



Classificação de Imagens de Resíduos Eletrônicos Usando Redes Neurais Convolucionais

Jose Alexandre F Silva,¹ Francisco Louzada Neto²
ICMC-USP

1 Introdução

O descarte ambientalmente correto de equipamentos elétricos e eletrônicos em fim de vida útil é um assunto de extrema relevância na sociedade atual, em que o constante surgimento de novos dispositivos incentiva seu consumo desenfreado, o que acaba por resultar na geração de enormes volumes de resíduos. Entretanto, a falta de tecnologia adequada para classificação e separação destes materiais impõe um grande desafio ao setor produtivo, já que a predominância de tarefas manuais, além de consumir tempo demais, ainda ocasiona perda de informações valiosas, bem como aumento dos riscos ocupacionais aos quais os profissionais envolvidos ficam expostos.

Logo, o desenvolvimento e a aplicação de novas técnicas para a otimização do processo de reciclagem consiste em fator preponderante para o desenvolvimento de uma Gestão Verde da Cadeia de Suprimentos. Neste sentido, alguns trabalhos têm sido propostos na área de classificação de imagens de resíduos, mas poucos abordam especificamente aqueles oriundos de equipamentos elétricos e eletrônicos, cujas especificidades devem ser tratadas de maneira apropriada quaisquer que sejam os métodos propostos.

Este artigo tem como objetivo apresentar uma abordagem para classificação de imagens de resíduos eletrônicos através de técnicas de Aprendizagem Profunda, notadamente Redes Neurais Convolucionais do tipo ResNet, já que estas têm se destacado na área de Visão Computacional. Após a seleção e a implementação dos algoritmos adequados, serão apresentadas comparações no que diz respeito ao tempo de execução do treinamento e à acuracidade medida a partir do conjunto de validação. Os resultados poderão servir de base para futuros trabalhos que almejem a automação do processo de classificação e separação de materiais nos estágios iniciais de uma planta de reciclagem.

¹josealexandre@usp.br

²louzada@icmc.usp.br

2 Redes Neurais Convolucionais

As Redes Neurais Convolucionais (*Convolutional Neural Network - CNN*) constituem uma área de Aprendizagem Profunda (*Deep Learning*) que foi inspirada no córtex visual e está associada ao desenvolvimento do campo de Visão Computacional, para classificação de imagens. Redes Neurais consistem em conjuntos de funções não-lineares dependentes. Cada função individual pode ser representada por um neurônio (*perceptron*), cujos componentes básicos são: entrada, peso, função de ativação e saída.

O processo de classificação consiste em atribuir à imagem uma determinada classe ou a probabilidade de pertencer a esta classe. Quando um computador vê uma imagem, esta nada mais é do que uma matriz de pixels, com valores variando entre 0 e 255, possuindo três canais de cores quando trata-se de uma imagem RGB (*Red, Green, Blue*).

Em uma arquitetura CNN, a imagem de entrada percorre uma série de processamentos (camadas) a fim de obter uma saída que possa permitir a classificação da imagem. Há diversos tipos de camadas em uma CNN, entre as quais temos: camadas de convolução (*convolutional*), camadas de aglomeração (*pooling*) e camadas totalmente conectadas (*fully connected*). Os primeiros estágios presentes em uma CNN, ou seja, suas primeiras camadas convolucionais e de *pooling*, são tidos como os mais importantes, sendo responsáveis pela maior parte do processamento computacional [5].

