

Uso de visão computacional na análise de testes cognitivos

1st Paulo Brasil
Universidade de Pernambuco
Recife, Brazil
orcid.org/0000-0001-7170-5196

2nd Lucas Matheus Rodrigues dos Santos
Universidade de Pernambuco
Recife, Brazil
orcid.org/0000-0003-1183-1216

3rd Elisson Lima Gomes da Silva
Universidade de Pernambuco
Recife, Brazil
orcid.org/0000-0001-6998-7809

4th Rodrigo Melo
Universidade Federal de Pernambuco
Recife, Brasil
orcid.org/0000-0002-9120-6156

5th Rodrigo de Paula Monteiro
Universidade Católica de Pernambuco
Recife, Brasil
orcid.org/0000-0002-2423-5088

6th Carmelo J. A. Bastos-Filho
Universidade de Pernambuco
Recife, Brasil
orcid.org/0000-0002-0924-5341

7th Sergio Campello Oliveira
Universidade de Pernambuco
Recife, Brasil
orcid.org/0000-0003-1058-1139

Resumo—O objetivo deste trabalho é apresentar um estudo da aplicação de diversas arquiteturas de redes neurais convolucionais para automação do teste do desenho do pentágono aplicado no processo de avaliação cognitiva que busca identificar a presença de um possível declínio cognitivo. O teste que esse trabalho busca automatizar é aplicado no Mini Exame do Estado Mental (MEEM), que é, atualmente, a avaliação cognitiva mais utilizada para identificar sintomas de demência no mundo. Uma de suas várias etapas consiste em copiar o desenho de dois pentágonos sobrepostos, onde a interseção desses polígonos forma um losango. Esse desenho feito pelo paciente será a entrada do modelo convolucional que deverá classificá-la como correto ou incorreto de acordo com os critérios estabelecidos no MEEM. Nos experimentos realizados neste trabalho foi utilizada uma base de dados desenvolvida através da coleta de amostras de desenhos publicados em artigos da literatura, assim como elaboradas por voluntários que simularam esta etapa do MEEM e forneceram um bom número de imagens que foram utilizadas para treinamento dos modelos. As melhores arquiteturas testadas obtiveram acurácia média de 70% no conjunto de teste.

Palavras-Chave—Redes Neurais Convolucionais; Mini Exame do Estado Mental; Teste do desenho do pentágono; Doença de Alzheimer; Aprendizado profundo; visão computacional.

I. INTRODUÇÃO

Com o aumento da expectativa de vida, pessoas em todo o mundo estão vivendo mais. Assim, problemas relacionados à idade, como perda auditiva, osteoartrite e demência, estão se tornando cada vez mais frequentes [1]. Dentre eles, a demência é considerada uma das causas significativas de incapacidade e dependência na população idosa de acordo com a Organização Mundial da Saúde (OMS) [2]. O diagnóstico precoce é fundamental, pois pode ajudar a prevenir algumas complicações dessa doença e melhorar a qualidade de vida dos pacientes. Assim, métodos que possam ser amplamente utilizados para a realização do diagnóstico são imprescindíveis. Alguns desses

métodos são a realização de testes cognitivos que são usados para se obter uma medida breve e objetiva da função cognitiva dos pacientes [3], [4]. A capacidade de copiar desenhos é chamada de Práxis construtiva. Sua avaliação é incluída em alguns testes cognitivos, pois sua disfunção, denominada apraxia construtiva, pode indicar a presença de doenças que causam demência, como a doença de Alzheimer [5]. Existe um número substancial de testes cognitivos, mas o Mini Exame do Estado Mental (MEEM) é o exame mais utilizado para essa finalidade e já está disponível em vários idiomas ao redor do mundo [6], [7].

O MEEM possui várias fases com o objetivo de avaliar diversos aspectos cognitivos. Se concentrando na análise de habilidades linguísticas, habilidades matemáticas, orientação e memória. Uma das fases é copiar um desenho de pentágonos que se cruzam, conforme mostrado na Figura 1, que avalia se o paciente tem apraxia, distúrbio motor ou não [8]. O resultado do MEEM pode ajudar a indicar a presença de declínio cognitivo ou da demência [7].

