

Classificação de Imagens de Satélite com Redes Neurais Convolucionais

Danilo Sampaio Magiri ^a e Celso Ap.de França ^b

Departamento de Engenharia Elétrica da Universidade Federal de São Carlos

As imagens de satélites são utilizadas em diversas áreas, como o agronegócio, planejamento urbano, monitoramento ambiental, entre outras. Essas imagens possuem uma grande e complexa quantidade de dados e a aplicação de técnicas de *Machine Learning* podem ajudar na classificação automática das informações contidas nelas.

O presente artigo tem como objetivo o desenvolvimento teórico e prático de uma rede neural convolucional que possa classificar imagens de satélites retiradas da Bacia Amazônica, podendo assim ser utilizado como uma ferramenta de monitoramento ambiental e detecção de mudanças no meio ambiente.

Para a realização deste trabalho foi utilizado o *dataset* da *Planet: Understanding The Amazon from Space*. Este *dataset* possui 40.479 imagens para o treinamento do modelo e 40.479 imagens para teste, divididas em 17 categorias distintas que englobam condições atmosféricas e uso/cobertura da terra.

Ao final do artigo o leitor terá a compreensão da teoria por trás das redes neurais convolucionais a partir da apresentação prática de um modelo desenvolvido e com acurácia superior a 95% na classificação de imagens de satélite.

Palavras-Chave – Classificação de Imagens, Imagens de Satélites, Inteligência Artificial, *Machine Learning*, Redes Neurais.

1. INTRODUÇÃO

De acordo com os dados da Global Forest Watch, somente no ano de 2021, cerca de 11 milhões de hectares de cobertura florestal foram perdidos nos trópicos devido às causas humanas. Deste total, 3,75 milhões de hectares são em florestas tropicais primárias e o Brasil corresponde a 40% desta perda [1].

Com o avanço da tecnologia, investimento e estudo em satélites, cada vez mais é possível utilizar estas imagens para diversas finalidades, desde o auxílio na agricultura até para identificar e controlar o avanço do desmatamento em áreas florestais. Imagens de satélites com resolução espacial de 30 metros, como o *Landsat-3* foram utilizadas durante muito tempo em pesquisas dedicadas ao rastreamento de mudanças nas florestas, entretanto, hoje, o projeto e lançamento da constelação de satélites da *Planet* consegue gerar imagens com resolução de 3-5 metros, o que possibilita uma maior eficácia em mapear desmatamento em pequena escala e degradação ambiental [2].

Portanto, com base neste cenário atual, este trabalho tem como objetivo o desenvolvimento de um sistema baseado em conceitos de *machine learning* que possa classificar imagens de satélite da bacia amazônica por condições atmosféricas e

condições de uso da terra/cobertura da terra. A partir da classificação, estes dados poderão ser utilizados para auxiliar entidades governamentais e partes interessadas na localização de desmatamentos e atividades humanas de degradação das florestas com maior rapidez e eficácia.

2. TRABALHOS RELACIONADOS

A classificação de imagens de satélite por redes neurais convolucionais (CNN) está em rápido desenvolvimento e tem atraído muita atenção e pesquisa nos últimos anos. Empresas como a *Planet*, responsável pelo conjunto de dados deste artigo, tem investido em satélites com maiores resoluções de captura de imagens e com isso, o uso de redes neurais convolucionais na classificação de imagens vem se mostrando bastante eficaz.

Zheng et al. [3] propôs um modelo de CNN baseada em uma arquitetura de rede residual para a classificação de imagens de satélite. Este modelo foi treinado utilizando imagens de alta resolução e obteve desempenho superior quando comparado aos métodos convencionais de classificação de imagem.

Já o modelo proposto por Zheng et al. [4] utilizou uma abordagem de aprendizado por transferência para a classificação de imagens de satélite, na qual, uma CNN pré-treinada com imagens de baixa resolução se mostrou extremamente eficaz no treinamento de um classificador para imagens de alta resolução.

Outro modelo interessante foi o utilizado por Chen et al. [5] que propuseram uma CNN baseada em redes profundas para a classificação de imagens de satélite. Esta rede foi projetada para lidar com problemas complexos de alta dimensionalidade e foi treinado utilizando um extenso *dataset*.

