideradas apenas relevantes, tendo em vista que as primeiras imagens da base apresentam expressões faciais neutras. Destaca-se que a implementação destas funções foram baseadas nos algoritmos de Sinha (2019), tal como apresentados em seu repositório na plataforma *GitHub*.

## 3.3 Definição do modelo

Tendo os dados corretamente ajustados, a terceira etapa do desenvolvimento é a definição do modelo de classificação de emoções. Para tanto, três modelos diferentes foram desenvolvidos visando identificar aquele com melhor desempenho e, a partir desta determinação, refinar o mesmo através da escolha dos seus melhores hiperparâmetros.

### 3.3.1 Arquiteturas bases para definição do melhor modelo

As duas primeiras arquiteturas utilizadas como base para os modelos são provenientes do artigo (RIBEIRO, 2018), enquanto o terceiro é do artigo (HASANPOUR et al., 2016). O primeiro modelo é baseado na arquitetura de redes neurais convolucionais VGG do artigo (SIMONYAN;

ZISSERMAN, 2014). No artigo que lhe define, a arquitetura regular desta CNN é apresentada tendo os componentes da Figura 3.4, enquanto no código foram introduzidos funções de *Batch Normalization*, de modo a aumentar a velocidade de funcionamento e acurácia da rede através da normalização.

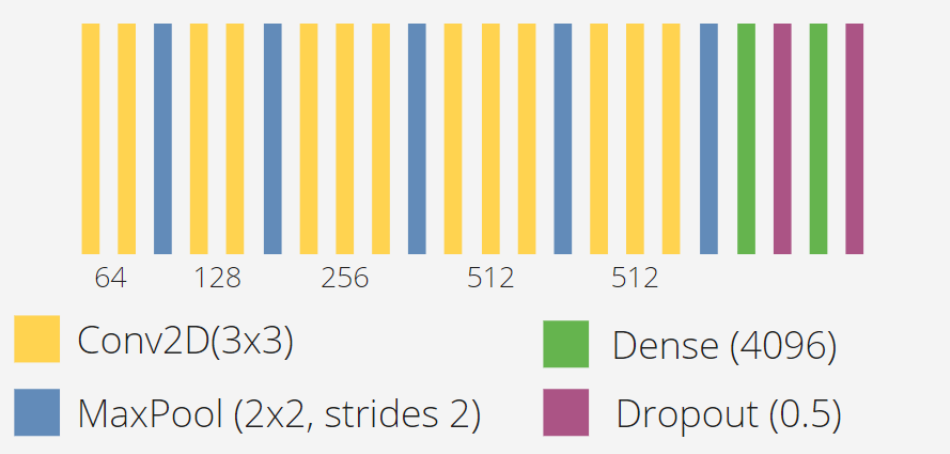


Figura 3.4 – Arquitetura da rede neural VGG-16. Fonte: elaborado pela autora.

O segundo modelo desenvolvido foi baseado na arquitetura *SimpleNet*, explicado no artigo (HASANPOUR et al., 2016). Seguindo o que foi demonstrado no artigo de Ribeiro (2018), a rede neural é composta pelas camadas da Figura 3.5.

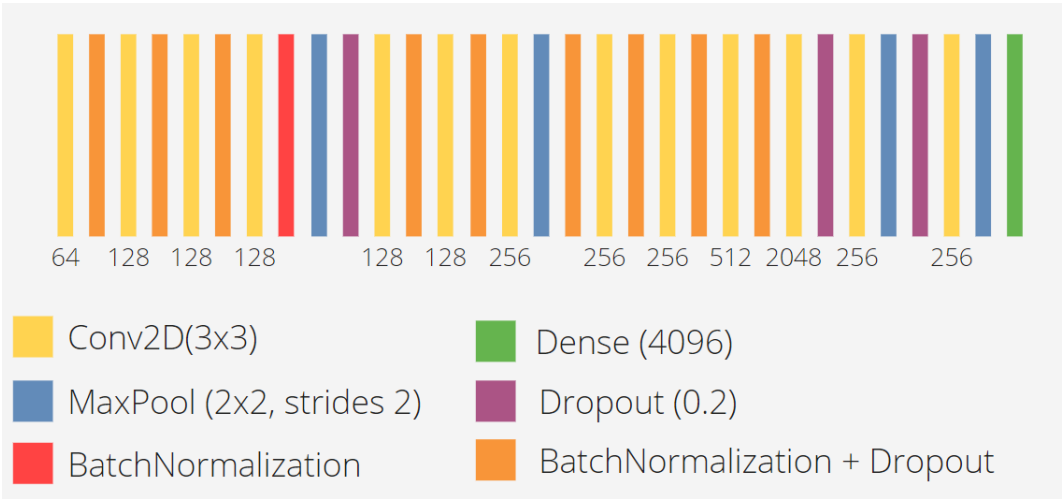


Figura 3.5 – Arquitetura da rede neural *SimpleNet v1*. Fonte: elaborado pela autora.

