

## **Análise comparativa entre redes neurais convolucionais e o ChatGPT-4 em termos de desempenho, custo e tempo de processamento na classificação de imagens**

Andrei Lima Alves dos Santos<sup>[1]</sup>, Daniel Carneiro Rosa<sup>[2]</sup>, Álvaro Alvares de Carvalho César Sobrinho<sup>[3]\*</sup>, Thiago Damasceno Cordeiro<sup>[4]</sup>, Diego Dermeval Medeiros da Cunha Matos<sup>[3]</sup>

<sup>[1]</sup> [andreiMD02@gmail.com](mailto:andreiMD02@gmail.com), <sup>[3]</sup> [alvaro.alvares@ufape.edu.br](mailto:alvaro.alvares@ufape.edu.br). Núcleo de Inovação Tecnológica e Empreendedorismo, Universidade Federal do Agreste de Pernambuco (UFape), Garanhuns, Pernambuco, Brasil

<sup>[2]</sup> [dan.rosa10@gmail.com](mailto:dan.rosa10@gmail.com). Universidade Federal Rural de Pernambuco (UFRPE), Recife, Pernambuco, Brasil

<sup>[4]</sup> [thiago@ic.ufal.br](mailto:thiago@ic.ufal.br), <sup>[5]</sup> [diego.matos@famed.ufal.br](mailto:diego.matos@famed.ufal.br). Universidade Federal de Alagoas (UFAL), Maceió, Alagoas, Brasil

\* autor correspondente

### **Resumo**

Este estudo apresenta uma comparação entre os resultados obtidos de Redes Neurais Convolucionais (*Convolutional Neural Networks* – CNNs) e ChatGPT-4 na classificação de imagens com ou sem desfoque. Comparar CNNs com ChatGPT-4 é relevante devido ao uso crescente de grandes modelos de linguagem na sociedade, que têm sido amplamente aplicados em vários domínios, incluindo tarefas de análise de imagens. O problema de classificação abordado é multiclasse, abrangendo imagens nítidas, desfoque de movimento e desfoque devido ao desfoque. Classificar imagens com ou sem desfoque apresenta um desafio devido às semelhanças entre diferentes tipos de desfoque e à presença de desfoques leves, que podem ser difíceis de detectar. As CNNs foram treinadas e testadas usando um conjunto de dados público do Kaggle, e os mesmos conjuntos de dados foram usados para avaliar o ChatGPT-4, acompanhados por consultas cuidadosamente projetadas. O tamanho do conjunto de dados variou entre 1.050 e 8.100 imagens, totalmente balanceado entre as classes. A metodologia incluiu pré-processamento de dados com transformadas de Fourier e Haar, implementação do modelo CNN, validação do modelo e comparação com resultados obtidos usando ChatGPT-4. Essa abordagem permitiu investigar qual método fornece o melhor desempenho, custo e tempo de processamento. Após experimentar todo o conjunto de dados, o modelo CNN com o melhor desempenho usou a transformada de Haar, alcançando uma precisão média de 0,8889, uma precisão média de 0,8913, um recall médio de 0,9503 e uma *Area sob a Curva* (*Area Under the Curve* – AUC) média de 0,9503. O melhor resultado obtido com ChatGPT-4 mostrou uma precisão média de 0,8198, uma precisão média de 0,8175, um recall médio de 0,8198 e uma AUC média de 0,8649. Este tipo de análise é de grande relevância, dado o potencial do ChatGPT-4 como uma ferramenta de suporte à decisão em vários campos, como saúde e educação. Este estudo contribui para o estado da arte ao fornecer insights valiosos para pesquisadores e empresas que visam otimizar seus processos de análise de imagens usando inteligência artificial. Trabalhos futuros devem incluir uma análise mais aprofundada dos algoritmos usados, estudos controlados e a expansão do conjunto de dados.

**Palavras-chave:** aprendizado profundo; classificação de imagens; classificação multiclasse; CNNs; LLM.

### ***Comparative analysis between convolutional neural networks and ChatGPT-4 in terms of performance, cost, and processing time in image classification***

#### ***Abstract***

*This study presents a comparison between the results obtained from Convolutional Neural Networks (CNNs) and ChatGPT-4 in the classification of images with or without blurring. Comparing CNNs with ChatGPT-4 is relevant due to the increasing use of large language models in society, which have been widely applied in various domains, including image analysis tasks. The classification problem*

addressed is multi-class, encompassing sharp images, motion blur, and blur due to defocus. Classifying images with or without blur presents a challenge due to the similarities between different types of blur and the presence of light blurs, which can be difficult to detect. The CNNs were trained and tested using a public dataset from Kaggle, and the same datasets were used to evaluate ChatGPT-4, accompanied by carefully designed queries. The dataset size varied between 1,050 and 8,100 images, fully balanced across the classes. The methodology included data preprocessing with Fourier and Haar transforms, CNN model implementation, model validation, and comparison with results obtained using ChatGPT-4. This approach allowed for investigating which method provides the best performance, cost, and processing time. After experimenting with the entire dataset, the CNN model with the best performance used the Haar transform, achieving an average accuracy of 0.8889, an average precision of 0.8913, an average recall of 0.9503, and an average Area Under the Curve (AUC) of 0.9503. The best result obtained with ChatGPT-4 showed an average accuracy of 0.8198, an average precision of 0.8175, an average recall of 0.8198, and an average AUC of 0.8649. This type of analysis is of great relevance, given the potential of ChatGPT-4 as a decision support tool in various fields such as healthcare and education. This study contributes to the state of the art by providing valuable insights for researchers and companies aiming to optimize their image analysis processes using artificial intelligence. Future work should include a more in-depth analysis of the algorithms used, controlled studies, and the expansion of the dataset.

**Keywords:** CNNs; deep learning; image classification; LLM; multiclass classification.

## 1 Introdução

Com o aumento da geração de imagens digitais, impulsionado pelo uso crescente de dispositivos móveis e câmeras de alta resolução, novos desafios têm surgido, como a identificação precisa de imagens borradas (Chen, 2023). Esse problema pode afetar negativamente diversos serviços, produtos e aplicações que dependem de imagens de alta qualidade. A redução da legibilidade das imagens compromete a interpretação precisa do conteúdo representado. Na área da educação, por exemplo, imagens de baixa qualidade em livros didáticos podem prejudicar o aprendizado dos estudantes, especialmente daqueles com limitações visuais (Singh; Zouhar; Sachan, 2023; Sobrinho *et al.*, 2023). Na área de segurança, a identificação de borras pode auxiliar na detecção de edições incorretas ou tentativas de mascarar conteúdo. Na medicina, borras podem comprometer o diagnóstico médico por meio de imagens. No entretenimento, a identificação de borras pode ajudar a detectar gravações que necessitam de melhorias.

Até o momento, redes neurais convolucionais (*Convolutional Neural Networks* – CNNs) têm sido amplamente utilizadas como principal ferramenta para classificação de imagens (Silva *et al.*, 2023). Baseadas em arquiteturas de aprendizado profundo, as CNNs demonstram elevada eficácia em tarefas de classificação. Contudo, o treinamento desses modelos é complexo, exigindo grandes volumes de dados e recursos computacionais significativos, o que pode limitar sua aplicabilidade em ambientes com recursos restritos (Lee *et al.*, 2019). Além disso, a personalização das CNNs para lidar com diferentes tipos de borras e cenários específicos demanda esforço substancial, o que compromete a escalabilidade da abordagem.