A partir destes trabalhos anteriormente citados e em conjunto com este artigo, é possível observar a capacidade do uso de inteligência artificial na classificação de imagens de satélite e a diversidade de temas que podem ser elencados como planejamento urbano, controle da degradação de florestas, controle de áreas de cultivo, entre outros.

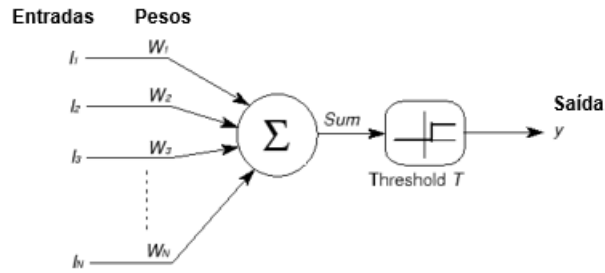
3. FUNDAMENTAÇÃO TEÓRICA

O neurônio biológico, também conhecido como célula nervosa [6], é responsável pelo processamento de informações e é o objeto estudado para todo o desenvolvimento da tecnologia de redes neurais artificiais. Warren McCulloch e Walter Pitts em 1943 propuseram um modelo computacional (Figura 1) baseado em matemática e algoritmos denominado de *threshold logic* [7], lógica de limiar na tradução literal, em que o neurônio possuía somente uma saída a partir do valor de suas diversas entradas.

^a E-mail autor: danilomagiri@estudante.ufscar.br

^b E-mail orientador: celsofr@ufscar.br

Figura 1 – Modelo de neurônio por McCulloch-Pitts



Fonte: Figura alterada de

<https://ideiasesquecidas.com/2017/02/06/um-neuronio-de-mcculloch-pitts/>

Quando comparado ao neurônio biológico as Entradas I_n correspondem aos impulsos elétricos recebidos de outros neurônios e, assim, cada uma dessas entradas é multiplicada por um peso sináptico W_n e somadas. Então a partir da soma ponderada das entradas é gerada uma saída 1, caso seja ultrapassado o limite “T” ou 0 caso contrário. Esta função linear é expressa conforme Equação 1, onde Y é a saída e T a função de ativação [8].

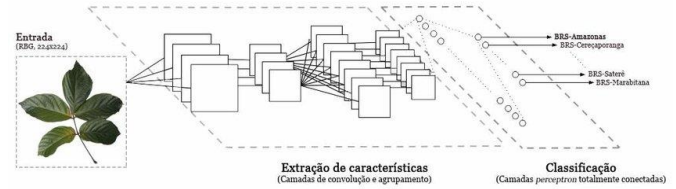
$$Y = T \left(\sum_{n=0}^n I_n \cdot W_n \right) \quad (1)$$

A função de ativação tem por objetivo limitar a saída desse neurônio e introduzir não-linearidade ao modelo. Existem diversas funções de ativação como a *threshold* que entrega uma saída binária ao modelo, a *sigmoid* que é bastante utilizada com o intuito de apresentar os resultados em forma de probabilidade, uma vez que a função percorre entre 0 e 1, a *ReLU* que é utilizada para eliminar as entradas negativas do modelo.

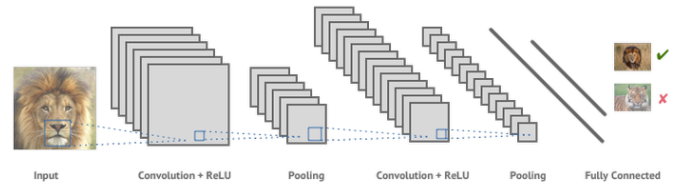
As redes neurais profundas, ou *Deep Learning*, são uma área da inteligência artificial e é composta por várias camadas (*multilayer*) de neurônios artificiais. Cada camada é responsável por extrair características dos dados de entrada e, assim, passar para a camada seguinte. Camadas posteriores extraem características mais complexas quando comparadas com camadas iniciais [9].

Uma das arquiteturas de redes neurais profundas mais utilizadas é a Rede Neural Convolutiva (RCN), também conhecida como *Convolutional Neural Network* (CNN), modelo apresentado e utilizado como base teórica deste trabalho, é projetado especialmente para processamento de dados em grade, como por exemplo, imagens. As CNNs usam das camadas de convolução - a Figura 2 exemplifica as múltiplas camadas que compõem o modelo de uma rede neural convolutiva - para extrair características relevantes dos dados de entrada e aplicar filtros, ou pesos, gerando regiões de ativação. A Figura 3 exemplifica a redução de dimensionalidade nas camadas de *pooling*, responsáveis por filtrar os *pixels* com as características mais importantes das imagens de entrada, e tem por objetivo reduzir a dimensionalidade e a partir das camadas conectadas gerar previsões, além disso, a redução de dimensionalidade auxilia na rapidez de resposta da rede e na eficiência do modelo [10].