Por fim, o último modelo desenvolvido foi baseado na versão original da arquitetura *SimpleNet*. Portanto, ela possui um total de 13 camadas, tal como apresentado na Figura 3.6.

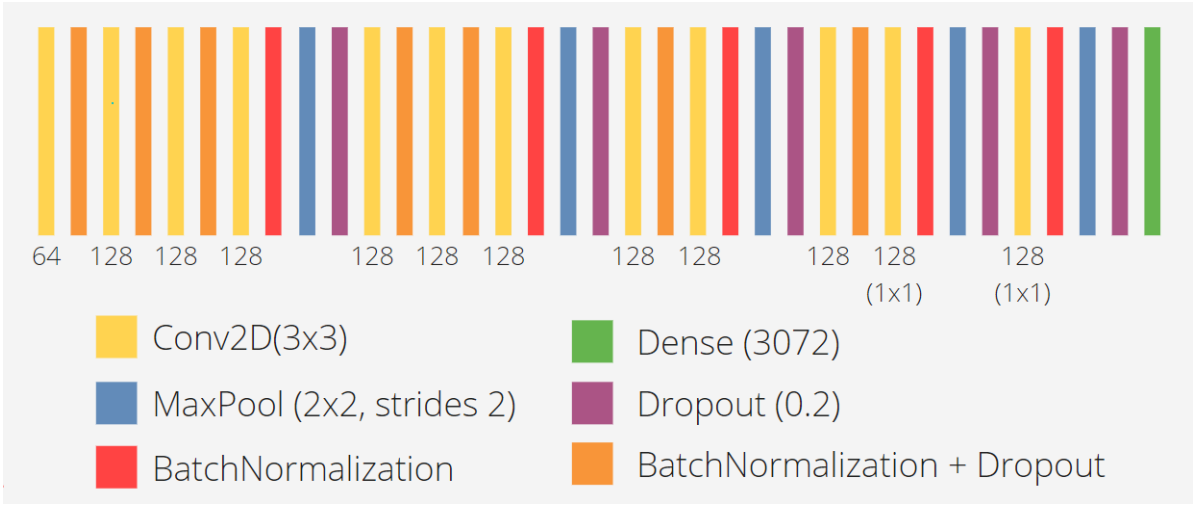


Figura 3.6 – Arquitetura da rede neural *SimpleNet v2*. Fonte: elaborado pela autora.

### 3.3.2 Definindo os melhores hiperparâmetros do modelo

Nesta etapa do desenvolvimento, define-se os hiperparâmetros com os melhores valores para a rede neural desenvolvida na etapa anterior. Assim, tendo em vista que o artigo de Bergstra e Bengio (2012) demonstra que o método de *Random Search* encontra tais valores de maneira mais eficiente que o método de *Grid Search*, exigindo menos tempo computacional, ele foi definido como a técnica a ser utilizada.

Visando visualizar qual arquitetura apresentava os melhores resultados e deveria ter seus hiperparâmetros otimizados, foram feitos testes com uma versão reduzida da base de dados CK+. Esta base foi disponibilizada por Ashadullan Shawon no site *Kaggle*, na qual mantém-se as setes classes (felicidade, tristeza, medo, nojo, raiva, surpresa e desprezo), tendo elas, respectivamente, 207, 84, 75, 177, 135, 249 e 54 imagens. A título de exemplo, a Figura 3.7 apresenta algumas destas imagens.

Figura 3.7 – Exemplos de imagens do *dataset* CK+ reduzido. Fonte: Ashadullan Shawon. Imagens selecionadas pela autora.