Nesse contexto, os avanços nos modelos de linguagem, como o ChatGPT (Bahrini *et al.*, 2023), têm aberto novas possibilidades. Desenvolvido pela OpenAI, o ChatGPT é um modelo de linguagem de grande porte (*Large Language Model* – LLM), baseado na arquitetura Transformer (Brown *et al.*, 2020). Treinado em um vasto conjunto de dados, o ChatGPT é capaz de executar tarefas como análise textual, tradução, criação de conteúdos diversos e, mais recentemente, processamento de imagens. Dado que o ChatGPT-4 possibilita que usuários enviem imagens para diferentes tipos de análises, é natural que tarefas como a classificação de imagens com ou sem borras sejam consideradas.

O presente trabalho tem como objetivo investigar o impacto do uso do ChatGPT-4 em comparação com uma implementação específica de CNN na classificação de imagens contendo dois tipos de borras comumente identificadas em diferentes contextos: *motion blur* e *defocused blur*. Embora o ChatGPT não tenha sido desenvolvido para solucionar problemas relacionados a imagens, como destacado anteriormente, suas funcionalidades permitem que usuários carreguem e processem

diferentes tipos de imagens, o que o torna uma alternativa interessante para tarefas de classificação. Desde seu lançamento, o ChatGPT tem sido explorado em diversos contextos, como educação (Adeshola; Adepoju, 2023) e saúde (Li *et al.*, 2024). O ChatGPT-4 se torna atrativo aos usuários finais pela facilidade na criação de consultas, com o potencial de reduzir a necessidade de conhecimento especializado.

Espera-se, com esta análise comparativa, identificar qual abordagem apresenta os melhores resultados em termos de desempenho, custo e tempo de processamento. Este estudo contribui para o estado da arte, fornecendo subsídios para pesquisadores e empresas que busquem otimizar seus processos de análise de imagens utilizando inteligência artificial. Além disso, visa contribuir para a literatura ao avaliar se modelos de linguagem de grande porte, como o ChatGPT, podem desempenhar tarefas tradicionalmente atribuídas a CNNs, considerando os desafios e as oportunidades inerentes a cada abordagem.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica e os trabalhos relacionados a este estudo. Na Seção 3, detalha-se a metodologia empregada. Os resultados e discussões são apresentados na Seção 4. Por fim, a Seção 5 traz as considerações finais.

## 2 Trabalhos relacionados

As CNNs constituem um modelo de aprendizado profundo desenvolvido especificamente para o processamento de imagens. Essas redes são compostas por múltiplas camadas, cada uma responsável por realizar operações de convolução entre os dados de entrada e um conjunto de filtros (ou *kernels*) treinados pelo modelo. A saída de cada camada é transferida para a camada subsequente, onde novas operações de convolução ou transformações não lineares são realizadas.

Estudos prévios investigaram a classificação de imagens desfocadas utilizando técnicas de aprendizado de máquina. Por exemplo, Wang, Li e Zhang (2019) empregaram CNNs para classificar imagens em categorias como desfoque gaussiano, neblina e movimento. Seguindo abordagem semelhante, Wang *et al.* (2019) utilizaram máquinas de vetores de suporte para classificar conjuntos de dados contendo imagens nítidas, desfocadas por movimento, desfoque gaussiano e neblina. Este trabalho baseia-se, em parte, nos achados de Tiwari (2020), que abordou a classificação de gestos de mão em imagens desfocadas, utilizando um conjunto de dados cuja similaridade reduziu a complexidade da classificação. Tiwari explorou ainda a transformada de Fourier para o pré-processamento dos dados, conforme descrito por Rao, Kim e Hwang (2010).

Adicionalmente, diversos estudos recentes têm investigado a utilização do ChatGPT e sua aceitação como ferramenta de suporte, evidenciando a relevância desta pesquisa. Exemplos incluem os trabalhos de AlGhamdi (2024), Sobaih *et al.* (2024), Abdaljaleel *et al.* (2024), Xu e Thien (2024) e Naz e Robertson (2024). O ChatGPT, como uma variante dos modelos de linguagem *Generative Pretrained Transformer* (GPT), fundamenta-se em conceitos de aprendizado profundo e processamento de linguagem natural. A arquitetura dos *transformers*, introduzida por Vaswani *et al.* (2017), é a base para esses modelos. O GPT, por sua vez, caracteriza-se como um modelo auto-regressivo, com elementos comuns como *embeddings*<sup>1</sup>, codificação posicional<sup>2</sup>, camadas de *transformer*<sup>3</sup> e uma camada de saída.

O ChatGPT-4, em particular, foi treinado utilizando aprendizado não supervisionado em um vasto conjunto de dados. Essa variante do GPT é um modelo de grande escala, com bilhões de parâmetros. A partir de uma entrada (*prompt*), utiliza informações contextuais para gerar sequências textuais. O treinamento desse modelo requer uma infraestrutura computacional robusta. Além disso, o ChatGPT-4 suporta múltiplos tipos de entrada, incluindo texto, imagens e dados multimodais, o que amplia suas possibilidades de aplicação.

---

<sup>1</sup> Vetor numérico que representa um elemento de texto, como uma palavra ou sentença.

<sup>2</sup> Introduz informações sobre a ordem de palavras ou subpalavras.

<sup>3</sup> Projetadas para processar e modelar sequências de texto.

### 3 Método da pesquisa

Na etapa inicial deste estudo, realizou-se uma revisão tradicional da literatura, com o objetivo de analisar trabalhos voltados à classificação de imagens desfocadas. Essa revisão permitiu compreender o estado da arte na área. Em seguida, foram conduzidos experimentos para uma análise comparativa entre modelos de CNNs e o ChatGPT-4. Um modelo foi implementado com base em CNNs profundas, e os resultados obtidos foram comparados, sempre que possível, com as saídas geradas pelo ChatGPT-4. Em alguns casos, tornou-se necessário executar localmente os códigos fornecidos pelo ChatGPT-4 para a obtenção dos resultados.

Este estudo adotou uma arquitetura distinta daquela proposta por Tiwari (2020). As configurações empregadas nas camadas de convolução e *max-pooling* foram as seguintes:

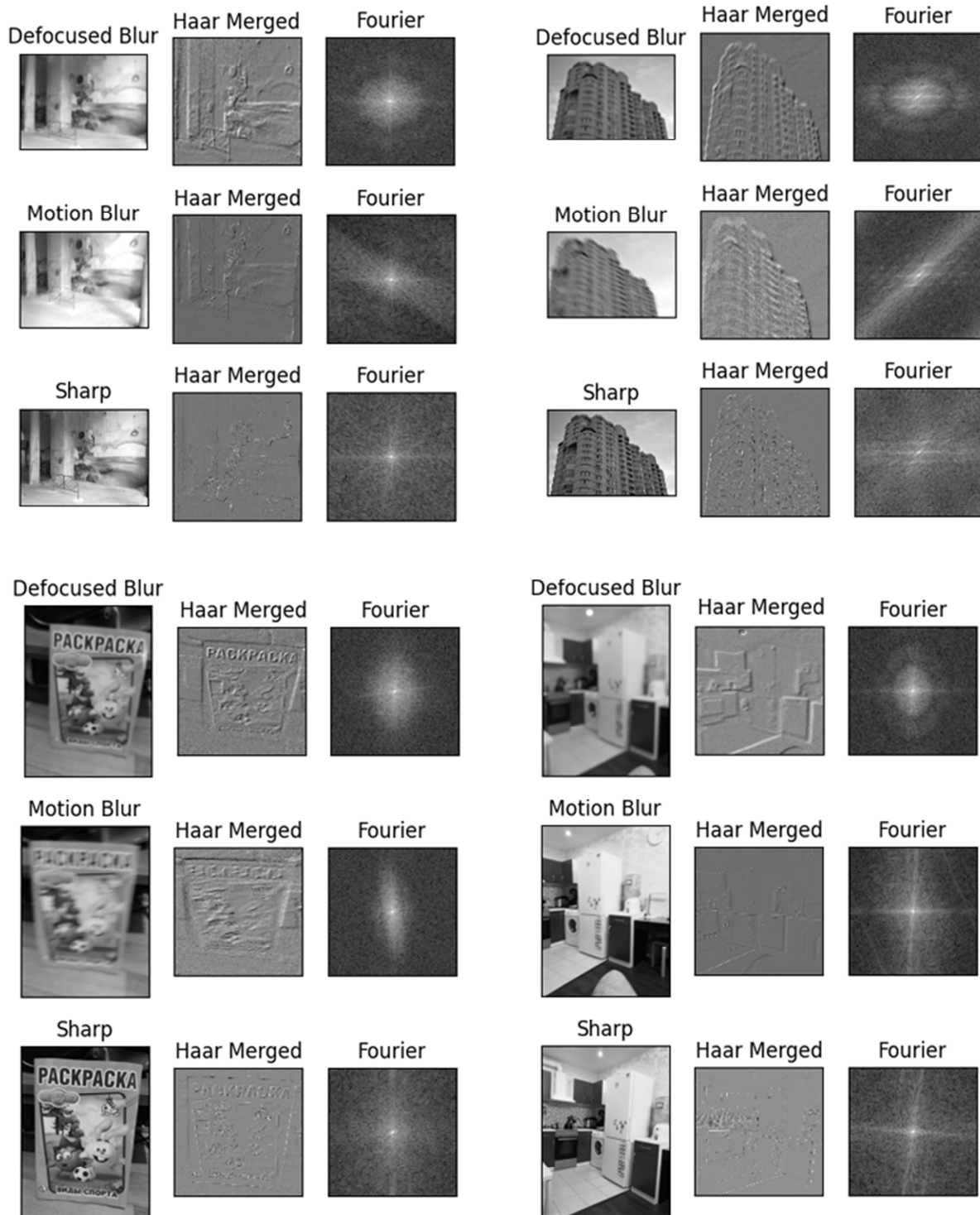
- A primeira camada de convolução contém 32 filtros de convolução de  $3 \times 3$ ;
- A segunda camada de convolução também possui 32 filtros de convolução de  $3 \times 3$ , seguida por uma camada de *max-pooling* com janela  $2 \times 2$ ;
- A terceira camada de convolução é composta por 64 filtros de  $3 \times 3$ , igualmente seguida por *max-pooling* com janela  $2 \times 2$ ;
- A quarta camada de convolução apresenta 128 filtros de  $3 \times 3$ , novamente sucedida por uma camada de *max-pooling*;
- Após as camadas de convolução, foram adicionadas duas camadas lineares, compostas por 1024 e 512 neurônios, respectivamente, sendo cada uma seguida por *Dropout* e *Batch Normalization*. A camada de *Dropout* é amplamente utilizada em arquiteturas de CNN, pois ajuda a prevenir o *overfitting*. Ela atua desativando aleatoriamente alguns neurônios da camada em que foi inserida durante o treinamento, de forma temporária. A camada de *Batch Normalization* é utilizada para aplicar transformações nas saídas da camada anterior, realizadas em *mini-batches*, para normalização dos dados. Essa técnica torna o processo de treinamento mais estável, além de auxiliar no aumento do desempenho dos modelos de CNN;
- Por fim, foi incluída uma camada linear com três saídas, representando as classes do problema.

Optou-se por utilizar uma entrada de tamanho  $120 \times 120$ , contrastando com o estudo anterior, que utilizou imagens de  $32 \times 32$ . Os experimentos indicaram que o tamanho menor não proporcionou melhorias significativas no desempenho do modelo aplicado ao conjunto de dados deste estudo.

Antes das experimentações, definiu-se uma etapa de pré-processamento dos dados. Foram investigados cenários que utilizaram as transformadas de Haar e Fourier. A Figura 1 apresenta exemplos dessas transformações aplicadas às imagens do conjunto de dados. Observa-se que, mesmo sob diferentes tipos de desfoque, as transformações evidenciam distinções perceptíveis entre as imagens, ilustrando a aplicação prática das técnicas no conjunto de dados utilizado.

Figura 1 – Amostras do conjunto de dados Blur do Kaggle com as transformadas de Fourier e Haar aplicadas. As amostras mescladas de Haar representam a soma entre as matrizes de coeficientes verticais e horizontais da transformada wavelet de Haar





Fonte: arquivo dos autores

Para esses experimentos, utilizou-se o conjunto de dados *Blur Dataset* disponível no Kaggle<sup>4</sup>. Esse conjunto contém 1.050 imagens organizadas em 350 trios, abrangendo classes como imagens nítidas, borras por desfoque e borras por movimento. As imagens desfocadas foram geradas a partir de

<sup>4</sup> Disponível em: <https://www.kaggle.com/datasets/kwentar/blur-dataset>.

algoritmos aplicados a imagens nítidas. Para o problema de classificação multiclasse, as três classes possuem quantidades equilibradas de 350 imagens cada. Além disso, o conjunto de dados original foi expandido de 1.050 para 8.100 imagens, enriquecendo a análise experimental.

Por fim, experimentou-se com as transformadas de Fourier e Haar, conforme descrito por Li *et al.* (2020), aumentando a complexidade da tarefa de classificação e utilizando um conjunto público diferente de estudos anteriores.

## 4 Resultados e discussões

Nesta seção, apresentam-se e discutem-se os procedimentos realizados e os resultados obtidos relacionados ao pré-processamento de dados e às tarefas de classificação. Embora as experimentações tenham considerado a variação do tamanho do conjunto de dados entre 1.050 e 8.100 imagens, os procedimentos realizados podem ser escalados em trabalhos futuros para conjuntos de dados maiores e aplicados a diferentes cenários. Além disso, as imagens utilizadas nos experimentos não estavam vinculadas a nenhum cenário de aplicação específico, o que reforça a relevância e a capacidade de generalização dos resultados obtidos.

### 4.1 Pré-processamento de dados

Antes do início da análise comparativa, foi necessário ajustar os dados utilizados para o treinamento e o teste das redes. Para cada conjunto de dados, aplicou-se um algoritmo que separou aleatoriamente a base em conjuntos de treinamento, teste e validação. Em seguida, as imagens foram salvas para cada um desses conjuntos. Assim, para a primeira base de dados, utilizaram-se as mesmas imagens para treinamento e teste nas redes com pré-processamento Haar, Fourier e para o envio ao ChatGPT-4.

As imagens foram redimensionadas para  $224 \times 224$  pixels antes de serem enviadas ao ChatGPT-4, procedimento também adotado no treinamento das redes neurais. Esse redimensionamento teve como objetivos padronizar os dados, garantir a consistência dos resultados e atender ao limite de 100 MB imposto pela ferramenta. Entretanto, essa limitação não impactou as comparações, pois o ChatGPT-4 possibilita a realização de diferentes consultas e a geração de código para execução local pelo usuário.

Adicionalmente, realizou-se o pré-processamento das imagens antes de enviá-las ao ChatGPT-4. As imagens foram redimensionadas para  $224 \times 224$  pixels, conforme o padrão adotado previamente para o treinamento das redes. A redimensionalização foi efetuada tanto para os conjuntos de treinamento quanto para os de teste. Esse procedimento visou padronizar os dados, garantir uma avaliação justa e reduzir o tamanho da base, tendo em vista que o ChatGPT-4 possui um limite de arquivo de 100 MB.

Após o redimensionamento, tornou-se possível enviar as imagens ao ChatGPT-4 e obter suas respostas. Ressalta-se que, mesmo utilizando o ChatGPT-4 para obter os resultados, ainda foi necessário realizar alguns pré-processamentos devido às limitações no tamanho da base de dados. Essa restrição pode influenciar a tomada de decisões em aplicações corporativas, considerando que, frequentemente, bases de dados são muito maiores do que o limite suportado pela ferramenta, o que torna a redução e compactação das imagens uma tarefa desafiadora.