Figura 2 – Rede Neural Convolutiva.



Fonte: Vargas et al. 2016

Figura 3 – Redução de dimensionalidade a partir das camadas de *pooling*.

Fonte: Retirada de

<https://www.electricalibrary.com/en/2018/11/20/what-are-convolutional-neural-networks/>

Para atingir uma previsão satisfatória são utilizados os recursos de treinamento como por exemplo o *backpropagation*, algoritmo no qual reajusta os pesos da rede percorrendo da saída até a entrada para minimizar a função de erro fazendo a diferença entre as previsões da rede e os valores esperados.

Outra técnica fundamental para o treinamento de modelos de *machine learning* é a otimização de redes. Neste documento a técnica empregada é o algoritmo de *Adam*, que utiliza uma média móvel exponencial do gradiente e da sua variância para adaptar a taxa de aprendizagem em cada camada de iteração ao longo de todo o treinamento. Além disso, ele também acelera a convergência dos dados evitando assim a oscilação do gradiente. Estes fatores possibilitam a adaptação do algoritmo em diferentes tipos de dados e cenários de treinamento, incluindo os modelos de classificação de imagens, base teórica deste documento [11].

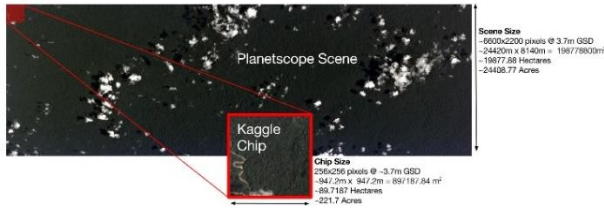
Durante o treinamento de redes neurais é possível aparecer problemas que afetam o desempenho e acurácia do modelo. Dois problemas comuns são o *overfitting* e *underfitting*. O primeiro ocorre quando o modelo se ajusta demasiadamente aos dados de treinamento, o que acaba atrapalhando na capacidade de generalizar novos dados. Já o *underfitting* ocorre quando o modelo não é complexo o suficiente para se ajustar às relações presentes nos dados de treinamento, o que acaba por resultar em desempenho insatisfatório em novas previsões [12].

4. METODOLOGIA

O *dataset* utilizado no desenvolvimento deste projeto foi retirado do *kaggle* - uma plataforma *Google* de aprendizado e transmissão de conhecimento sobre *Data Science*, que também apresenta competições premiadas como a utilizada neste trabalho - “*Planet: Understanding the Amazon from Space*” em que cada imagem de 6600x2200 *pixels* derivou em vários *chips* de resolução 256x256 *pixels* a partir da análise de *frame* completo utilizando satélite de 04 (quatro) bandas em órbita síncrona ao Sol (Sun Synchronous Orbit – SSO) e na Estação Espacial

Internacional (International Space Station – ISS) fornecidas pela empresa *Planet*, como mostrado na Figura 4.

Figura 4: *Chips* utilizados no desenvolvimento do modelo da CNN.