Tais testes demonstraram que as duas primeiras arquiteturas (VGG-16 e *SimpleNet v1*) exibiram uma acurácia de aproximadamente 25%, enquanto a terceira (*SimpleNet v2*) exibiu uma acurácia de aproximadamente 80%. Portanto, o modelo definido como o melhor foi o terceiro. Através do uso da biblioteca *Keras Tuner*, sua arquitetura definida na seção anterior é re-implementada de tal modo que os hiperparâmetros selecionados possam ser alterados em cada

etapa da busca. Logo, o seguinte espaço de busca foi estabelecido com auxílio da documentação da biblioteca *Keras*, permitindo portanto a escolha de valores apropriados:

- **units:** valor mínimo de 64, valor máximo de 512, de 64 em 64;
- **kernel size:** (1, 1), (2, 2), (3, 3);
- **pool size:** (1, 1), (2, 2), (3, 3);
- **strides:** (1, 1), (2, 2), (3, 3);

Demais hiperparâmetros, tal como tipo de ativação, porcentagem de *dropout*, dentre outros, foram mantidos com valores fixos, considerando que eles já haviam definidos na arquitetura *SimpleNet* original. Ao fim da busca, o resultado obtido (evidenciado pela função *get best hyperparameter*) permitiu a definição da arquitetura otimizada apresentada na Figura 3.8.

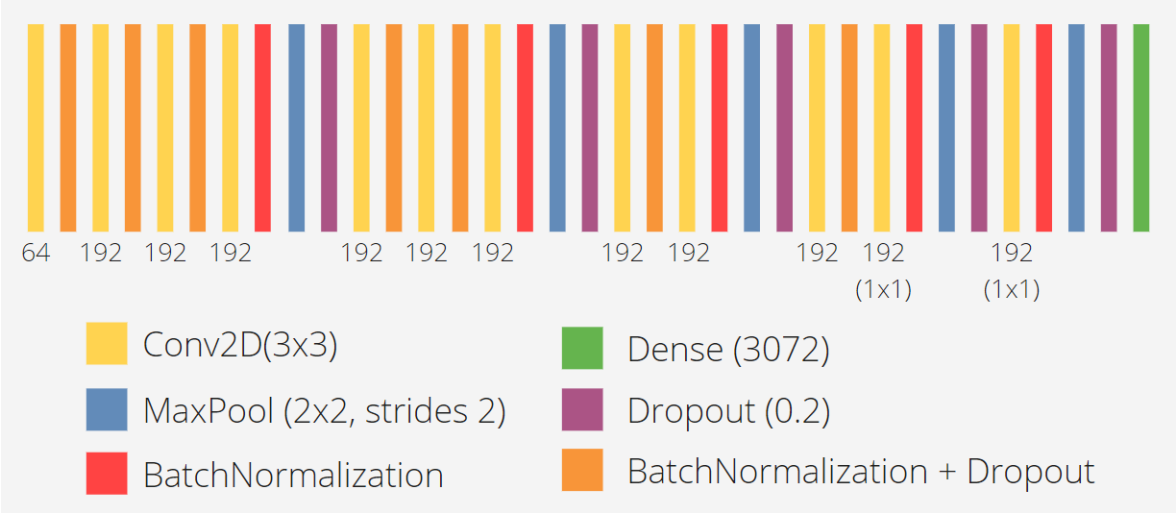


Figura 3.8 – Arquitetura da rede neural *SimpleNet v2* com mudança em seus hiperparâmetros. Fonte: elaborado pela autora.

## 3.4 Treino e teste do modelo

Levando em consideração as três bases de dados selecionadas para uso neste trabalho, sete configurações diferentes foram determinadas para o treino e teste do modelo, compondo, deste modo, a etapa de experimentação prática do projeto. As três primeiras tratam-se das bases utilizadas sozinhas, sendo, portanto, apenas a base CK+, apenas a base *Extended and Augmented Google FER* e apenas a base *Natural Human Face Images for Emotion Recognition*.

Já as três configurações seguintes são definidas pela mistura de duas bases. Logo, tem-se a base CK+ com a *Extended FER*, a CK+ com a *Natural* e a *Extended FER* com a *Natural*. Finalmente, mistura-se as três bases em uma só, combinando as imagens de classes iguais.

Considerando a grande quantidade de dados nestas combinações, foi implementado um *callback* de *Early Stopping* no qual o treino seria interrompido caso a perda não melhorasse em até 2 épocas (*epochs*).

Ademais, realiza-se também um teste com o modelo *VGG16* definido na biblioteca *Keras* e outro teste com o modelo desenvolvido pela autora com a base de dados *MNIST* (composta por uma grande quantidade de dígitos, 0 a 9, desenhados à mão), visando observar se os resultados ruins observados durante a execução do código eram causados pelos dados ou pelos modelos. Deste modo, os resultados obtidos neste teste e nos apresentados anteriormente são salvos em um arquivo texto e apresentados em maiores detalhes no Capítulo 4.