### 4.2 Avaliação sem geração de dados

Para comparar os resultados das redes neurais convolucionais (utilizando transformadas no pré-processamento) com os obtidos pelo ChatGPT-4, utilizou-se uma base de dados contendo 1050 imagens para cada uma das seguintes classes: *sharp*, *motion blur* e *defocused blur*. Inicialmente, todas as imagens foram misturadas em uma única base e, em seguida, divididas em três conjuntos: treinamento, validação e teste.

Os conjuntos foram formados com as seguintes proporções: 72,25% para treinamento, 15% para validação e 12,75% para teste. A seleção das imagens foi realizada aleatoriamente, assegurando-se o registro preciso de quais imagens compunham cada conjunto, o que garantiu a comparabilidade entre os modelos de redes neurais convolucionais e os resultados do ChatGPT-4.

No primeiro passo, aplicou-se a transformada de Haar à base de dados e realizou-se o treinamento de redes neurais convolucionais durante 30 épocas. Foi realizado um teste com mais épocas (100); entretanto, ao acompanhar o treinamento dos modelos, verificou-se que eles convergiam pouco antes das 20 épocas. Estender para 100 épocas seria desnecessário, pois, usualmente, se fosse inserida uma parada precoce de 10 épocas, o treino seria finalizado aproximadamente na 30ª época. Posteriormente, repetiu-se o procedimento com a transformada de Fourier. Após o treinamento, a base de teste foi utilizada para calcular as métricas de desempenho, aplicando-se a transformada correspondente nas imagens de teste. As métricas avaliadas incluíam acurácia, precisão, *recall* e Área Sob a Curva (*Area Under the Curve* – AUC). Os resultados obtidos estão apresentados na Tabela 1. Os resultados são quantificados entre 0 e 1, indicando que um modelo com desempenho próximo a 1 possui quase 100% de acurácia, precisão, *recall* ou AUC.

Tabela 1 – Comparação de desempenho entre os modelos treinados com base no pré-processamento usando as transformadas de Haar e Fourier

Método	Acurácia	Precisão	Recall	AUC
Transformada de Haar	0,7911	0,8014	0,7927	0,9150
Transformada de Fourier	0,7658	0,7960	0,7613	0,9127

Fonte: dados da pesquisa

Por fim, o ChatGPT-4 foi empregado para solucionar o mesmo problema de classificação, utilizando-se as imagens originalmente disponibilizadas na base de dados. As mesmas métricas de desempenho usadas para as CNNs foram consideradas para o ChatGPT-4. A solicitação feita ao ChatGPT-4 foi descrita como segue:

*"Olá chat, gostaria de pedir uma análise de imagens. Irei enviar um arquivo zip que contém um conjunto de 3 tipos de imagens: motion\_blur, defocused\_blur e sharp. Gostaria que você salvasse as classes de cada uma das imagens. Após isso, aplicasse a sua melhor técnica para definir de qual das classes é a imagem e, no final, comparasse os seus resultados com as classes reais de cada imagem."*

O arquivo compactado (com tamanho de 3 MB), contendo as imagens de teste, foi enviado ao ChatGPT-4, que executou um algoritmo para extrair as imagens. Em seguida, a ferramenta respondeu da seguinte forma:

*"As imagens foram extraídas com sucesso, e elas estão organizadas em subpastas que correspondem às classes defocused\_blurred, motion\_blurred e sharp. Agora, vou preparar os dados para a análise, identificando as classes reais de cada imagem e, em seguida, aplicarei uma técnica para prever a classe de cada imagem. Vou proceder com a análise."*

Inicialmente, o ChatGPT-4 tentou empregar técnicas de redes neurais convolucionais, optando por criar e executar um modelo baseado em VGG16. Contudo, a ferramenta retornou o seguinte erro: *"ModuleNotFoundError: No module named tensorflow."* Nesse contexto, o ChatGPT-4 demonstrou sua limitação em executar diretamente algoritmos que utilizam o TensorFlow, restringindo-se a gerar códigos para execução local pelo usuário. Reconhecendo a limitação, o ChatGPT-4 alterou a abordagem e aplicou o algoritmo de Laplace para resolver o problema. Na Tabela 2, são apresentados os resultados obtidos pelo ChatGPT-4 em comparação com os métodos utilizados anteriormente.

Ao comparar os resultados, observou-se que a solução fornecida pelo ChatGPT-4 apresentou um desempenho significativamente inferior ao das redes neurais convolucionais. Diante disso, foi solicitado ao ChatGPT-4 que tentasse melhorar sua abordagem. Procurou-se minimizar qualquer influência sobre o modelo durante a formulação da nova solicitação, uma vez que o objetivo era testar a capacidade da ferramenta em resolver problemas de maneira autônoma, sem a dependência de instruções de um usuário especialista. A única orientação fornecida foi que utilizasse uma abordagem

baseada em treinamento e teste, comumente aplicada em aprendizado de máquina. Assim, foi realizada a seguinte solicitação:

*"Em uma abordagem anterior, utilizei x base de treino e y de teste. para gerar um classificador de 3 tipos para as imagens: motion\_blurred, sharp e defocused\_blurred. Gostaria que você utilizasse a mesma base de teste para classificar entre as classes. Caso necessário, a base de treino pode ser utilizada para melhorar a implementação da abordagem implementada. Utilize as métricas acurácia, precisão, recall e AUC para avaliar o desempenho (com base no conjunto de teste) da abordagem implementada. Ambas as bases estão com as imagens padronizadas em 224 x 224 e com as 3 pastas das devidas classes. Irei enviar dois arquivos .zip."*

Os arquivos foram enviados conforme descrito (treino com 18 MB e teste com 3 MB). O ChatGPT-4 optou por gerar um código utilizando uma abordagem de aprendizado de máquina com a técnica SVM (*Support Vector Machine*). No entanto, ao tentar executar o código, a ferramenta não conseguiu fornecer os resultados devido a problemas de instabilidade na conexão, retornando a seguinte resposta:

*"Parece que o problema de perda de conexão persiste, mesmo após tentar reiniciar o processo. Isso pode estar relacionado a limitações temporárias no ambiente de execução ou outras questões técnicas. Como uma alternativa, você pode tentar executar o código localmente ou em outro ambiente de execução que suporte Python. Posso fornecer o código completo necessário para que você possa rodá-lo sem interrupções. Se você estiver de acordo, posso preparar e compartilhar o código com você. Como gostaria de proceder?"*

Diante da situação, aguardou-se o período recomendado pela ferramenta para verificar se o problema técnico seria resolvido. Após um dia, foi solicitado ao ChatGPT-4 que tentasse novamente executar o código gerado. Contudo, a ferramenta apresentou novamente a mesma limitação, sugerindo a execução local. Para dar continuidade à experimentação e avaliar o desempenho do ChatGPT-4 como gerador de código para um problema específico de aprendizado de máquina, solicitou-se à ferramenta que adaptasse o código para execução local.

O ChatGPT-4 forneceu o código adaptado, com orientações detalhadas sobre as bibliotecas necessárias, ajustes nas variáveis do código e instruções para execução local. O uso do ChatGPT-4 como gerador de código foi considerado em razão da elevada popularidade desse LLM, o que o torna atrativo para diversos usuários em diferentes cenários de aplicação. Seguindo as orientações fornecidas, o treinamento foi realizado com as bases de treino e teste. Após algumas horas, o modelo SVM foi testado.