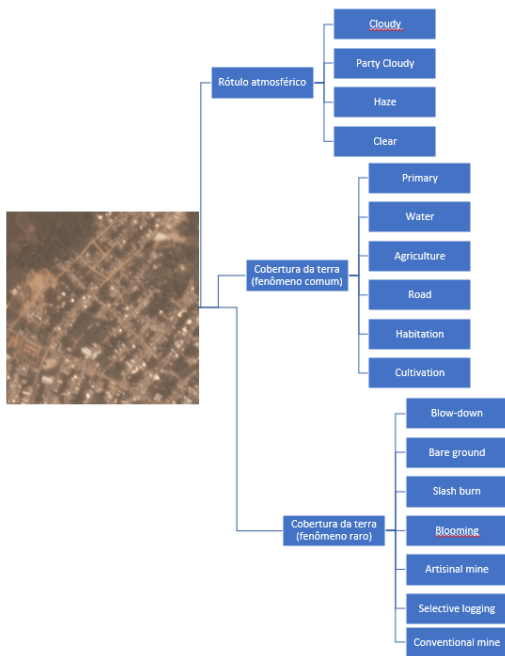


Fonte: Retirada de

<https://www.kaggle.com/c/planet-understanding-the-amazon-from-space/data>

O conjunto de dados possui 40.479 imagens para treinamento do modelo e 40.479 imagens para teste. Com ajuda das equipes da *Planet* de Berlim e de São Francisco, cada imagem de treino foi rotulada de acordo com as condições atmosféricas, fenômenos comuns de cobertura da terra e fenômenos raros de cobertura da terra. Os 17 tipos de classificação são mostrados no Diagrama 1 abaixo: [13].

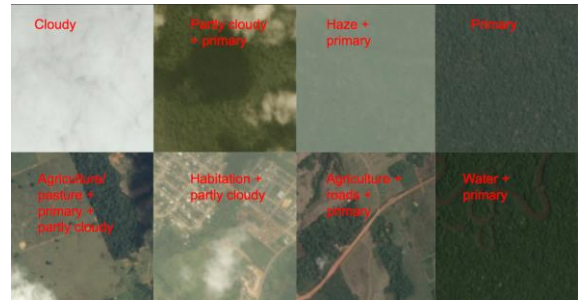
Diagrama 1: Tipos de classificação.



Fonte: Autoria Própria.

As imagens podem ter mais de um rótulo para os fenômenos de cobertura de terra e nenhum ou apenas um para as condições atmosféricas. Na Figura 5 são mostrados exemplos de como as imagens foram classificadas.

Figura 5: Classificação de imagens a partir dos rótulos pré-estabelecidos.



Fonte: Autoria Própria.

Para quantificar os acertos e erros do modelo são utilizadas métricas como a acurácia em que a quantidade de acertos é dividida pelo todo. Porém para dados que apresentam discrepâncias nem sempre a acurácia será a melhor maneira de se expressar a taxa de acerto da rede neural, pois não leva em consideração os erros de classificação do modelo. A Tabela 1 exemplifica uma Matriz de Confusão em que apresenta os valores verdadeiros e falsos que um dado pode ser classificado.

Tabela 1: Matriz de Confusão.

		Valor Predito	
		Sim	Não
Imagem Real	Sim	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Fonte: Autoria Própria.

Então para este trabalho foi utilizada, além da acurácia, a métrica F_β score que a partir da média harmônica dos valores de precisão (considera os falsos positivos) e dos valores de *recall* (considera os falsos negativos) resulta em uma boa classificação em 1 e uma má predição em 0. Na Equação 2 é mostrada a fórmula utilizada para calcular o F_β score em que o parâmetro β determina o peso do *recall* na pontuação combinada, permitindo ajustar a importância dada a cada uma dessas medidas [14]. Este tipo de métrica se mostra muito útil quando é necessário avaliar o desempenho de um modelo em relação a um conjunto de dados desbalanceados, como é o caso deste trabalho. Para uma melhor apresentação, foi plotado na Figura 6 o gráfico com as quantidades de cada classe das imagens de treino, sendo assim, é possível observar a diferença entre o evento de maior ocorrência (*primary*) com o evento de menor ocorrência (*blow down*).

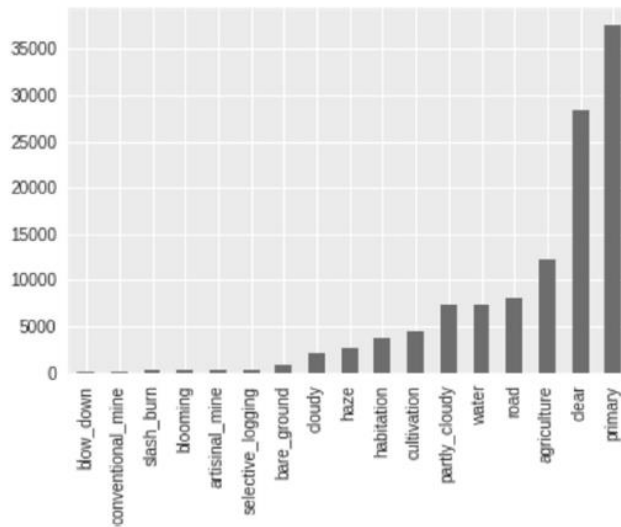
$$F_\beta = (1 + \beta^2) \frac{p \cdot r}{\beta^2 p + r} \tag{2}$$

Onde, p e r são precisão e *recall*, definidos como

$$p = \frac{tp}{tp + fp}, \quad r = \frac{tp}{tp + fn}$$

Onde, tp , fp e fn são o número de verdadeiros positivos, falsos positivos e falsos negativos.