# 4 Resultados

Este capítulo apresenta os resultados obtidos através da realização das etapas definidas no desenvolvimento, visando permitir uma análise adequada do desempenho dos modelos construídos. Sendo assim, as seções a seguir detalham e discutem os resultados provenientes dos testes de diferentes configurações dos bancos de dados e, finalmente, é feita a comparação de tais resultados com os que foram obtidos em trabalhos relacionados.

Além disso, destaca-se também que os resultados foram coletados usando um microcomputador com um processador Intel(R) Core(TM) i7-9700F CPU @ 3.00GHz, RAM de 16 GB e sistema operacional de 64 bits.

## 4.1 Resultados obtidos nos testes das bases de dados

Em primeiro lugar, foram feitos testes individuais em cada base de dados. Nestes testes iniciais, foi possível observar que os resultados obtidos eram extremamente insatisfatórios, apresentando valores de acurácia de treino bons, mas valores de acurácia de teste ruins, como, por exemplo, uma acurácia de apenas 30% para a base de dados *Extended FER*. Portanto, isto indicava que havia um problema de *overfit*, descrito como uma situação em que o modelo é incapaz de ser aplicado em outros dados além daqueles em que ele foi projetado e testado inicialmente, tal como observado neste contexto.

Deste modo, foram feitas várias mudanças visando melhorar a acurácia, tal como adicionar *Checkpoints* e *Early Stopping* ao modelo, aumentar o tamanho do *batch* de 3 para 32, aumentar a quantidade de épocas (*epochs*) para 20 e adicionar regularizadores. Entretanto, tais alterações não apresentaram uma melhora significativa nos resultados. Assim, foi implementado o modelo *VGG16* da biblioteca *Keras*, visando obter seus resultados e, através deles, analisar se o problema estava nos dados ou nos modelos utilizados.

Foi observado, então, que não houve nenhuma mudança significativa nos resultados, permanecendo a situação de *overfit* e indicando que havia algum problema nos dados. Com isto em mente, a função responsável pela leitura e pré-processamento dos dados foi substituída por uma função que fez uso da classe *Image Data Generator*, e os dados passaram a ser separados em diretórios de teste, treino e validação.

Com estas alterações no código, houve uma melhora nos resultados da acurácia dos testes. A título de exemplo, a base CK+ teve um aumento de acurácia de aproximadamente 30% para 50%. Todavia, considerando que estes resultados ainda não estavam adequados, outra mudança foi feita tentando melhorá-los. Esta mudança foi relacionada ao uso dos regularizadores, que tiveram seus valores alterados visando observar seus efeitos na acurácia do modelo. Assim, os regularizadores

L2 utilizados foram testados com os valores 0,001 e 0,0005, estes que apresentaram com a base CK+ acurácias de teste de 54,06% e 50,63%, respectivamente.

Tendo em vista o pequeno impacto da mudança anterior e visando confirmar novamente se o problema estaria nos dados ou no modelo desenvolvido, foi realizado um teste com a base de dados *MNIST*. Ao fim da execução deste teste, foi possível observar uma acurácia de teste de 99% e uma perda de 0,2338, condizente com os resultados obtidos nesta mesma base e divulgados no artigo original que apresenta a arquitetura *SimpleNet*.

Além disso, levando em conta que a execução deste teste foi feito por meio do uso da função *image_dataset_from_directory()*, ao invés de utilizar apenas a classe *Image Data Generator*, foi feito um teste com a base CK+ utilizando esta mesma função. Deste modo, foi possível obter uma acurácia de teste de 90,94%, o que indicou que as alterações feitas solucionaram o problema de *overfit*.

Portanto, com esta complicação resolvida, a última etapa antes da execução das diferentes configurações de testes foi a aplicação de *Data Augmentation* nas bases de treino, utilizando a classe *Image Data Generator* para gerar as imagens e salvá-las no diretório. Através de diversas experiências, foi possível observar uma melhora na acurácia de teste de 90,94% para 93,75% e, em outra execução, para 94,68% com a base CK+, e uma melhora de 33,09% para 46,09% com a base *Natural Human Face Images*, o que, deste modo, justifica a relevância desta etapa durante o desenvolvimento de um modelo de classificação.