Tabela 2 – Comparação de desempenho entre modelos com base nas transformadas de Haar e Fourier e o ChatGPT-4

Método	Acurácia	Precisão	Recall	AUC
ChatGPT-4 Laplace	0,4051	0,3975	0,4051	0,5205
ChatGPT-4 Local SVM	0,2122	0,2141	0,2122	0,6852

Fonte: dados da pesquisa

Os resultados do SVM, executado localmente, demonstraram que o algoritmo não conseguiu convergir adequadamente durante o treinamento, resultando em um desempenho abaixo do esperado no conjunto de teste.

#### 4.3 Avaliação com geração de dados

Foi decidido realizar testes e comparações entre redes neurais convolucionais baseadas nas transformações de Haar e Fourier e o ChatGPT-4. Para ampliar a base de treinamento, foram



realizados experimentos que incluíram a aplicação de filtros de desfoque em um novo conjunto de imagens, não pertencente ao conjunto de dados desfocados originalmente disponível no Kaggle. Foram selecionadas 1.000 novas imagens nítidas do conjunto de dados COCO, amplamente utilizado na área, e aplicados filtros de desfoque por movimento e de desfoque simples. Esse processo gerou um total de 3.000 novas imagens, elevando para 4.050 o total de imagens no conjunto de dados<sup>5</sup>.

Durante a geração dessas novas imagens, buscou-se manter a consistência dos efeitos de desfoque em relação ao conjunto de dados original do Kaggle. Essa uniformidade foi considerada crucial para evitar distorções nas métricas, especialmente devido ao aumento significativo do volume de dados, que quase quadruplicou.

A nova base de dados foi submetida aos mesmos procedimentos de pré-processamento adotados anteriormente. As imagens foram divididas aleatoriamente para treinamento (72,25%), validação (15%) e teste (12,75%), garantindo proporcionalidade e robustez no experimento.

O treinamento das redes neurais convolucionais foi realizado utilizando as transformações de Haar e Fourier, cada uma com 30 épocas para as etapas de treino e validação. Em seguida, os modelos foram testados com a nova base de teste, resultando nos dados apresentados na Tabela 4. Com o aumento do conjunto de dados, o desempenho foi aprimorado para todas as métricas avaliadas.

Tabela 4 – Comparação de desempenho entre os modelos treinados com base nas transformadas de Haar e Fourier com gerações de dados

Método	Acurácia	Precisão	Recall	AUC
Transformada de Haar	0,8783	0,8775	0,8770	0,9589
Transformada de Fourier	0,8799	0,8945	0,8801	0,9671

Fonte: dados da pesquisa

Após a obtenção dos resultados com as transformações Haar e Fourier, os mesmos procedimentos foram replicados utilizando o ChatGPT-4. Para isso, foi enviada ao modelo a seguinte solicitação:

*"Olá chat. gostaria de pedir uma análise de imagens. Irei enviar um arquivo zip que contém um conjunto de 3 tipos de imagens: motion\_blur, defocused\_blur e sharp. Gostaria que você salvasse as classes de cada uma das imagens. Após isso, aplicasse a sua melhor técnica para definir de qual das classes é a imagem e, no final, comparasse os seus resultados com as classes reais de cada imagem."*

Em seguida, foi enviado um arquivo compactado contendo as imagens de teste (agora com tamanho de 11 MB). O ChatGPT-4 processou o arquivo e respondeu:

*"Parece que o módulo necessário para o processamento de imagens com TensorFlow não está disponível neste ambiente. Vou tentar uma abordagem alternativa para analisar as imagens e classificar cada uma nas categorias fornecidas. Vou utilizar técnicas clássicas de processamento de imagem e machine learning para realizar a classificação. Vamos seguir em frente com essa abordagem."*

Isso evidenciou que o ChatGPT-4, inicialmente, não estava ciente da indisponibilidade do TensorFlow no ambiente utilizado. Reconhecendo essa limitação, o modelo adaptou sua abordagem e gerou um código baseado na extração de características por HOG (*Histogram of Oriented Gradients*), combinado com florestas aleatórias. Os resultados obtidos pelas redes neurais convolucionais e pelo ChatGPT-4 utilizando HOG e florestas aleatórias foram: 0,3934 de acurácia média, 0,3939 de precisão média, 0,3934 de recall médio e 0,5357 de AUC médio. O HOG extrai características baseadas nos

<sup>5</sup> Os scripts para geração estão disponíveis em [https://github.com/Kwentar/blur\\_dataset/tree/master/scripts](https://github.com/Kwentar/blur_dataset/tree/master/scripts)

gradientes locais de intensidade, enquanto as florestas aleatórias utilizam os conceitos de árvore e votação majoritária para realizar classificações.

Como se pode observar, enquanto as redes baseadas em transformadas apresentaram melhorias, o ChatGPT-4 entregou resultados inferiores em comparação à base de dados anterior. Esse resultado pode estar relacionado ao uso do algoritmo de HOG com florestas aleatórias, em vez do algoritmo de Laplace. Dessa forma, conclui-se que o ChatGPT-4, mesmo utilizando as mesmas instruções, pode optar por métodos distintos para resolver o problema.

No segundo passo do experimento, foi solicitada uma melhoria ao ChatGPT-4, enviando uma base de treino completa, a fim de viabilizar comparações mais justas com as redes que utilizaram transformadas. Para tanto, foi enviada a seguinte instrução ao ChatGPT-4:

*"Em uma abordagem anterior, utilizei x base de treino e y de teste. para gerar um classificador de 3 tipos para as imagens: motion\_blurred, sharp e defocused\_blurred. Gostaria que você utilizasse a mesma base de teste para classificar entre as classes. Caso necessário, a base de treino pode ser utilizada para melhorar a implementação da abordagem implementada. Utilize as métricas acurácia, precisão, recall e AUC para avaliar o desempenho (com base no conjunto de teste) da abordagem implementada."*

Em resposta, o ChatGPT-4 indicou:

*"Parece que a extração das características HOG está demorando mais do que o esperado. Infelizmente, mesmo com a intenção de não interromper o processo, ele ainda foi interrompido automaticamente após 60 segundos. A extração de características HOG para o conjunto completo de imagens continua sendo muito demorada neste ambiente. Como alternativa, sugiro que eu faça um pré-processamento para reduzir o tamanho das imagens ou o número de características extraídas, ou ainda que eu utilize outro método de classificação menos intensivo em termos de tempo de processamento. Isso deve permitir que o processo seja concluído dentro dos limites de tempo. Qual dessas opções você prefere?"*

Essa resposta evidenciou uma limitação do ChatGPT-4 para tarefas de aprendizado de máquina, devido ao limite de tempo de execução de 60 segundos por comando. Em casos que exijam algoritmos mais complexos ou bases de dados maiores, a execução no ambiente do ChatGPT-4 torna-se inviável.