Figura 6: Gráfico de ocorrência de cada um dos 17 eventos de classificação utilizadas no trabalho.



Fonte: Autoria Própria.

Para acelerar o treinamento sem obter perdas significativas para a acurácia do modelo foi feita uma redução na resolução das imagens para 128x128 *pixels*, isso permite uma resposta mais rápida para o modelo previsto.

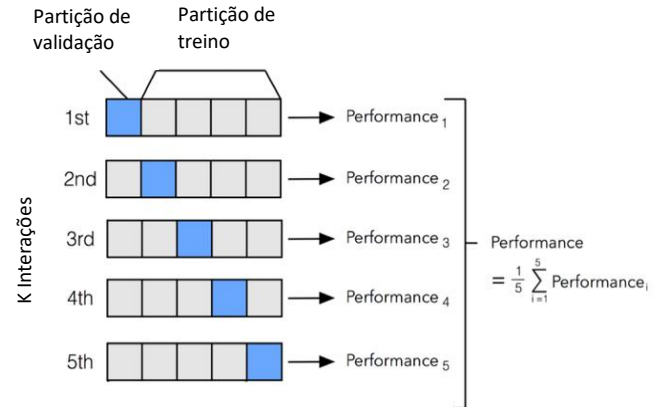
Com isso, a implementação do modelo seguiu os parâmetros listados abaixo:

- Input: Imagens 128x128 *pixels*.
- 40.479 imagens para o treinamento da CNN.
- 40.479 imagens para o teste do modelo da CNN.
- Dropout de 10% a cada camada de *Pooling*.
- 25 épocas, medida do número de vezes que todo o conjunto de dados é treinado pelo modelo.
- Threshold $[0,2] * 17$, com isso limitamos que quando a previsão for maior que 0,2 para cada uma das 17 classes o modelo classifica como positivo.
- Em cada camada de convolução foi utilizada a função de ativação ReLU.
- Otimizador de *Adam* e *binary_crossentropy* (função de erro).
- Função de ativação *Sigmoid*.
- Validação cruzada de *k-partições* (*K-fold-cross-validation*), em que os dados de treino foram divididos aleatoriamente em 5 partes iguais (*folds*).

5. RESULTADOS E DISCUSSÕES

Para o treinamento foi utilizada a técnica de validação cruzada, que consiste em dividir aleatoriamente o *dataset* em *k* grupos e escolher apenas 1 para ser testado, enquanto os *k-1* grupos são usados para treinar o modelo. Após anotar o valor da métrica o modelo é descartado para treinar o grupo seguinte até finalizar os *k* grupos e, assim, realizar a média aritmética do modelo como um todo. Na Figura 7 é mostrado como funciona o método de validação cruzada usada neste trabalho com *k = 5*, ou seja, *5-fold-cross-validation*.

Figura 7: Validação cruzada de 5 partições.



Fonte: Adaptada de

https://zitaoshen.rbind.io/project/machine_learning/machine-learning-101-cross-validation/

Após finalizar o treinamento das 5 partições para o *dataset* de treino, foram gravados os pesos que o modelo definiu em cada *fold* para cada uma das 17 classes do projeto em questão e rodado o modelo da CNN no *dataset* com 40.479 imagens de teste e as acurácias de cada época, a Tabela 2 é a amostra da acurácia dos dados de teste por época para o *fold 1* da rede. É possível observar que o modelo possui uma boa taxa de acerto na classificação das imagens de teste em todas as partições, mostrando que a distribuição dos pesos está coerente com o que se espera e os parâmetros escolhidos estão adequados. Esses valores são utilizados para plotar a curva de aprendizado do modelo.

Tabela 2 – Resultado da classificação dos dados de teste do *Fold 1*.