Um dos problemas atuais relacionados aos sistemas de saúde é a falta de cobertura das regiões empobrecidas. Que são ainda mais relevante nos países em desenvolvimento, onde as pessoas fora das regiões metropolitanas não têm acesso a exames para detectar doenças que diminuem a qualidade de vida ou a expectativa de vida [9]. Uma possibilidade de contornar essa limitação é automatizar algumas fases dos exames e encaminhar essas informações a um centro de saúde especializado para análise dos resultados por meio de serviços de computação em nuvem.

O MEEM possui diferentes formas de interação com o paciente, sendo uma delas através de um desenho, o qual o paciente deve copiar. Nossa hipótese é de que seria possível automatizar essa etapa através do uso de algoritmos baseados

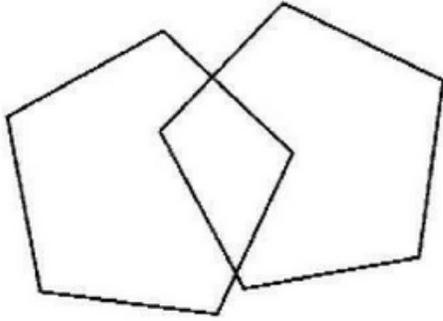


Figura 1. Desenho da interseção dos pentágonos usado durante o MEEM.

em inteligência artificial para identificação de padrões nos desenhos. Porém, não se trata de um problema simples visto a grande variabilidade de desenhos possíveis, tanto para os casos positivos quanto para os negativos. Assim, é necessário desenvolver uma metodologia de avaliação de desenhos do teste da cópia do desenho dos pentágonos que se cruzam para automatizar o MEEM.

As Redes Neurais Artificiais (RNAs) são compostas por neurônios agrupados em camadas. Cada elemento de uma camada da rede recebe sinais de entrada, que são ponderados e passados por uma função de ativação não linear, e então são passados para as próximas camadas. Existem vários pesos e polarizações entre cada uma dessas camadas, que são precisamente responsáveis por aprender os padrões entre as entradas. Eles são multiplicados pelos sinais passados e assim continuam até o final da rede [10]. A retropropagação atualiza os pesos relativos a cada camada, e assim se aproxima da saída desejada, realizando o aprendizado. Um dos problemas das RNAs é o treinamento, pois é um desafio reunir dados suficientes para treiná-las. No entanto, sua aplicabilidade é vasta, variando do reconhecimento de padrões, classificações e até mesmo previsões de valores futuros [10]. As Redes Neurais Convolucionais, do inglês *Convolutional Neural Network* (CNNs) é uma arquitetura que utiliza diferentes tipos de estratégias para o aprendizado de padrões (campos receptivos, pesos compartilhados, subamostragem espacial) [11]. Essas redes são compostas, normalmente, por: camadas convolucionais, ou seja, que utilizam filtros convolucionais, camadas de *pooling* que servem para reduzir a dimensionalidade, em outras palavras simplificar a informação, e, por fim, camadas densas, que são camadas totalmente conectadas.

Diferentemente de uma rede neural artificial rasa, as CNNs tentam aprender com pequenas partes e características da imagem usando filtros. Além disso, essas redes têm apresentado resultados promissores em tarefas relacionadas à detecção de padrões em desenhos [12], [13]. Zhang et al. [13] propôs o *SketchNet* para classificação de rascunhos. Essa abordagem empregou imagens reais positivas e negativas com esboços durante o treinamento do modelo. O objetivo era descobrir estruturas visuais coerentes entre os esboços e seus pares

positivos. Por outro lado, o *DeepSketch*, proposto por Seddati et al. [12], usou apenas esboços para treinar os modelos, que eram redes neurais convolucionais. Eles também mostraram que os recursos aprendidos por esses modelos são úteis para pesquisa por similaridade.