Dando continuidade ao experimento, foi solicitado que o ChatGPT-4 escolhesse a abordagem mais adequada para otimizar o desempenho. O modelo optou por reduzir o tamanho das imagens para acelerar a extração de características, reduzindo-as de  $224 \times 224$  para  $64 \times 64$ . Ainda assim, o processo falhou novamente, com a seguinte resposta:

*"Mesmo com a redução do tamanho das imagens, o processo ainda está demorando mais do que o tempo limite de 60 segundos permite. Isso indica que, mesmo com essas modificações, o processamento ainda é intensivo. Para resolver isso, posso tentar: Reduzir ainda mais a resolução das imagens. Usar um classificador mais simples ou menos intensivo, como uma regressão logística ou uma SVM linear. Subamostrar o conjunto de dados para treinar e testar com um subconjunto menor. Qual dessas opções você prefere?"*

Em seguida, foi solicitado que o ChatGPT-4 escolhesse novamente a melhor opção para resolver o problema. Isso resultou na troca do método para um classificador mais simples, optando pela regressão logística. No entanto, ao executar o algoritmo de Regressão Logística, o ChatGPT-4 falhou novamente, pois o processo também excedeu o limite de 60 segundos. Diante disso, o ChatGPT-4 chegou à seguinte conclusão:

*"Mesmo utilizando um classificador mais simples, o processo ainda está excedendo o tempo limite de 60 segundos. Isso indica que a quantidade de dados ou a complexidade do problema*

*continua sendo um desafio significativo para o ambiente atual. Uma última abordagem possível seria subamostrar o conjunto de dados para reduzir a quantidade de dados usados no treinamento e teste. Isso deve permitir que o processo seja concluído dentro do tempo limite. Vou implementar essa abordagem e tentar novamente."*

Com isso, o ChatGPT-4 conseguiu executar o algoritmo de regressão logística. No entanto, ao analisar as linhas 3 e 4, percebe-se que o ChatGPT-4 reduziu em 90% o tamanho da base de dados, utilizando apenas 10% das imagens disponibilizadas. Essa redução significativa inviabilizou o experimento, pois limitou drasticamente a quantidade de dados disponíveis.

Diante das limitações do ChatGPT-4 para executar algoritmos de aprendizado de máquina e aprendizado profundo, decidiu-se continuar o experimento utilizando os algoritmos gerados pelo ChatGPT-4, mas executando-os localmente, conforme recomendado pelo próprio ChatGPT-4. Esse processo exige a configuração de um ambiente para comparar os algoritmos gerados pelo ChatGPT-4 com as redes baseadas em transformadas, mantendo o objetivo da pesquisa, que é comparar os resultados das redes com transformadas com os resultados das opções propostas pelo ChatGPT-4. Ainda é possível analisar o esforço, tempo e conhecimento necessários para ambas as abordagens.

Dando continuidade, foi decidido utilizar o algoritmo de redes neurais que o ChatGPT-4 recomendou durante a conversa: uma VGG-16 pré-treinada. A arquitetura VGG-16 é bastante relevante para a classificação de imagens em tarefas que envolvem CNNs (Depuru *et al.*, 2024; Putranto; Lin; Huang, 2024; Praveen; Isravel; Dhas, 2024). No início deste capítulo, foi mencionado que o ChatGPT-4 tentou utilizar uma VGG-16 para a tarefa solicitada, porém não consegue rodar algoritmos de aprendizado profundo, pois não possui o TensorFlow instalado em suas dependências.

Com a decisão de executar o código localmente, foi realizada a execução da VGG-16 sobre o conjunto de teste, obtendo-se os resultados iniciais em uma situação em que não seria possível construir uma base de dados para retreiná-la. Os resultados foram obtidos com base no algoritmo gerado pelo ChatGPT-4 para execução local. Como destacado por Tan e Le (2019), a transferência de aprendizado tem como objetivo aprimorar o desempenho dos modelos aplicados, aproveitando o conhecimento adquirido em tarefas distintas, porém relacionadas. Essa técnica facilita a criação de modelos robustos, especialmente em cenários de aprendizado profundo, onde a obtenção de dados categorizados é um desafio. Na Tabela 6 são apresentados os resultados obtidos por meio da execução local da sugestão apresentada pelo ChatGPT-4 somente com dados de teste, comparando com as transformadas de Haar e Fourier (pré-processamento para uma CNN).

Tabela 6 – Comparação de desempenho entre os modelos com base nas transformadas de Haar e Fourier e o ChatGPT-4 com gerações de dados, utilizando uma rede pré treinada e aprendizado por transferência

Método	Acurácia	Precisão	Recall	AUC
ChatGPT-4 VGG-16	0,3832	0,2919	0,3832	0,5127
ChatGPT-4 VGG-16 pré-treinada	0,7484	0,7570	0,7484	0,8123

Fonte: dados da pesquisa

Os resultados indicaram que o desempenho do modelo ChatGPT-4 foi significativamente inferior em comparação com as transformadas de Haar e Fourier. Esse desempenho inferior provavelmente ocorreu devido ao pré-processamento das imagens, que explicitou as principais características dos desfoques. Dessa forma, decidiu-se que o ChatGPT-4 deveria utilizar a base de treino previamente preparada, possibilitando uma comparação mais equitativa. Para isso, foi questionado ao ChatGPT-4 se o uso de uma base de treino poderia aprimorar o desempenho do código gerado. A pergunta foi formulada da seguinte maneira:

*"Caso eu tivesse uma base de treino para você utilizar para treinar a rede antes de utilizar direto nos testes. seria útil? Se sim. faça o código para que eu rode localmente."*

Então o ChatGPT-4 gerou a seguinte resposta:

"Sim. ter uma base de treino seria muito útil para treinar a rede neural antes de avaliar o desempenho no conjunto de teste. O uso da base de treino permitirá que o modelo VGG16, pré-treinado no ImageNet, seja ajustado ao seu conjunto de dados específico (aprendizado por transferência), o que geralmente melhora os resultados. Aqui está o código atualizado que utiliza tanto o conjunto de treino quanto o conjunto de teste para ajustar a rede e, em seguida, avaliar o modelo."

Foi indicado que o uso de uma base de treino seria benéfico para ajustar o modelo VGG16 pré-treinado ao conjunto de dados específico, por meio do aprendizado por transferência. Essa abordagem visa melhorar os resultados, aproveitando as camadas iniciais do modelo, que aprendem características genéricas, como bordas e formas, aplicáveis a diferentes problemas. Dessa forma, foi gerado um algoritmo com pequenas modificações, treinando a rede VGG-16 com a base de treino separada previamente.

Após o treinamento, foi observada uma melhora significativa no desempenho do modelo ChatGPT-4. Contudo, as transformadas continuaram a apresentar resultados superiores na tarefa proposta. Em relação ao tempo de treinamento, execução e custo de processamento, as três redes tiveram desempenhos similares, com tempos médios de cerca de 2 horas para treinamento. Entretanto, o tempo necessário para desenvolver soluções baseadas em transformadas foi significativamente maior, exigindo leitura de artigos científicos e experimentações, enquanto as soluções com ChatGPT-4 foram alcançadas em poucas interações.

As sugestões fornecidas pelo ChatGPT-4 podem reduzir o tempo necessário para selecionar modelos, mas ainda demandam a configuração de ambientes e a realização de treinamentos e testes localmente. Apesar disso, o desempenho das transformadas continuou superior.

Considerando os bons resultados obtidos com a rede VGG16 recomendada pelo ChatGPT-4, decidiu-se realizar novas comparações utilizando uma terceira base de dados, contendo o dobro de imagens da base anterior. Os mesmos procedimentos foram seguidos, incluindo o pré-processamento e a separação de dados para teste e treino, assegurando a utilização das mesmas imagens em todas as redes.

Os resultados do treinamento e testes com a terceira base estão apresentados na Tabela 7.