Epoch	Accuracy	Epoch	Accuracy
1	0.9122	14	0.9453
2	0.9183	15	0.9446
3	0.9249	16	0.9445
4	0.9277	17	0.9458
5	0.9354	18	0.9467
6	0.9363	19	0.9487
7	0.9390	20	0.9485
8	0.9420	21	0.9475
9	0.9407	22	0.9473
10	0.9417	23	0.9491
11	0.9426	24	0.9484
12	0.9435	25	0.9494
13	0.9428		

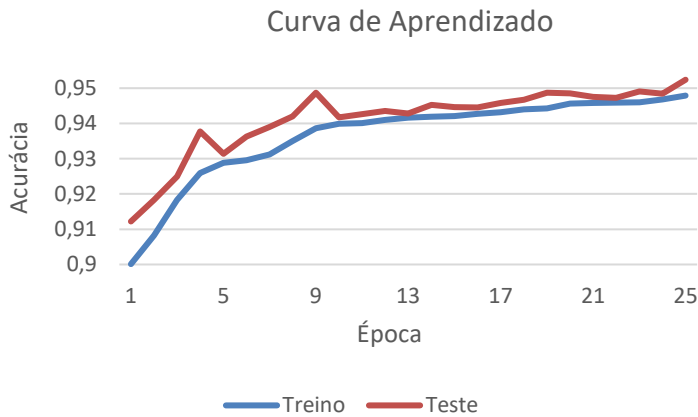
Fonte: Autoria Própria.

Com isso, foi possível calcular o melhor valor do F_{β} score que foi de 0.7396.

A seguir, foi plotado o gráfico, Figura 8, em que a curva laranja representa a média aritmética da acurácia por época dos dados de teste do modelo e a curva azul a acurácia média por época dos dados de treinamento. É possível identificar pela

proximidade das curvas de aprendizado que o modelo não apresentou *overfitting* e *underfitting*, ou seja, não houve treinamento excessivo ou mau treinamento da rede neural.

Figura 8 – Acurácia do modelo.



Fonte: Autoria própria.

A partir dos resultados gerados na interação do treinamento com a validação dos dados de teste, foi possível realizar a classificação das imagens criando um vetor em que cada valor representa a pontuação (*score*) de cada classe para cada imagem do *dataset* de teste. Isso representa a previsão por *fold* que cada classe tem em cada imagem. Na Tabela 3 é mostrado o *score* alcançado no *fold 1*.

Tabela 3 – Pontuação alcançada no *fold 1*.

	Imagem 1	Imagem 2	Imagem 3	Imagem 4	Imagem 5
Agriculture	0.03585	0.06938	0.11503	0.01720	0.05452
Artisinal_mine	0.00115	0.00110	0.00132	0.00019	0.00004
Bare_ground	0.00274	0.00091	0.00110	0.00084	0.00072
Blooming	0.00331	0.00292	0.00001	0.00140	0.00000
Blow_down	0.00418	0.00701	0.00009	0.00094	0.00000
Clear	0.98246	0.90213	0.00106	0.98036	0.00079
Cloudy	0.00004	0.00000	0.01626	0.00001	0.42050
Conv._mine	0.00027	0.00014	0.00039	0.00005	0.00001
Cultivation	0.02375	0.06586	0.03034	0.01223	0.00565
Habitation	0.00564	0.00927	0.01043	0.00205	0.00725
Haze	0.00062	0.00007	0.00026	0.00010	0.00086
Partly_cloudy	0.01373	0.10622	0.97621	0.01653	0.56873
Primary	0.99964	0.99998	0.98366	0.99990	0.53221
Road	0.02319	0.03075	0.07578	0.01034	0.03407
Select._logging	0.01062	0.01714	0.00007	0.00327	0.00000
Slash_burn	0.00162	0.00146	0.00111	0.00032	0.00002
Water	0.04939	0.04847	0.09872	0.02344	0.07866

Fonte: Autoria Própria.

Disso, foi criado um *loop* para iterar a soma de cada *fold* e, assim, realizar o *score* médio de todas partições.

Após o levantamento da pontuação média das classes para cada imagem foi feito um arquivo em formato *.csv* com a previsão final do resultado do modelo da rede neural

convolucional, em que o valor de cada pontuação por classe determina se um evento está presente ou não na imagem.

A Tabela 4 representa a pontuação obtida das 5 primeiras imagens do conjunto de dados de teste. Por exemplo, pode-se observar que na coluna referente a Imagem 3 da tabela, as pontuações de *partly_cloudy* e *primary* estão acima do valor de 0,2 definido como valor limite do modelo, portanto na Figura 9, após a tabela, é possível observar a ocorrência dos eventos que obtiveram pontuação maior que 0,2.