Assim, é proposto neste artigo o uso de modelos baseados em CNN para avaliar os desenhos presentes no teste da cópia do desenho de pentágonos que se cruzam. Para tal, foi desenvolvido um conjunto de dados para que fosse possível treinar e testar a rede. Também é apresentada uma análise para encontrar uma arquitetura CNN adequada. O restante deste artigo foi organizado da seguinte maneira. A Seção 2 apresenta mais informações sobre os desenhos de pentágonos que se cruzam e alguns detalhes sobre o conjunto de dados usado. O algoritmo e os modelos usados são apresentados na Seção 3. Os resultados obtidos são discutidos na Seção 4. Finalmente, a conclusão e os trabalhos futuros são apresentados na Seção 5.

II. O PROBLEMA DO DESENHO DOS PENTÁGONOS QUE SE CRUZAM

A última fase do MEEM é a cópia do desenho de pentágonos que se cruzam. O examinador mostra uma figura, como a da Figura 1, e pede ao paciente que a copie exatamente como está. O desenho é então analisado para observar se todos os lados dos pentágonos são preservados, e a interseção entre eles forma um losango. Nesse caso, é atribuído um ponto ao resultado do MEEM. Por outro lado, se as condições não forem satisfeitas, nenhum ponto será dado. O examinador deve ignorar tremores e rotações [14].

É preciso observar que a posição e os tamanhos dos pentágonos podem variar, dificultando o uso de heurísticas pré-determinadas para resolver o problema. Além disso, algumas linhas podem ser irregulares e a conexão entre as linhas pode não ser perfeita. Após o procedimento de pré-processamento, as linhas também podem apresentar pontos faltantes, o que pode atenuar o desempenho dos algoritmos de processamento de imagens.

III. MATERIAIS E MÉTODOS

Nosso objetivo é avaliar a implantação da CNN para resolver o problema da classificação da cópia do desenho dos pentágonos. Assim, foi construído um conjunto de dados usando desenhos encontrados, principalmente, na literatura, usando a classificação dos autores na maioria dos casos como rótulos dos casos positivos e negativos, e nós contribuimos com mais alguns desenhos feitos à mão. Nos casos em que os desenhos não tinham uma classificação clara, duas terapeutas ocupacionais, que trabalham com a aplicação do MEEM, foram consultadas para se averiguar a classificação correta do desenho.

Deste modo, foram coletados 401 desenhos, sendo 201 da classe correta, ou seja, que satisfazem as condições estipuladas pelo MEEM - alguns exemplos são mostrados na Figura 2 - e 200 classificados como incorretos de acordo com os padrões exigidos pelo teste, Figura 3 mostra alguns exemplos.

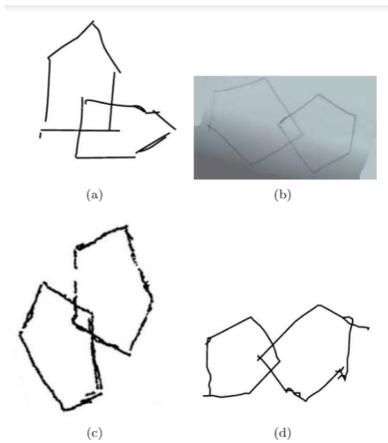


Figura 2. Imagens corretas da base de dados.

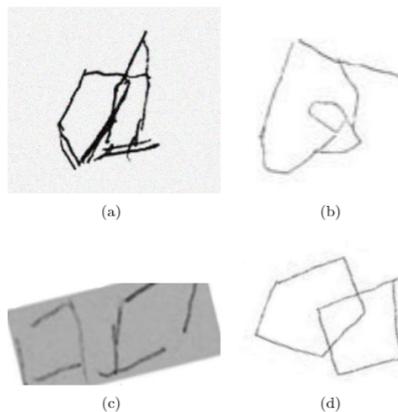


Figura 3. Imagens incorretas da base de dados.