Por fim, foi feito o teste das diferentes configurações de testes. Tais resultados, apresentados na tabela 4.1, demonstram que a base de dados CK+ sozinha possui os melhores valores de acurácia e perda, enquanto a base de dados *Natural Human Face Images* possui os piores valores de acurácia e de perda. Em relação às combinações das bases, é importante ressaltar que a utilização de *Early Stopping* pode ter influenciado negativamente no resultado, sendo isto evidenciado pela utilização deste *callback* na base de dados CK+, o que resultou em uma acurácia de teste de 86,25%.

Ademais, é essencial destacar também que não foi feita validação cruzada nos testes. Como apontado anteriormente, a configuração de teste com a base CK+ apresentou dois valores distintos de acurácia quando executada duas vezes, o que demonstra que a utilização de validação cruzada e o cálculo da média das acurácias obtidas nela possivelmente levariam a um resultado diferente daquele apresentado na Tabela 4.1, onde foi feito apenas uma execução de teste para cada configuração.

| **Configuração** | **Acurácia** | **Perda** | **Quantidade de dados** |
|---|---|---|---|
| CK | 0,9469 | 0,6920 | 15597 (train)<br>309 (val)<br>320 (test) |
| FER | 0,5331 | 1,7328 | 91774 (train)<br>3262 (val)<br>3271 (test) |
| Natural | 0,4609 | 2,0071 | 26402 (train)<br>552 (val)<br>562 (test) |
| CK + FER | 0,4909 | 1,8945 | 107371 (train)<br>3571 (val)<br>3591 (test) |
| CK + Natural | 0,6213 | 1,6778 | 41999 (train)<br>861 (val)<br>882 (test) |
| FER + Natural | 0,5451 | 1,5918 | 119176 (train)<br>3814 (val)<br>3271 (test) |
| CK + FER + Natural | 0,5295 | 1,7506 | 133773 (train)<br>3814 (val)<br>3271 (test) |

Tabela 4.1 – Comparação entre as diferentes configurações de bases de dados testadas. Fonte: elaborado pela autora.

## 4.2 Comparação com trabalhos relacionados

A escolha de trabalhos para realizar a comparação dos resultados foi feita a partir da utilização das mesmas bases de dados que aquelas utilizadas neste trabalho. Logo, a melhor acurácia obtida pelos modelos desenvolvidos nos artigos é apresentada na Tabela 4.2.

Analisando cada um dos trabalhos escolhidos e suas metodologias aplicadas durante a experimentação prática, observa-se, primeiramente, que o trabalho de Tautkute, Trzcinski e Bielski (2018) obteve uma acurácia de 73,60% em testes com 7 categorias de emoções para serem classificadas e uma acurácia de 92,10% em testes com 3 categorias (positiva, negativa e neutra). Além disso, é importante destacar que o treino do modelo desenvolvido foi feito usando a base *AffectNet*, enquanto o teste foi feito usando a base CK+.

Em segundo lugar, tem-se o trabalho de Jain, Shamsolmoali e Sehdev (2019), com 93,24% de acurácia em seu modelo. Analisando os trabalhos que vieram posteriormente a este, foi possível encontrar o artigo de Fei et al. (2020), que alcançou 94,70% de acurácia na base CK+ utilizando o modelo *AlexNet* (extraindo características da sua *Fully Connected Layer 6*) junto com *Linear Discriminant Analysis Classifier* (LDA).

Em relação ao trabalho de Mehendale (2020), foi alcançado 96% de acurácia a partir de uma validação cruzada de 25 folds, todavia a base de dados CK+ foi complementada com outras bases, tendo em vista que o autor alcançou apenas 45% de acurácia ao utilizar apenas esta base. Já dentre os trabalhos posteriores a este, destaca-se o de Karanchery e Palaniswamy (2021) e o de Kumari e Bhatia (2022).

O primeiro foi capaz de alcançar 99,80% de acurácia, porém foi treinado com os usuários do sistema de reconhecimento. Já o segundo alcançou 98,01% na base CK+, porém seu desenvolvimento contou não apenas com a utilização de um modelo CNN para a classificação, como também foram feitas outras etapas durante o pré-processamento, como aplicação de CLAHE (*contrast-limited adaptive histogram equalization*) para aumentar a visibilidade das imagens.