Tabela 7 – Comparação de desempenho entre os modelos com base nas transformadas de Haar e Fourier e o ChatGPT-4 com gerações de dados dupla, utilizando uma rede pré treinada e aprendizado por transferência

Método	Acurácia	Precisão	Recall	AUC
Transformada de Haar	0,8889	0,8913	0,8886	0,9503
Transformada de Fourier	0,8403	0,8827	0,8418	0,9504
ChatGPT-4 VGG-16 pré-treinada	0,8198	0,8175	0,8198	0,8649

Fonte: dados da pesquisa

Com o aumento do tamanho da base de dados, o desempenho do modelo ChatGPT-4 continuou a melhorar em termos de acurácia. No entanto, os modelos baseados em transformadas mantiveram desempenho superior, sugerindo que já haviam atingido um limite no ganho de desempenho. Na literatura, a transformada de Fourier foi utilizada no pré-processamento de outro conjunto de dados e outras classes, com o modelo validado por meio do método de validação *hold-out* (Tiwari, 2020). Os resultados apresentados por Tiwari (2020) evidenciaram uma acurácia média de 0,93 para um modelo *multilayer perceptron* e uma acurácia média de 0,97 para um modelo CNN. O autor utilizou um conjunto de dados com imagens de gestos de mão. No entanto, não foram encontrados trabalhos que utilizassem a transformada de Haar no pré-processamento de dados para análise de imagens com desfoque. Observa-se que, embora o resultado de acurácia média obtido neste estudo tenha sido um



pouco inferior aos resultados reportados por Tiwari (2020), foi utilizado um conjunto de dados consideravelmente mais variado. Portanto, os resultados apresentados estão respaldados pela literatura.

Como descrito anteriormente, o pré-processamento das imagens com as transformadas destaca as principais características dos desfoques, o que pode justificar o desempenho superior dos modelos. Contudo, a etapa de pré-processamento pode ser custosa, dependendo do tamanho do conjunto de dados. Além disso, a discrepância de desempenho inicial do ChatGPT-4 pode ser explicada pela tentativa de classificação sem o uso de aprendizado de máquina e pela tentativa de redução no tamanho do conjunto de dados de treinamento e teste, como forma de viabilizar a implementação sem a execução local pelo usuário.

## 5 Considerações finais

Neste trabalho, foi realizada uma análise comparativa entre redes neurais convolucionais e o ChatGPT-4, com o objetivo de classificar imagens borradas. O estudo investigou e comparou o desempenho de ambas as abordagens em termos de métricas como acurácia, custo computacional e tempo de processamento, buscando identificar o método mais eficaz para o tratamento de imagens com diferentes tipos de desfoque.

Durante a análise, as redes neurais convolucionais, reconhecidas por sua relevância no reconhecimento automatizado de padrões visuais, apresentaram um desempenho satisfatório com o conjunto de dados utilizado. O uso das transformadas de Haar e Fourier na etapa de pré-processamento das imagens contribuiu significativamente para a melhoria dos resultados de classificação. Os experimentos realizados reforçaram que as redes neurais convolucionais continuam sendo, em geral, a melhor escolha para problemas complexos de visão computacional, incluindo a análise de imagens borradas por movimento e desfoque de foco.

Por outro lado, o ChatGPT-4, embora amplamente conhecido como um modelo de linguagem de grande porte, demonstrou limitações ao lidar diretamente com grandes volumes de dados em tarefas de classificação de imagens. A necessidade de adaptar e executar localmente os códigos gerados pelo modelo evidenciou que ele ainda não está totalmente preparado para lidar, de forma autônoma, com problemas de aprendizado profundo envolvendo grandes volumes de dados. Contudo, é importante destacar o potencial do ChatGPT-4 como uma ferramenta de suporte, especialmente na geração de código e na sugestão de soluções rápidas, características que podem ser valiosas em cenários com restrições de tempo no desenvolvimento.

O ChatGPT-4 mostrou-se útil na geração de códigos e na sugestão de abordagens que podem ser rapidamente testadas e ajustadas por desenvolvedores. Essa funcionalidade é particularmente relevante em contextos onde a agilidade é crucial, oferecendo soluções com uma curva de aprendizado mais acessível, facilitando o trabalho de desenvolvedores menos experientes. Entretanto, deve-se considerar o custo financeiro associado ao uso do ChatGPT-4, especialmente na versão paga, como foi utilizado neste trabalho. Em cenários de produção, pode ser necessária a utilização da API da OpenAI, com custos variáveis por requisição, dependendo da versão do modelo escolhido (GPT-4 ou GPT-3.5 Turbo).

Em relação aos trabalhos futuros, recomenda-se uma análise mais aprofundada dos algoritmos utilizados neste estudo, a fim de explicar as diferenças observadas nos desempenhos das redes neurais e do ChatGPT-4. Além disso, seria interessante realizar experimentos envolvendo um grupo de controle e tratamento, com desenvolvedores, cientistas de dados e especialistas em aprendizado de máquina de diferentes níveis de experiência, para avaliar o tempo e o esforço necessários para implementar as soluções propostas. Ampliar o conjunto de dados, incorporando a geração de imagens com novos desfoques, também pode contribuir para resultados mais robustos e permitir comparações mais detalhadas entre as abordagens. Além disso, planeja-se experimentar outras arquiteturas de redes neurais.

Conclui-se que, embora as redes neurais convolucionais com o uso de transformadas ainda se mostrem a opção mais eficaz para a classificação de imagens borradas, o ChatGPT-4 apresenta um potencial promissor para acelerar o desenvolvimento de soluções em aprendizado de máquina. Esse potencial é particularmente evidente quando o modelo é utilizado em conjunto com arquiteturas de

redes previamente treinadas. Assim, o ChatGPT-4 poderia ser utilizado como ferramenta de suporte para a geração de código para a tarefa de classificação.

### **Financiamento (obrigatória; não deve ser numerada)**

Os autores gostariam de agradecer o Fundo Nacional do Desenvolvimento da Educação, TED 10320, pelo auxílio oferecido a essa pesquisa.

### **Conflito de interesses**

Os autores declaram não haver conflito de interesses.

### **Contribuições ao artigo**

**SANTOS, A. L. A.; ROSA, D. C.:** concepção ou desenho do estudo/pesquisa; análise e/ou interpretação dos dados; implementação e validação de modelos de aprendizado de máquina. **CÉSAR SOBRINHO, A. A. C.; CORDEIRO, T. D.; MATOS, D. D. M. C.:** revisão final com participação crítica e intelectual no manuscrito. Todos os autores participaram da escrita, discussão, leitura e aprovação da versão final do artigo.

### **Referências**

ABDALJALEEL, M.; BARAKAT, M.; ALSANAFI M.; SALIM, N. A.; ABAZID, H.; MALAEB, D.; MOHAMMED, A. H.; HASSAN, B. A. R.; WAYYES, A. M.; FARHAN, S. S.; KHATIB, S. E.; RAHAL, M.; SAHBAN, A.; ABDELAZIZ, D. H.; MANSOUR, N. O.; ALZAYER, R.; KHALIL, R.; FEKIH-ROMDHANE, F.; HALLIT, R.; HALLIT, S.; SALLAM, M.. A multinational study on the factors influencing university students' attitudes and usage of ChatGPT. **Scientific Reports**, v. 14, 1983, 2024. DOI: <https://doi.org/10.1038/s41598-024-52549-8>.

ADESHOLA, I.; ADEPOJU, A. P. The opportunities and challenges of ChatGPT in education. **Interactive Learning Environments**, 2023. DOI: <https://doi.org/10.1080/10494820.2023.2253858>.

ALGHAMDI, R. Exploring the impact of ChatGPT-generated feedback on technical writing skills of computing students: a blinded study. **Education and Information Technologies**, v. 29, p. 18901-18926, 2024. DOI: <https://doi.org/10.1007/s10639-024-12594-2>.

BAHRINI, A.; KHAMOSHIFAR, M.; ABBASIMEHR, H.; RIGGS, R. J.; ESMAEILI, M.; MAJDABADKOHNE, R. M.; PASEHVAR, M. ChatGPT: applications, opportunities, and threats. In: SYSTEMS AND INFORMATION ENGINEERING DESIGN SYMPOSIUM (SIEDS), 2023, Charlottesville. **Proceedings [...]**. Charlottesville: IEEE, 2023. p. 274-279. DOI: <https://doi.org/10.1109/SIEDS58326.2023.10137850>.