Devido a baixa quantidade de dados levantados, o treinamento da CNN ficaria comprometido. Portanto, técnicas de aumento de dados, do inglês *data augmentation*, foram utilizadas para expandir os exemplos do conjunto de treino. Para se verificar a hipótese que o conjunto de dados era pequeno, se testou os modelos escolhidos várias vezes. Deste modo, variamos a quantidade de imagens escolhidas para o conjunto de treino para apurar o impacto que ocorreria na acurácia e perda das redes.

Foram utilizadas um total de três técnicas para se gerar as imagens que ampliariam o conjunto de dados para treino. A primeira se trata de um simples inversão horizontal da imagem, em outras palavras, pegando a imagem espelhada em relação a sua vertical. Já a segunda técnica utilizada foi a adição de um ruído aleatório. Para tal, foi usado o *random_noise* da biblioteca *skimage* [15], que pode adicionar diferentes tipos de ruídos na imagem gerada. Por fim, a última técnica utilizada foi a rotação das imagens originais. Para esse fim, os desenhos selecionados eram rotacionados de forma aleatória entre 15% para esquerda e 15% para direita da imagem original. A ferramenta utilizada para aplicar as rotações foi a função

transform.rotate, também, da biblioteca *skimage* [16]. Um detalhe que mais de uma técnica poderia ser utilizada em cada imagem gerada, a determinação da quantidade de técnicas que seriam usadas em cada desenho foi feita de forma aleatória também.

Como os desenhos apresentam linhas finas e uma relação complicada entre essas duas figuras geométricas, que podem apresentar distorções. Projetar uma arquitetura CNN, e definir os hiperparâmetros a serem utilizados, se torna uma atividade desafiadora.

Inicialmente se realizou um estudo preliminar para definir certas características, como número de camadas da rede CNN a ser utilizada. Assim, diferentes arquiteturas foram testadas, na base da tentativa e erro, para se determinar quais combinações de valores de hiperparâmetros forneciam os melhores resultados. Primeiro, se testou a quantidade de camadas e características dos filtros. Depois, se testou as funções de otimização (Adam, RMSprop, SGD) e de perda (MSE, Cross entropy).

Após esses testes iniciais, decidimos explorar melhor as características dos filtros nas camadas convolucionais, pois já se sabia a quantidade de camadas mínimas e máximas. Por exemplo, quando se utilizou CNNs com mais de cinco camadas convolucionais e duas camadas densas, o modelo começava a sofrer com sobreajuste e perder eficácia. Por outro lado, quando utilizávamos menos de três camadas convolucionais, os resultados obtidos não eram satisfatórios. Assim, se testou, dentro desse intervalo de camadas, a quantidade de filtros que seriam necessários nas camadas convolucionais. Além disso, também testamos a quantidade de neurônios nas camadas densas.

Todas as arquiteturas testadas foram treinadas com o mesmo conjunto de dados. Após esses testes iniciais, os três modelos, que apresentaram as melhores respostas, foram escolhidos para análise mais profunda. A Tabela I apresenta a arquitetura desses modelos. Com o objetivo de verificar a influência do tamanho do conjunto de dados de treinamento e averiguar se o conjunto de dados seria suficiente ou se seria necessário coletar mais imagens. Visto que a quantidade de casos positivos e negativos do desenho são muito amplos e que os resultados de classificação obtidos foram abaixo do que se era esperado. Assim, foram feitos testes com três tamanhos diferentes do conjunto de treinamento. Os experimentos utilizaram as configurações do conjunto de dados mostradas na Tabela II.

O conjunto de treinamento foi separado de forma aleatória do conjunto de dados totais. A partir desse ponto foram geradas o triplo de imagens com as técnicas de *data augmentation*. Desse conjunto de treino maior se extraía as imagens que seriam usadas no processo de validação do modelo. Já o conjunto de teste se limitava a 101 imagens, selecionadas de forma aleatória, do que sobrava do conjunto de dados originais. Após a separação do conjunto de dados, os três modelos eram treinados, validados e testados com o mesmo conjunto de dados, determinando assim a primeira rodada de treinamento. Para a segunda rodada, uma nova divisão da base de dados era elaborada de forma aleatória. Esse processo se repetiu

Tabela I
ARQUITETURAS E PARÂMETROS DE TREINAMENTO DAS CNNs QUE OBTIVERAM OS MELHORES RESULTADOS.