Por último, o trabalho de Ribeiro (2018) apresentou como resultado para base CK+ 97% de acurácia. Para tanto, o autor fez um processo de meta-aprendizagem para a base, com treinamento de 100 épocas e um otimizador realizando 10 avaliações no modelo da arquitetura *VGG-16*.

| **TRABALHO** | **MELHOR ACURÁCIA** | **MODELO** |
|---|---|---|
| TAUTKUTE; TRZCINSKI; BIELSKI, 2018 | 73,6% | CNN (DAN) |
| JAIN; SHAMSOLMOALI; SEHDEV, 2019 | 93,24% | DNN |
| MEHENDALE, 2020 | 96% | CNN (FERC) |
| RIBEIRO, 2018 | 97% | CNN (VGG-16 otimizada) |
| **ESTE TRABALHO** | **94,69%** | **CNN (SimpleNet otimizado)** |

Tabela 4.2 – Comparação entre a melhor acurácia dos trabalhos relacionados e do trabalho proposto. Fonte: elaborado pela autora.

Tendo em vista as observações apresentadas nesta seção, conclui-se que o modelo desenvolvido e exposto nesta Monografia possui um desempenho apropriado quando comparado com outros modelos desenvolvidos, sendo superior aos de Tautkute, Trzcinski e Bielski (2018) e Jain, Shamsolmoali e Sehdev (2019), mas inferior aos de Mehendale (2020) e Ribeiro (2018). Deste modo, tendo como base as metodologias utilizadas na elaboração destes outros modelos, é possível melhorar este desempenho através da utilização de outras formas de pré-processamento e de outros dados, como, por exemplo, a base *AffectNet*.

# 5 Considerações Finais

Este capítulo explicita as conclusões obtidas ao final deste trabalho, descrevendo sucintamente quais objetivos foram alcançados e quais resultados foram obtidos na Seção 5.1. Ademais, o capítulo apresenta também na Seção 5.2 as propostas para continuação deste trabalho, tendo em vista o que foi alcançado pelo mesmo até o presente momento.

## 5.1 Conclusões

O estabelecimento de métricas adequadas para a avaliação e desenvolvimento de sistemas interativos para o ensino se faz cada vez mais necessário, tendo em vista seu crescente uso em salas de aulas presenciais e virtuais. Tais métricas, como, por exemplo, formulários de *feedback*, são conjuntos de dados que permitem a análise do desempenho e uso, e devem ser elaboradas considerando o contexto em que serão utilizadas, evitando o desconforto dos usuários e sendo fáceis de serem obtidas e usadas. Portanto, ferramentas que se baseiam apenas no reconhecimento de estados emocionais por meio de expressões faciais se apresentam como soluções para esta problemática.

Tendo isto em vista, este trabalho teve como objetivo principal desenvolver um modelo de rede neural capaz de realizar o reconhecimento de emoções por meio de expressões. Através da metodologia apresentada neste documento, foi possível desenvolver modelos base de redes neurais convolucionais para a classificação das emoções e realizar testes preliminares nestes, constatando qual a melhor arquitetura e obtendo também seus melhores hiperparâmetros. Em seguida, foram feitos testes com diferentes configurações de bases de dados, visando observar qual apresentaria o melhor desempenho.

Os testes revelaram, tal como descrito em mais detalhes na Seção 4.1, que a melhor configuração de bases de dados para o modelo desenvolvido neste trabalho utiliza apenas a base CK+, alcançando uma acurácia de teste de 94,69%. Ademais, os testes também demonstraram a importância da implementação adequada da leitura e pré-processamento dos dados, considerando o seu impacto no desempenho do modelo.

Deste modo, constata-se que os objetivos definidos foram alcançados com sucesso. Ademais, o trabalho também apresentou em seus capítulos anteriores trabalhos relacionados importantes, como o de Ribeiro (2018) e de Jain, Shamsolmoali e Sehdev (2019), além de uma fundamentação teórica que descreve de modo breve, porém suficientemente completo, os principais conceitos necessários para seu entendimento.

Assim, infere-se que o modelo desenvolvido possui um bom desempenho e arquitetura, podendo ser utilizado no reconhecimento de emoções com bons resultados. Além disto, destaca-

se também que este trabalho contribui para o meio acadêmico através da exposição de uma metodologia em que se evidencia a importância da otimização de modelos por meio da seleção automática de hiperparâmetros e a importância de uma pesquisa por diferentes arquiteturas e bases de dados, de modo a identificar qual seria a mais apropriada para a situação em que ela será utilizada.

## 5.2 Trabalhos Futuros

Em relação aos trabalhos futuros, considerando que o modelo de classificação de emoções foi elaborado com sucesso, destaca-se, primeiramente, a possibilidade de implementação de um sistema de reconhecimento de rostos e emoções por *webcam*. Para esta finalidade, recomenda-se a utilização da biblioteca *open-source* (código aberto) *OpenCV*, voltada ao desenvolvimento de aplicações na área de Visão Computacional. Logo, o módulo de entrada e saída de vídeo (*Video I/O*) seria usado para capturar o vídeo proveniente da *webcam* enquanto o usuário não pressiona a tecla Q de seu teclado.