BROWN, T. B.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A.; AGARWAL, S.; HERBERT-VOSS, A.; KRUEGER, G.; HENIGHAN, T.; CHILD, R.; RAMESH, A.; ZIEGLER, D. M.; WU, J.; WINTER, C.; HESSE, C.; CHEN, M.; SIGLER, E.; LITWIN, M.; GRAY, S.; CHESS, B.; CLARK, J.; BERNER, C.; MCCANDLISH, S.; RADFORD, A.; SUTSKEVER, I.; AMODEI, D. Language models are few-shot learners. In: INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 34., 2020, Vancouver. **Proceedings [...]**. Vancouver: ACM, 2020, p. 1877-1901. Disponível em: <https://dl.acm.org/doi/abs/10.5555/3495724.3495883>. Acesso em: 24 nov. 2024.

CHEN, T.-J. A novel image blurring detection scheme using spatial autocorrelation. In: IRISH SIGNALS AND SYSTEMS CONFERENCE (ISSC), 34., 2023, Dublin. **Proceedings [...]**. Dublin: IEEE, 2023. DOI: <https://doi.org/10.1109/ISSC59246.2023.10162072>.

DEPURU, S.; AMALA, K.; SUPRIYA, P.; REDDY, A. B.; GIREESH R. S. VGG-16 technique to reduce the global food crises and enhance the crop yields: deep learning approaches. *In: INTERNATIONAL CONFERENCE ON APPLIED ARTIFICIAL INTELLIGENCE AND COMPUTING (ICAAIC)*, 3., 2024, Salem. **Proceedings [...]**. Salem: IEEE, 2024. p. 596-599. DOI: <https://doi.org/10.1109/ICAAIC60222.2024.10575562>.

LEE, S.; KANG, Q.; MADIREDDY, S.; BALAPRAKASH, P.; AGRAWAL, A.; CHOUDHARY, A.; ARCHIBALD, R.; LIAO, W. Improving scalability of parallel CNN training by adjusting mini-batch size at run-time. *In: IEEE INTERNATIONAL CONFERENCE ON BIG DATA (BIG DATA)*, 2019, Los Angeles. **Proceedings [...]**. Los Angeles: IEEE, 2019. p. 830-839. DOI: <https://doi.org/10.1109/BigData47090.2019.9006550>.

LI, J.; DADA, A.; PULADI, B.; KLEESIEK, J.; EGGER, J. ChatGPT in healthcare: a taxonomy and systematic review. **Computer Methods and Programs in Biomedicine**, v. 245, 108013, 2024. DOI: <http://dx.doi.org/10.1016/j.cmpb.2024.108013>.

LI, M.; MA, Z.; WANG, Y. G.; ZHUANG, X. Fast Haar transforms for graph neural networks. **Neural Networks**, v. 128, n. 4, p. 188-198. 2020. DOI: <http://dx.doi.org/10.1016/j.neunet.2020.04.028>.

NAZ, I.; ROBERTSON, R. Exploring the feasibility and efficacy of ChatGPT3 for personalized feedback in teaching. **Electronic Journal of e-Learning**, v. 22, n. 2, p. 98-111, 2024. DOI: <https://doi.org/10.34190/ejel.22.2.3345>.

PRAVEEN, B.; ISRAVEL, D. P.; DHAS, J. P. M. VGG-16 based deep learning approach for plant species detection. *In: INTERNATIONAL CONFERENCE ON ADVANCES IN MODERN AGE TECHNOLOGIES FOR HEALTH AND ENGINEERING SCIENCE (AMATHE)*, 2024, Shivamogga. **Proceedings [...]**. Shivamogga: IEEE, 2024. p. 1-6. DOI: <http://dx.doi.org/10.1109/AMATHE61652.2024.10582179>.

PUTRANTO, A.; LIN, T.-H.; HUANG, B.-X. Deep learning-integrated electromagnetic imaging for evaluating reinforced concrete structures in water-contact scenarios. **Automation in Construction**, v. 164, 105459, 2024. DOI: <https://doi.org/10.1016/j.autcon.2024.105459>.

RAO, K. R.; KIM, D. N.; HWANG, J. J. **Fast Fourier transform**: algorithms and applications. Dordrecht: Springer, 2010. DOI: <https://doi.org/10.1007/978-1-4020-6629-0>.

SILVA, L. C.; CESAR SOBRINHO, Á. A. C.; CORDEIRO, T. D.; MELO, R. F.; BITTENCOURT, I. I.; MARQUES, L. B.; MATOS, D. D. M. C.; SILVA, A. P.; ISOTANI, S. Applications of convolutional neural networks in education: a systematic literature review. **Expert Systems with Applications**, v. 231, 120621, 2023. DOI: <https://doi.org/10.1016/j.eswa.2023.120621>.

SINGH, J.; ZOUHAR, V.; SACHAN, M. Enhancing textbooks with visuals from the web for improved learning. *In: CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING (EMNLP 2023)*, 2023, Singapore. **Proceedings [...]**. Singapore: ACL, 2023. p. 11931-11944. DOI: <https://doi.org/10.3929/ethz-b-000653494>.

SOBAIH, A. E. E.; ELSHAER, I. A.; HASANEIN, A. M. Examining students' acceptance and use of ChatGPT in Saudi Arabian higher education. **European Journal of Investigation in Health, Psychology and Education**, v. 14, n. 3, p. 709-721, 2024. DOI: <http://dx.doi.org/10.3390/ejihpe14030047>.

SOBRINHO, Á.; BITTENCOURT, I. I.; SILVEIRA, A. C. M.; SILVA, A. P.; DERMEVAL, D.; MARQUES, L. B.; RODRIGUES, N. C. I.; SOUZA, A. C. S.; FERREIRA, R.; ISOTANI, S. Towards digital transformation of the validation and triage process of textbooks in the Brazilian educational policy. *Sustainability*, v. 15, n. 7, 5861, 2023. DOI: <https://doi.org/10.3390/su15075861>.

TAN, M.; LE, Q. EfficientNet: rethinking model scaling for convolutional neural networks. *In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING (ICML)*, 3., 2019, Long Beach. **Proceedings [...]**. Long Beach: ICML, 2019. p. 6105-6114. Disponível em: <http://proceedings.mlr.press/v97/tan19a.html>. Acesso em: 24 nov. 2024.

TIWARI, S. A blur classification approach using deep convolution neural network. *International Journal of Information System Modeling and Design (IJISMD)*, v. 11, n. 1, p. 93-111, 2020. DOI: <https://doi.org/10.4018/IJISMD.2020010106>.

VASWANI, A.; SHAZEER, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSHUKHIN, I. Attention is all you need. *In: CONFERENCE ON NEURAL PROCESSING SYSTEMS (NIPS 2017)*, 31., Long Beach. **Proceedings [...]**. Long Beach, 2017. Disponível em: [https://papers.nips.cc/paper\\_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html](https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html). Acesso em: 24 nov. 20204.

WANG, R.; LI, W.; ZHANG, L. Blur image identification with ensemble convolution neural networks. *Signal Processing*, v. 155, p. 73-82, 2019. DOI: <https://doi.org/10.1016/j.sigpro.2018.09.027>.

WANG, R.; LI, W.; LI, R.; ZHANG, L. Automatic blur-type classification via ensemble SVM. *Signal Processing: Image Communication*, v. 71, p. 24-35, 2019. DOI: <https://doi.org/10.1016/j.image.2018.08.003>.

XU, X.; THIEN, L. M. Unleashing the power of perceived enjoyment: exploring Chinese undergraduate EFL learners' intention to use ChatGPT for English learning. *Journal of Applied Research in Higher Education*, Ahead of Print, 2024. DOI: <https://doi.org/10.1108/JARHE-12-2023-0555>.