

Reconhecimento de componentes eletrônicos com Redes Neurais Convolucionais: banco de dados escasso

Yahgo Sá Amaral¹, Josecley Fialho Góes¹

¹Instituto de Engenharia e Geociências – Universidade Federal do Oeste do Pará (UFOPA) – Santarém – PA – Brasil

yahgostm@hotmail.com, josecley.goes@ufopa.edu.br

Abstract. *The applications of classificatory algorithms, such as neural networks, have increased and need a large active community for the improvement of the technique. For the vast database dependence, the application of Convolutional Neural Networks (CNNs) in industrial field is still not high. In this article, some CNN architectures were used for 5 classes of components, with the use of a scarce database, much smaller than the ideal, to verify how some parameters may interfere with accuracy with few images.*

Resumo. *Aplicações de algoritmos classificadores, como redes neurais, só aumentam e necessitam de uma vasta comunidade para o aperfeiçoamento da técnica. Pela necessidade de um banco de dados amplo, a aplicação de Redes Neurais Convolucionais (CNNs) em âmbitos industriais/técnicos ainda não é alta. Neste trabalho foram testadas algumas arquiteturas de CNNs para 5 classes de componentes eletrônicos, com o uso de um banco de dados escasso, muito menor do que o ideal, para verificar como alguns parâmetros podem interferir na acurácia com poucas imagens.*

1. Introdução

Estudos na área de inteligência artificial para classificação de imagens vêm se intensificando desde o seu princípio. Dentre as técnicas de Deep Learning, as Redes Neurais Convolucionais, ou simplesmente CNNs (Convolutional Neural Networks), são destaques [LECUN et al., 2015], por possuírem os melhores resultados nas aplicações em reconhecimento de imagens.

Redes Neurais Convolucionais são facilmente aplicadas em chips, com isso, algumas empresas como NVIDIA e Intel estão desenvolvendo CNNs para os processamentos das classificações em tempo real em eletrônicos diversos [LECUN. BENGIO; HINTON, 2015]. Apesar do avanço rápido nas aplicações, essas arquiteturas são extremamente dependentes de um banco de dados muito extenso para atingir resultados bons [RUSSAKOVSKY et al, 2015], sendo essa a principal limitação do uso desses algoritmos em aplicações sem disponibilidade de uma quantidade razoável de imagens.

Circuitos Eletrônicos têm uma variedade de componentes com aparência singular. Isso faz com que a aplicação de CNNs para classificações de suas imagens seja algo possível, e, por consequência, trazendo inúmeras possibilidades para a comunicação entre

computer vision e de Componentes Eletrônicos para uso tanto educacional como técnico. Tendo isso em vista, o presente trabalho visa experimentar algumas arquiteturas para verificar como alguns parâmetros num algoritmo de rede neural poderiam ajudar na classificação de imagens com um *database* escasso.

2. Dados

A obtenção dos dados foi realizada através de downloads pelo Google Imagens. Os dados foram imagens de 5 classes de componentes eletrônicos escolhidos por serem os mais comumente utilizados em circuitos, que são: resistores, LEDs, capacitores, transistores e indutores, mostrados na imagem 1.