		Modelo 1.	Modelo 2.	Modelo 3.
	Treino	<i>Batch Size=16, 30 Épocas</i>		
Índice	Camada	Detalhe		
1	Convolutacional	16 filtros 3x3; <i>stride</i> 1	16 filtros 3x3; <i>stride</i> 1	16 filtros 3x3; <i>stride</i> 1
2	<i>Max pooling</i>	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2
3	Convolutacional	16 filtros 3x3; <i>stride</i> 1	16 filtros 3x3; <i>stride</i> 1	16 filtros 3x3; <i>stride</i> 1
4	<i>Max pooling</i>	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2
5	Convolutacional	32 filtros 3x3; <i>stride</i> 1	16 filtros 3x3; <i>stride</i> 1	32 filtros 3x3; <i>stride</i> 1
6	<i>Max pooling</i>	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2
7	Convolutacional	64 filtros 3x3; <i>stride</i> 1	32 filtros 3x3; <i>stride</i> 1	64 filtros 3x3; <i>stride</i> 1
8	<i>Max pooling</i>	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2
9	Convolutacional	128 filtros 3x3; <i>stride</i> 1	32 filtros 3x3; <i>stride</i> 1	64 filtros 3x3; <i>stride</i> 1
10	<i>Max pooling</i>	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2	2x2, <i>stride</i> 2
11	Densa	50 Neurônios	100 Neurônios	50 Neurônios
12	Densa	10 Neurônios	20 Neurônios	50 Neurônios
13	Saída	02 Neurônios	02 Neurônios	02 Neurônios

Tabela II
CONFIGURAÇÕES DA DIVISÃO DO CONJUNTO DE DADOS UTILIZADAS NOS EXPERIMENTOS.

	Experimento 1	Experimento 2	Experimento 3
Treinamento	1200 (300 reais + 900 geradas)	800 (200 reais + 600 geradas)	400 (100 reais + 300 geradas)
Validação	10% do treinamento	10% do treinamento	10% do treinamento
Teste	101	101	101

por 30 vezes. A divisão da base se dava através da função *train_test_split* do *sklearn* [17].

Finalmente, os modelos de redes escolhidos usaram a função de otimização Adam, para a função de perda foi utilizado a entropia cruzada. Nas camadas densas foi empregue a função de ativação unidade linear retificada, do inglês *rectified linear unit* (ReLU).

IV. RESULTADO

Nessa seção será discutido os resultados dos testes obtidos com as três arquiteturas selecionadas, Tabela I. Foram analisados os valores de acurácia e de perda que esses modelos obtiveram ao serem apresentadas ao conjunto de testes.

As Figuras 4 e 5 mostram os resultados obtidos no primeiro experimento, como demonstrado na Tabela II e explicado na Seção III. Os modelos obtiveram desempenhos semelhantes, em relação a acurácia Figura 4, quando treinados com esse conjunto de dados. Porém o resultado alcançado foi baixo, ficando em torno dos 70%. Foi devido a essa resposta baixa que decidimos avaliar o tamanho do conjunto de dados levantado. O Modelo 2 obteve valores de acurácia com menor variabilidade entre as simulações, enquanto o Modelo 1 teve a maior. Em relação a função de perda, os modelos apresentaram valores um pouco diferente. Sendo o Modelo 1 o que obteve a menor distribuição. Quando analisamos a função de perda,

desejamos que ela tenha valores baixos, pois isso representaria uma melhor separação das classes.

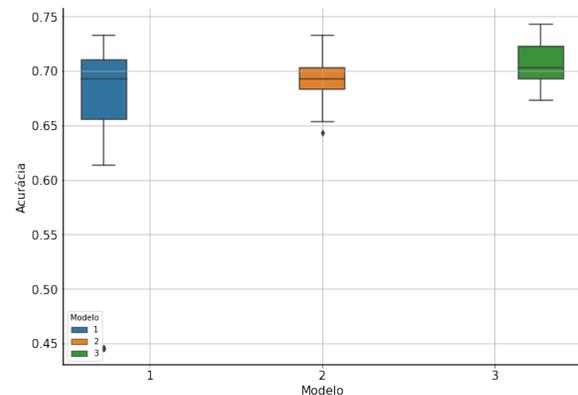


Figura 4. Box plot com os resultados do teste usando 300 imagens para treino.