A Figura 1 mostra a arquitetura simplificada de uma Rede Neural Convolutiva:

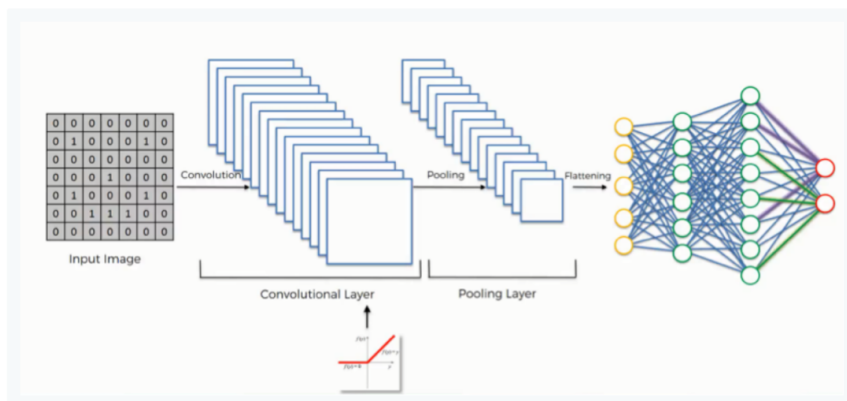


Figura 1: Arquitetura simplificada de uma Rede Neural Convolutiva. Fonte: SuperDataScience

As ResNets fazem parte de uma família de arquiteturas de redes baseadas no conceito de aprendizado profundo residual e foi introduzida por He et al. Este modelo conquistou a primeira posição no ILSVRC 2015 (*ImageNet Large Scale Visual Recognition Challenge*), obtendo um *top-5 error* no valor de 3,57%.

A primeira arquitetura ResNet foi uma ResNet-34, que envolvia a inserção de conexões de atalho (*shortcut connections*) para transformar uma rede mais “simples” em sua contraparte de rede residual. Neste caso, a rede mais “simples” foi inspirada na VGG, com camadas de convolução 3x3. Porém, quando comparadas às redes VGG, as ResNets possuem menos filtros e menor complexidade. Além disso, foram seguidas duas simples regras de *design*: as camadas tinham o mesmo

número de filtros para o mesmo tamanho de mapa de características de saída, e o número de filtros dobrava caso o tamanho do mapa de características fosse reduzido à metade, a fim de preservar a complexidade de tempo por camada. Consistia de 34 camadas ponderadas.

Embora a arquitetura ResNet-50 seja baseada no modelo acima descrito, existe uma diferença fundamental: o bloco de construção (*building block*) foi modificado para um *bottleneck design* devido às preocupações com o tempo decorrido para treinar as camadas. O modelo ResNet-50 utilizou uma pilha de 3 camadas ao invés de duas, obtendo maior acuracidade em relação ao modelo ResNet-34.

Grandes redes residuais, como ResNet-101 ou ResNet-152, são construídas usando mais blocos de 3 camadas. E mesmo com maior profundidade de rede, a ResNet de 152 camadas, por exemplo, tem complexidade muito menor (em 11,3 bilhões de FLOPS) do que as redes VGG-16 ou VGG-19 (15,3/19,6 bilhões de FLOPS) [3].

3 Experimentos e Resultados

Quatro modelos ResNet pré-treinados (com profundidades 34, 50, 101 e 152, respectivamente) foram aplicados em um *dataset* composto por 1.670 imagens de resíduos eletrônicos associados à operação de *Scrap e Waste Disposal* de uma grande empresa de TI. Tais imagens foram previamente rotuladas em 12 classes: (1) Alumínio, (2) Baterias, (3) Cabos e Fios, (4) Papel e Papelão, (5) Conectores, (6) Cobre, (7) Ferro, (8) Fitas Magnéticas, (9) Placas de Circuitos Impressos, (10) Plástico, (11) *Toner* ou *Ribbon* de Impressoras e (12) Tubos ou Telas.

Todos os modelos foram treinados através da plataforma Google Colab, utilizando uma GPU com memória de 16 GB. Foi utilizada a linguagem de programação Python com a biblioteca *fast.ai*, construída sobre o *framework* PyTorch. Cada modelo foi treinado por 10 *epochs* com as camadas iniciais “congeladas” (*freeze*) (ou seja, foram utilizados somente os parâmetros de treinamento das últimas *fully connected layers*) e, posteriormente, tais modelos foram ajustados por mais 10 *epochs* após o “decongelamento” (*unfreeze*) de todas as camadas. Antes de cada congelamento / descongelamento, foram determinadas as *learning rates* apropriadas.