Cada imagem estática que compõem o vídeo (*frame*) é convertida para a escala de cinza e os rostos nela são detectados através do modelo de classificação *Cascade Classifier*. Para cada um destes rostos, obtém-se os valores de sua *bounding box*, retângulo que enquadra o rosto e é composto por x, y, largura e altura. Em seguida, o *frame* é redimensionado para 48 x 48 *pixels*, e os atributos da *bounding box* são utilizados para separar o rosto do *frame* inteiro.

Por último, o modelo de classificação de emoções seria aplicado sobre o *frame* cortado, retornando a emoção identificada. Esta emoção pode ser escrita na tela por meio da biblioteca *OpenCV*, e os relatórios gerados. Para a criação de relatórios do uso do sistema, recomenda-se a biblioteca *PyAutoGUI*, que permite controlar mouse e teclado por meio de código.

Para cada emoção observada, utiliza-se uma operação desta biblioteca para capturar a tela (*screenshot*) e uma operação da biblioteca *OpenCV* para salvar esta captura em um diretório que seria criado apenas para armazenar essas imagens. Após, a emoção é adicionada ao final de uma lista e, no fim do uso do sistema, esta lista é escrita em um arquivo de tipo texto (.txt) que seria salvo dentro do diretório de relatórios.

Com este sistema funcionando, haveria a possibilidade de desenvolver uma interface gráfica para ele, facilitando, assim, seu uso para o usuário padrão. Ademais, sugere-se também a expansão dos relatórios gerados, fornecendo ao usuário não apenas as emoções observadas durante o uso do sistema, mas também gráficos acerca destas emoções.

Para a elaboração de uma interface gráfica, existem várias bibliotecas disponíveis. No contexto deste trabalho, recomenda-se a *PySimpleGUI*, tendo em vista sua maior facilidade de uso. Já para a produção de diferentes tipos de relatórios, recomenda-se a *Matplotlib*, a biblioteca mais conhecida para a elaboração de gráficos.

Todavia, é possível constatar também que apesar do desempenho apropriado, o modelo pode ser melhorado. Primeiramente, os hiperparâmetros do modelo podem ser encontrados para a base de dados CK+, ao invés da versão reduzida utilizada anteriormente no trabalho. Já em segundo lugar, o pré-processamento pode ser expandido, adicionando novas etapas, tal como o uso de *CLAHE*, e outras bases de dados podem ser também utilizadas, como, por exemplo, a *AffectNet*. Além disso, a utilização de validação cruzada também é importante, já que permitiria uma melhor análise do desempenho do modelo.

# Referências


BATISTA, G. E. d. A. P. et al. *Pré-processamento de dados em aprendizado de máquina supervisionado*. Tese (Doutorado) — Universidade de São Paulo, 2003.

BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. *Journal of machine learning research*, v. 13, n. 2, 2012.

DONAHUE, J.; HENDRICKS, L. A.; GUADARRAMA, S.; ROHRBACH, M.; VENUGOPALAN, S.; SAENKO, K.; DARRELL, T. Long-term recurrent convolutional networks for visual recognition and description. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 2625–2634.

DRIMALLA, H.; BASKOW, I.; BEHNIA, B.; ROEPKE, S.; DZIOBEK, I. Imitation and recognition of facial emotions in autism: a computer vision approach. *Molecular autism*, Springer, v. 12, n. 1, p. 1–15, 2021.

EKMAN, P.; FRIESEN, W. V. *Unmasking the face: A guide to recognizing emotions from facial clues*. [S.l.]: Ishk, 2003. v. 10.

FEI, Z.; YANG, E.; LI, D. D.-U.; BUTLER, S.; IJOMAH, W.; LI, X.; ZHOU, H. Deep convolution network based emotion analysis towards mental health care. *Neurocomputing*, Elsevier, v. 388, p. 212–227, 2020.

FREIRE, P. et al. *A importância do ato de ler*. [S.l.]: Moderna, 2003.

GHOFRANI, A.; TOROGHI, R. M.; GHANBARI, S. Realtime face-detection and emotion recognition using mtcnn and minishufflenet v2. In: IEEE. *2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)*. [S.l.], 2019. p. 817–821.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>.