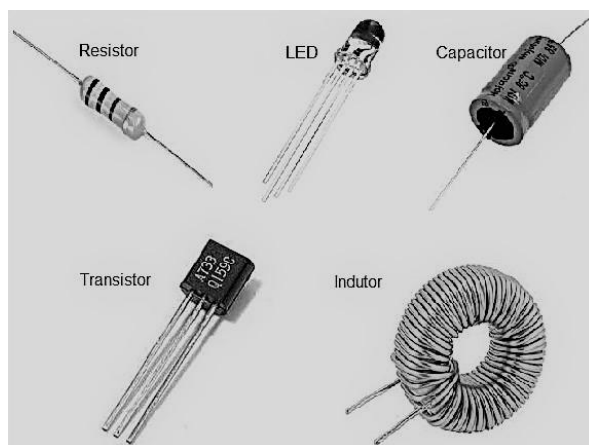


Imagem 1. Componentes utilizados no trabalho

Enquanto as CNNs aplicadas em grandes concursos possuem milhões de imagens para obter-se acurácias elevadas, no caso do trabalho foi possível arrecadar apenas 1757, por limitações tanto de tempo quanto de disponibilidade na internet. A quantidade por tipo de componente foi: Capacitores: 624; Indutores: 397; LEDs: 101; Resistores: 310; e Transistores: 325. Todas coloridas (RGB), com seus tamanhos redimensionados para 150 x 150 pixels.

Os dados, então, foram separados nas suas porções para treino e teste para utilização no algoritmo, usando uma proporção de 80% das imagens para o treino e o restante para o teste, em cada classe de componente.

3. Arquitetura das CNNs e implementação

Foram escolhidas três arquiteturas para os experimentos, baseando-se em um formato similar a de [CHOKSI, V.; RAGHUVANSHI, A., 2016]. Utilizando-se de um pré-processamento para o aumento da quantidade de imagens e com mudança na quantidade nas camadas de convolução e mudanças nas camadas totalmente conectadas, faz-se a tentativa de compensar o banco de dados abaixo do ideal.

Os algoritmos foram implementados em Keras e foram treinados no Laboratório de Modelagem Computacional na Universidade Federal do Oeste do Pará, usando uma GPU, o que possibilitaria o uso de muitas épocas para serem rodadas em pouco tempo.

Porém, verificou-se que com apenas 200 épocas os resultados já chegavam ao seu limite de convergência.

3.1 CNN simples

O primeiro modelo foi feito com objetivo de verificar o comportamento da acurácia com uma arquitetura simples, contendo duas camadas de convolução com 32 filtros cada, seguidas por uma camada de max-pooling, uma camada totalmente conectada (fully connected) com um taxa de dropout alta, para evitar overfitting pelo uso de poucos dados de entrada, e uma camada final de softmax, como mostra a tabela 1.

Tabela 1. Arquitetura da CNN mais simples

Conv - 32
Conv - 32
Max-pool
FC - 64
Dropout
Softmax

3.2 CNN com 5 camadas de convolução

A segunda arquitetura foi aumentando o número de camadas de convolução e sua quantidade de filtros, assim como aumentando o número de neurônios na camada totalmente conectada. O modelo segue a arquitetura mostrada na tabela 2.

Tabela 2. Arquitetura da CNN com 5 camadas convolucionais

Conv - 64
Conv - 64
Max-pool
Dropout
Conv - 64
Conv - 128
Conv - 128
Max-pool
FC- 512
Dropout
Softmax

3.3 CNN mais densa

Agora, aumentando a densidade do sistema, fez-se uma arquitetura com 3 camadas convolucionais e 3 camadas totalmente conectadas. As taxas de dropout foram aplicadas em todas as camadas totalmente conectadas, mas ficam omitidas para facilitar visualização. A arquitetura segue na tabela 3.

Tabela 3. Arquitetura da CNN mais densa

Conv - 32
Conv - 32
Max-pool
Conv - 64
Max-pool
FC - 512
FC - 256
FC- 128
Softmax

4. Resultados e discussões

Foi verificado que as acurácias entre os modelos foram bem parecidas, contrariando a ideia de o sistema mais simples ter sua precisão muito mais baixa. Foram traçados os gráficos para a demonstração das acurácias por época do programa, mostrados a seguir.