As seguintes técnicas de pré-processamento foram aplicadas ao configurar os parâmetros de treinamento: *Train/Validation split: 80%-20%; 5-Fold Cross Validation; Oversampling; Augmentation Transforms: flip, rotate, zoom, lighting; Batch Size: 64; Image Size: 224x224; Deterministic Learner; Adam Optimization algorithm*. Além disso, a grande maioria das imagens passou previamente por um processo de segmentação de *background* através de uma rede neural pré-treinada denominada U2-Net [4].

As métricas utilizadas para comparação dos resultados referem-se à acuracidade convencional (top-1), à acuracidade top-5 e ao tempo total de treinamento. A acuracidade top-1 representa o número de predições corretas (classe com a maior probabilidade) dividido pelo número total de predições, enquanto a acuracidade top-5 significa que qualquer uma das 5 respostas de probabilidade mais alta do modelo deve corresponder à resposta esperada [2]. Na Tabela 1, apresentam-se, para cada modelo, as médias do tempo total de treinamento e das acuracidades top-1 e top-5 de todas as 5 *folds*, calculadas a partir dos tempos totais de treinamento (10 *epochs* com *frozen initial layers* + 10 *epochs* com *unfrozen layers*) e das acuracidades médias (nas 10 *epochs* com *unfrozen layers*) de cada *fold*, respectivamente.

Tabela 1: ResNets: *Avg Accuracy*, *Avg Top-5 Accuracy* e *Avg Total Time*

<i>Model</i>	<i>Avg Accuracy</i>	<i>Avg Top-5 Accuracy</i>	<i>Avg Total Time (HH:mm:ss)</i>
ResNet-34	0,727425	0,973772	00:14:54
ResNet-50	0,740599	0,971737	00:16:43
ResNet-101	0,757425	0,973892	00:17:08
ResNet-152	0,766108	0,976108	00:16:20

Avaliando-se as acuracidades médias top-1 e top-5, nota-se que à medida em que aumentam as profundidades dos modelos considerados, obtém-se melhores resultados, com exceção feita à acuracidade média top-5 da ResNet-34 que é ligeiramente maior do que aquela obtida pela ResNet-50. Em todos os modelos, dentre as classes que apresentaram maior índice de predições incorretas, observam-se primeiramente algumas samples originalmente rotuladas como Alumínio, mas incorretamente classificadas como Ferro, dadas as semelhanças visuais entre os componentes. Considerando os outros casos de erros, pode-se destacar também algumas predições incorretas entre “Alumínio/Plástico”, “Ferro/Plástico”, “Tubos e Telas / Plástico”.

4 Conclusões

Os resultados mostraram que todos os modelos ResNet testados foram capazes de atingir acuracidades top-1 acima de 70% e acuracidades top-5 acima de 97%, bem como apresentaram dificuldades semelhantes em relação às classes confundidas com maior frequência. Experimentos futuros deverão considerar outras arquiteturas de Redes Neurais Convolucionais, bem como a possibilidade de agrupar determinadas classes, tais como “Alumínio” e “Ferro”.

Referências

- [1] K. He; X. Zhang; S. Ren; J. Sun. *Deep residual learning for image recognition*. Cornell University - arXiv: 1512.03385, 2015.
- [2] J. Howard; S. Gugger. *Deep learning for coders with fastai & PyTorch, 1a. edição*. O’Reilly, Canada. 2020.
- [3] Neurohive. *ResNet (34, 50, 101): Residual CNNs for image classification tasks*. 2019. Disponível em: <https://neurohive.io/en/popular-networks/resnet/>. Acesso em: 09/08/2022.
- [4] Towards Dev. *SOD and U2-Net: an overview*. 2021. Disponível em: <https://towardsdev.com/sod-e-u2-net-an-overview-b0ac8c254e3f>. Acesso em: 09/08/2022.
- [5] J. Wu. *Introduction to convolutional neural networks*. National Key Lab for Novel Software Technology, Nanjing University, p. 1–31, 2017.