Tabela 4 – Previsão final do modelo.

	Imagem 1	Imagem 2	Imagem 3	Imagem 4	Imagem 5
Agriculture	0.01772	0.02514	0.04901	0.02037	0.03116
Artisinal_mine	0.00024	0.00021	0.00051	0.00023	0.00017
Bare_ground	0.00141	0.00090	0.00110	0.00084	0.00072
Blooming	0.00739	0.01532	0.00002	0.00956	0.00001
Blow_down	0.00226	0.00604	0.00016	0.00157	0.00002
Clear	0.24540	0.24328	0.00012	0.18860	0.00113
Cloudy	0.00001	0.00000	0.00093	0.00000	0.10956
Conv._mine	0.00004	0.00002	0.00039	0.00003	0.00006
Cultivation	0.01369	0.03759	0.03034	0.01764	0.00921
Habitation	0.00190	0.00339	0.00213	0.00178	0.00389
Haze	0.00083	0.00007	0.00008	0.00004	0.00110
Partly_cloudy	0.00433	0.01023	0.24999	0.08164	0.16421
Primary	0.25114	0.25119	0.24967	0.25119	0.14616
Road	0.00656	0.00621	0.01562	0.00811	0.01462
Select._logging	0.00254	0.00670	0.00018	0.00394	0.00005
Slash_burn	0.00060	0.00123	0.00066	0.00040	0.00008
Water	0.02136	0.01820	0.03964	0.01670	0.02534

Fonte: Autoria Própria.

Figura 9 – Imagem 3 com os eventos *partly_cloud* e *primary*.



Fonte: Autoria Própria.

Baseando-se nas previsões geradas durante o treinamento e teste, apontadas no arquivo de resultados finais, pode-se mostrar na prática como ficaram algumas classificações de imagens. A Figura 10, por exemplo, é a amostra 1 e claramente é possível observar de forma visual os eventos *clear*, *habitation*, *road* e *primary*, e na previsão (mostrada na linha logo abaixo da figura)

foi identificado exatamente o proposto. A Figura 11 (*amostra 2*), outro bom exemplo, representa uma imagem com *partly_cloudy* e *primary*, também estando de acordo com a previsão.

Figura 10: Amostra 1 - Exemplo de boa previsão.



Fonte: Autoria própria.

Figura 11 – Segundo exemplo de boa previsão.



Fonte: Autoria própria.

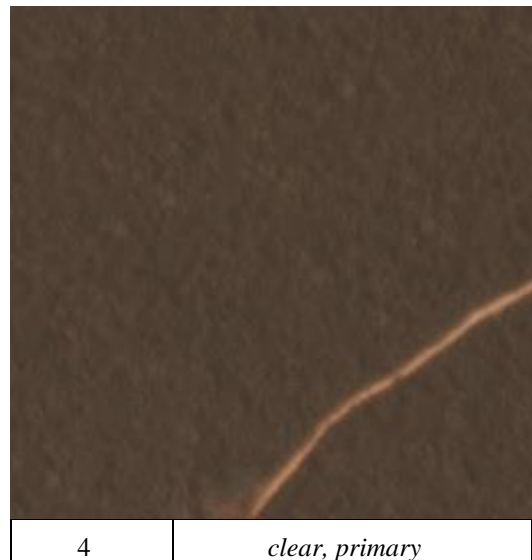
Já a Figura 12, amostra 3, é um exemplo de má predição, em que visualmente é um evento claro de neblina (*haze*) e foi classificado como *primary*. A Figura 13 que representa a amostra 4 deveria ser classificada como *clear, primary e road*, porém só atingiu pontuação satisfatória para os eventos *clear e primary*, portanto, classificou parcialmente a imagem analisada.

Figura 12 – Exemplo de má previsão.



Fonte: Autoria própria.

Figura 13 – Segundo exemplo de má previsão.



Fonte: Autoria própria.

Além das previsões que apresentaram erro, houve imagens em que a pontuação mínima de acordo com o *threshold* não foi atingida em nenhuma classe e, portanto, o modelo não foi capaz de classificá-las, o que infere outro erro indesejado ao modelo.

O tempo total de execução do treinamento e da classificação dos dados de teste foi de 6h 32 min e foi utilizado como *hardware* um computador de 16 *gigabytes* de memória RAM e uma GPU do modelo RTX 3050 com memória VRAM GDDR6 de 8 *gigabytes*.