Quando reduzimos a quantidade de imagens utilizadas para o conjunto de treinamento, como visto na Tabela II para o experimento 2, os modelos utilizados apresentaram um aumento da variabilidade da distribuição da acurácia, Figura

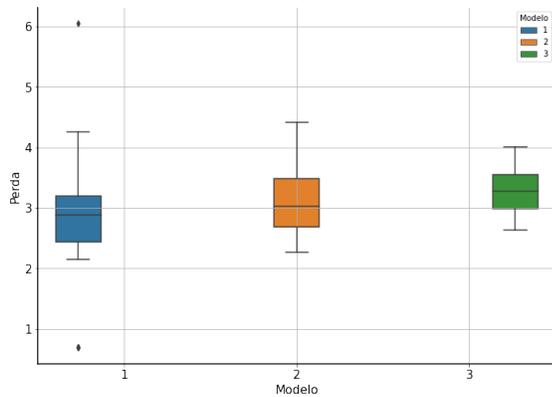


Figura 5. Box plot com os resultados de perda do teste usando 300 imagens para treino.

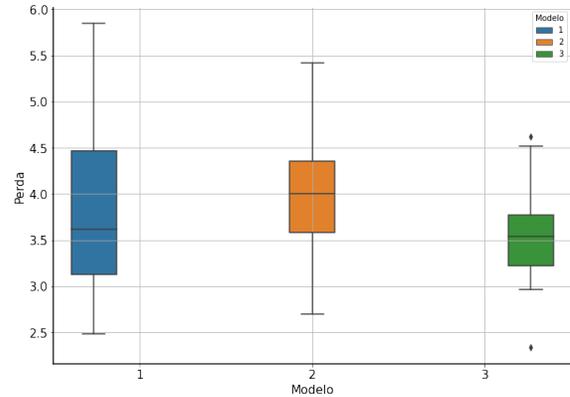


Figura 7. Box plot com os resultados de perda do teste usando 200 imagens para treino.

6. O Modelo 1 obteve um resultado um pouco melhor, deslocando o primeiro quartil em torno de 5%, a mediana em torno de 2% e o terceiro quartil em torno de 5%. Já os modelos 2 e 3 tiveram pouca diferença nos valores de suas medianas, porém os quartis se deslocaram, ampliando a variabilidade da distribuição. Em relação as perdas, Figura 7, vemos que houve um aumento dos valores e uma leve aumento na variabilidade também, com exceção do Modelo 1 que a variabilidade aumentou bastante. Assim, vemos que a definição das classes ficou mais confusa com a redução do conjunto de treinamento, o que é algo esperado.

inclusive, um leve aumento da mediana das distribuições. Assim como, uma leve redução da variabilidade presente, com exceção do Modelo 3. Porém, quando observamos o gráfico das perdas, Figura 9, vemos que os modelos passaram a ter uma definição das classes mais imprecisa. Esse fato pode justificar o leve aumento da acurácia, pois os modelos estariam acertando alguns casos com valores mais próximos do limiar. Quando comparamos os resultados do primeiro experimento, vemos um leve aumento da acurácia acompanhado de um crescimento das perdas.

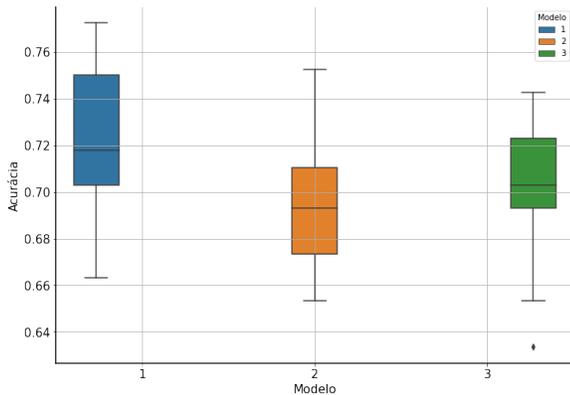


Figura 6. Box plot com os resultados do teste usando 200 imagens para treino.