HAO, X.; ZHANG, G.; MA, S. Deep learning. *International Journal of Semantic Computing*, World Scientific, v. 10, n. 03, p. 417–439, 2016.

HASANPOUR, S. H.; ROUHANI, M.; FAYYAZ, M.; SABOKROU, M. Lets keep it simple, using simple architectures to outperform deeper and more complex architectures. *arXiv preprint arXiv:1608.06037*, 2016.

HAYKIN, S. *Redes neurais: princípios e prática*. [S.l.]: Bookman Editora, 2001.

JAIN, D. K.; SHAMSOLMOALI, P.; SEHDEV, P. Extended deep neural network for facial emotion recognition. *Pattern Recognition Letters*, Elsevier, v. 120, p. 69–74, 2019.

JANIESCH, C.; ZSCHECH, P.; HEINRICH, K. Machine learning and deep learning. *Electronic Markets*, Springer, v. 31, n. 3, p. 685–695, 2021.

KANADE, T.; COHN, J. F.; TIAN, Y. Comprehensive database for facial expression analysis. In: IEEE. *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. No. PR00580)*. [S.l.], 2000. p. 46–53.

KARANCHERY, S.; PALANISWAMY, S. Emotion recognition using one-shot learning for human-computer interactions. In: IEEE. *2021 International conference on communication, control and information sciences (ICCISc)*. [S.l.], 2021. v. 1, p. 1–8.

KO, B. C. A brief review of facial emotion recognition based on visual information. *sensors*, MDPI, v. 18, n. 2, p. 401, 2018.

KUMARI, N.; BHATIA, R. Efficient facial emotion recognition model using deep convolutional neural network and modified joint trilateral filter. *Soft Computing*, Springer, v. 26, n. 16, p. 7817–7830, 2022.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015.

LISBOA, A.; YEHIA, H. C.; NASCIMENTO, C. D.; MAGALHãES, H.; VENâNCIO, P.; NETO, A. Interação com a sociedade por meio de sistema de visão computacional para monitoramento ambiental de linhas de transmissão. In: . [S.l.: s.n.], 2019.

LIU, J.; SU, Y.; LIU, Y. Multi-modal emotion recognition with temporal-band attention based on lstm-rnn. In: SPRINGER. *Pacific rim conference on multimedia*. [S.l.], 2017. p. 194–204.

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: IEEE. *2010 ieee computer society conference on computer vision and pattern recognition-workshops*. [S.l.], 2010. p. 94–101.

MATSUMOTO, D. More evidence for the universality of a contempt expression. *Motivation and Emotion*, Springer, v. 16, n. 4, p. 363–368, 1992.

MEHENDALE, N. Facial emotion recognition using convolutional neural networks (ferc). *SN Applied Sciences*, Springer, v. 2, n. 3, p. 1–8, 2020.

RANGANATHAN, H.; CHAKRABORTY, S.; PANCHANATHAN, S. Multimodal emotion recognition using deep learning architectures. In: IEEE. *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. [S.l.], 2016. p. 1–9.

RIBEIRO, J. L. M. Meta-aprendizado aplicado ao problema de reconhecimento de expressões faciais. Universidade Federal do Maranhão, 2018.

SANT'ANNA, A.; NASCIMENTO, P. R. A história do lúdico na educação. *REVEMAT: Revista Eletrônica de matemática*, Universidade do Extremo Sul Catarinense, v. 6, n. 2, p. 19–36, 2011.

SHINDE, P. P.; SHAH, S. A review of machine learning and deep learning applications. In: IEEE. *2018 Fourth international conference on computing communication control and automation (ICCUBEA)*. [S.l.], 2018. p. 1–6.

SILVA, R. H. H. Análise facial para detecção de estresse e tédio em jogos digitais. Universidade Federal da Fronteira Sul, 2022.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

SINHA, J. *Realtime Facial Expression Recognition using geometric analysis*. 2019. <https://github.com/jitensinha98/Realtime-Facial-Expression-Recognition-using-geometric-analysis>.

TAUTKUTE, I.; TRZCINSKI, T.; BIELSKI, A. I know how you feel: Emotion recognition with facial landmarks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. [S.l.: s.n.], 2018. p. 1878–1880.

ZHENG, X. Research on video emotion recognition based on attention mechanism lstm model. In: ATLANTIS PRESS. *2018 International Conference on Network, Communication, Computer Engineering (NCCE 2018)*. [S.l.], 2018. p. 894–898.