6. CONCLUSÕES

Nesse trabalho foi passada uma visão teórica e prática sobre as redes neurais artificiais, com ênfase no uso de redes neurais convolucionais em projetos de classificação de imagens de satélites.

Por meio de tabelas e imagens, foi demonstrada a eficácia do modelo proposto em classificar o *dataset* de teste disponibilizado

pelo *kaggle*, que ficou acima dos 95% de acurácia e um F_β score de 0.7396. Devido à falta de *hardware* disponível, como por exemplo, servidores *cluster* que por características ajudaria na rapidez do treinamento do modelo, não foi possível realizar a validação dos dados por mais épocas, o que possibilitaria um incremento na acurácia do modelo apresentado.

Para melhorias futuras, vale a pena experimentar um tratamento durante a fase de pré-processamento das imagens, utilizando *softwares* específicos para essa finalidade. Isso poderia ajudar na resolução do problema de falta de classificação em imagens com neblina ou parcialmente cobertas por nuvens. Além disso, utilizar imagens com maiores resoluções também possibilitaria uma maior assertividade do modelo projetado.

7. REFERÊNCIAS

- [1] WEISSE, M and GOLDMAN, L. Perda florestal permanece resistentemente alta em 2021. **Global Forest Watch**. 28 abr. 2022. Disponível em: <https://www.globalforestwatch.org/blog/pt/data-and-research/dados-globais-de-perda-de-cobertura-de-arvore-2021/>. Acesso em: 06 fev. 2023.
- [2] WORLD ECONOMIC FORUM. 4 ways Landsat satellites have dramatically changed how we see the world. WeForum, 2023. Disponível em: <https://www.weforum.org/agenda/2023/03/4-ways-landsat-satellites-have-dramatically-changed-how-we-see-the-world/>. Acesso em: 21 mar. 2023.
- [3] ZHENG, Y., ZHAO, C., & WANG, J. (2018). A CNN-Based Residual Network for Classification of Satellite Images. *Remote Sensing*, 10(9), 1397.
- [4] ZHENG, Y., WANG, J., & ZHAO, C. (2020). Transfer Learning Based on a Pre-Trained CNN for Classification of Satellite Images. *Remote Sensing*, 12(3), 516.
- [5] CHEN, P., LIU, Y., GHAMISI, P., PLAZA, A., & LI, J. (2018). A Deep Learning Framework for Hyperspectral Image Classification Using Convolutional Neural Network and Extreme Learning Machines. *Remote Sensing*, 10(4), 559.
- [6] HECHT-NIELSEN, R. Artificial neural networks: a tutorial. IEEE, 1992. Disponível em: <https://ieeexplore.ieee.org/document/67838>. Acesso em: 06 fev. 2023.
- [7] CHVATAL, V. McCulloch-Pitts brains and pseudorandom functions. *Mathematics Magazine*, v. 64, n. 1, p. 21-37, 1991. Disponível em: <https://ieeexplore.ieee.org/document/7482919>. Acesso em: 06 fev. 2023.
- [8] CHVATAL, V. McCulloch-Pitts brains and pseudorandom functions. *Mathematics Magazine*, v. 64, n. 1, p. 21-37, 1991. Disponível em: <https://ieeexplore.ieee.org/document/7482919>. Acesso em: 06 fev. 2023.
- [9] GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning. MIT Press, 2016. Capítulo 5 - "Machine Learning Basics". Seção 5.1 - "What Is a Neural Network?". p. 161-192.
- [10] LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436-444, 2015. Disponível em: <https://www.nature.com/articles/nature14539>. Acesso em: 13 fev. 2023.
- [11] KINGMA, D.P; BA, J.L. Adam: A Method for Stochastic Optimization. University of Amsterdam, University of Toronto, 2015.D
- [12] KOEHRSEN, W. Overfitting vs. Underfitting: A Complete Example. Towards data science, 2018.E
- [13] KAGGLE. Planet: Understanding the Amazon from Space. 2017. Disponível em: <https://www.kaggle.com/c/planet-understanding-the-amazon-from-space>. Acesso em: 17jan. 2023.
- [14] ABBAS, M. Convolutional Neural Network for Satellite Image Classification. *International Journal of Advanced Computer Science and Applications*, v. 9, n. 10, p. 49-55, 2018.