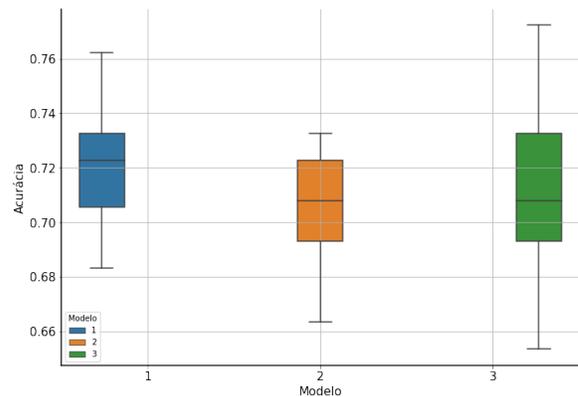


Figura 8. Box plot com os resultados do teste usando 100 imagens para treino.

O nosso último experimento foi realizado com um conjunto ainda menor de dados para treino. Como pode se ver na Tabela II, foram utilizadas 100 imagens do conjunto original e 300 imagens geradas com as técnicas de *data augmentation*. A Figura 8 mostra os *boxplots* para a acurácia dos modelos. Podemos observar que em relação ao experimento dois, Figura 6, os modelos sofreram pouca alteração em sua acurácia, ocorrendo,

Era esperado que com a redução do conjunto de treinamento nos experimentos os modelos adotados apresentariam uma queda na acurácia e um aumento das perdas. Porém, apenas o último caso foi verificado, enquanto que houve um leve aumento da distribuição da acurácia dos modelos. Nossa hipótese é que isso tenha ocorrido devido a casos que os modelos estavam classificando errado, mas com uma definição próxima do limiar das classes. Assim, ao deixarmos esse limiar

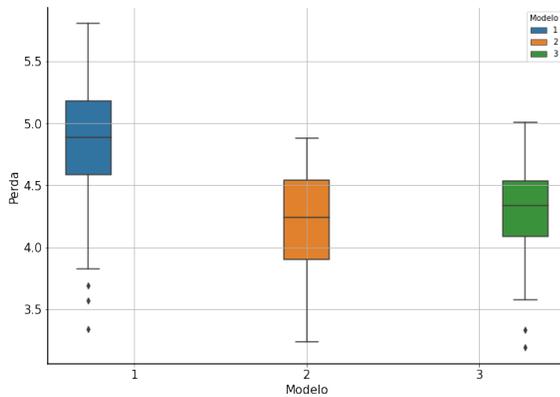


Figura 9. Box plot com os resultados de perda do teste usando 100 imagens para treino.

Tabela III
MÉDIA DAS ACURÁCIAS OBTIDAS A CADA TESTE.

Experimento	1	2	3
Modelo 1	65,18%	72,11%	71,91%
Modelo 2	69,27%	69,47%	70,56%
Modelo 3	70,62%	70,23%	71,32%

menos preciso, devido a redução do conjunto de treinamento, os modelos passaram a classificar corretamente, porém ainda próximo do limiar.

V. CONCLUSÕES

É possível perceber que uma rede neural convolucional pode obter bons resultados na classificação do teste da cópia do desenho dos pentágonos que se cruzam. Foi mostrado neste artigo que a abordagem adotada não conseguiu realizar a classificação de forma satisfatória, tendo uma média de acurácia como vista na Tabela III. Assim, levantamos a hipótese que nosso conjunto de dados, apesar de possuir um tamanho adequado, não possui a variabilidade de casos necessários para que a CNN possa separar as duas classes precisamente. Logo, mantendo esse conjunto de dados, seria necessário criar outras estratégias de treinamento. Uma das possibilidades seria com o aprendizado multimodal. Assim, treinariamos os modelos com os desenhos e os espectros de Fourier. É esperado com essa estratégia que ocorra uma melhora na precisão total do modelo, pois aumentaríamos a quantidade de informação passada para a rede.