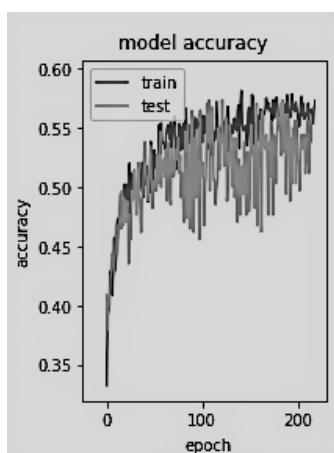


Figura 1: CNN simples

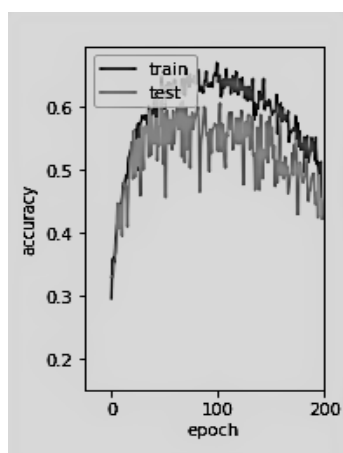


Figura 2: CNN com 5 conv.

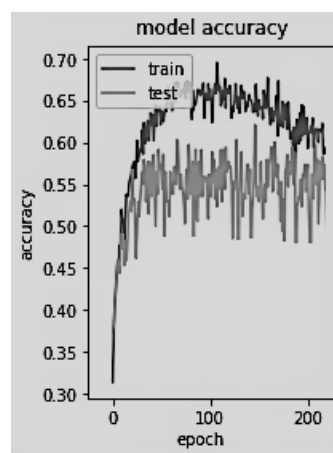


Figura 3: CNN mais densa

Nos gráficos é possível verificar uma acurácia instável para todos os três modelos. A ideia mais plausível é que isso seja consequência do uso de poucas imagens, o que não permite que o sistema tenha uma linha de aprendizado mais estável.

As acurácias de testes variaram entre 50% e 60% na CNN mais densa, enquanto no modelo simples e o modelo com mais camadas de convolução, elas variaram entre 45% e 59%. Valores não ideais, mas já esperados pelo uso de banco de dados escasso.

Mesmo com uso de um dropout elevado (0,5), não foi possível evitar o overfitting na seção de treino das redes. A média delas são mostradas na tabela 4.

Tabela 4. Comparação entre acurácias do modelo

Modelo	Acc de treino	Acc de teste
Simple	0,54	0,50
5conv	0,62	0,53
Densa	0,63	0,55

O modelo mais simples, como esperado, teve sua acurácia mais baixa. Porém sem uma diferença gritante como era esperado. Os modelos com 5 camadas convolucionais e o modelo mais denso apresentarem comportamentos parecidos e suas acurácias muito próximas, sendo o modelo mais denso o com média de acurácia mais elevado, em 0,55.

5. Conclusão

Com este trabalho, foi possível verificar que o principal fator para uma classificação de sucesso usando Redes Neurais Convolucionais é o banco de dados em abundância. Esforços alterando alguns parâmetros para a melhoria dos resultados não causaram consequências notáveis no final. Com uma quantidade de imagens consideravelmente maior, seria possível atingir resultados melhores para possíveis aplicações. A faixa de 0,5 a 0,55 de acurácia não é suficiente para aplicações, pelas suas predições incorretas em praticamente metade das vezes.

Faz-se necessária uma comunidade ativa para arrecadação de mais imagens, para, assim, usufruir melhor de todo o potencial que a área de CNNs podem oferecer ao computer vision.

6. Referências

- LeCun, Y., Bengio, Y., & Hinton, G. (2015). "Deep learning". Nature, 521(7553), 436-444.
- Rawat, W.; Wang, Z. (2017). "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review". The MIT Press Journals. Volume 29, p.2352-2449, Setempro, 2017.
- Russakovsky, J. Deng and L. Fei-Fei. "Imagenet large scale visual recognition challenge". IJCV, 2015.
- F. Chollet. keras. <https://github.com/fchollet/keras>, 2015.
- Raghuvanshi A., Choksi V. 2016. Facial Expression Recognition with Convolutional Neural Networks, Tech Report.