AGRADECIMENTOS

Esse estudo foi financiado em parte pela Fundação de Amparo a Ciência e Tecnologia do Estado de Pernambuco (FACEPE) sob concessão IBPG-0844-3.04/17.

REFERÊNCIAS

[1] “10 facts on ageing and health,” World Health Organization. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/10-facts-on-ageing-and-health>

[2] “Dementia,” World Health Organization. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/dementia>

[3] N. A. F. Frota, R. Nitrini, B. P. Damasceno, O. V. Forlenza, E. Dias-Tosta, A. B. da Silva, E. Herrera Junior, and R. M. Magaldi, “Criteria for the diagnosis of alzheimers disease: recommendations of the scientific department of cognitive neurology and aging of the brazilian academy of neurology,” *Dement. neuropsychol.*, 2011.

[4] J. E. Galvin and C. H. Sadowsky, “Practical guidelines for the recognition and diagnosis of dementia,” *The Journal of the American Board of Family Medicine*, vol. 25, no. 3, pp. 367–382, 2012.

[5] L. Trojano and G. Gainotti, “Drawing disorders in alzheimer’s disease and other forms of dementia,” *Journal of Alzheimer’s Disease*, vol. 53, no. 1, pp. 31–52, 2016.

[6] K. I. Shulman, N. Herrmann, H. Brodaty, H. Chiu, B. Lawlor, K. Ritchie, and J. M. Scanlan, “Ipa survey of brief cognitive screening instruments,” *International psychogeriatrics*, vol. 18, no. 2, pp. 281–294, 2006.

[7] L. Yang, J. Yan, X. Jin, Y. Jin, W. Yu, S. Xu, H. Wu, Y. Xu, and C. Liu, “Estimation of diagnostic performance of dementia screening tests: mini-mental state examination, mini-cog, clock drawing test and ascertain dementia 8 questionnaire,” *Aging & mental health*, vol. 22, no. 8, pp. 948–952, 2018.

[8] J. E. Martinelli, J. F. Cecato, M. O. Martinelli, B. A. R. d. Melo, and I. Aprahamian, “Performance of the pentagon drawing test for the screening of older adults with alzheimer’s dementia,” *Dementia & neuropsychologia*, vol. 12, pp. 54–60, 2018.

[9] A. L. Teldeschi, M. Perez, M. A. Sanchez, and R. A. Lourenço, “O uso de testes de fluência verbal como ferramenta de rastreio cognitivo em idosos,” *Revista Hospital Universitário Pedro Ernesto (TÍTULO NÃO-CORRENTE)*, vol. 16, no. 1, pp. 56–60, 2017.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[11] W. Rawat and Z. Wang, “Deep convolutional neural networks for image classification: A comprehensive review,” *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[12] O. Seddati, S. Dupont, and S. Mahmoudi, “Deepsketch: deep convolutional neural networks for sketch recognition and similarity search,” in *2015 13th international workshop on content-based multimedia indexing (CBMI)*. IEEE, 2015, pp. 1–6.

[13] H. Zhang, S. Liu, C. Zhang, W. Ren, R. Wang, and X. Cao, “Sketchnet: Sketch classification with web images,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1105–1113.

[14] M. F. Folstein, S. E. Folstein, and P. R. McHugh, ““mini-mental state”: a practical method for grading the cognitive state of patients for the clinician,” *Journal of psychiatric research*, vol. 12, no. 3, pp. 189–198, 1975.

[15] “module: util,” scikit-image. [Online]. Available: https://scikit-image.org/docs/0.13.x/api/skimage.util.html#skimage.util.random_noise

[16] “module: transform,” scikit-image. [Online]. Available: <https://scikit-image.org/docs/dev/api/skimage.transform.html?highlight=transform%20rotate#skimage.transform.rotate>

[17] “sklearn.model_selection.train_test_split,” scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html