

Utilização de Rede Convolutacional Deep-Learning Pré-treinada Para Reconhecimento de Placas de Trânsito

Luan Ferreira Reis de Jesus, Rânik Guidolini

Resumo — Reconhecimento de placas de trânsito é uma tarefa importante dos veículos inteligentes, este consiste em, dada uma placa de trânsito, reconhecer o seu tipo. Uma rede pré treinada do tipo convolutacional foi usada para o reconhecimento. O emprego de redes neurais profundas (deep-learning) tem se popularizado nos últimos anos devido ao desenvolvimento de técnicas eficientes de treinamento destas redes, o intuito de se empregar deep-learning é de eliminara necessidade de pré-processamentos das imagens, como a aplicação de filtros, e sua desvantagem é que necessita de uma enorme quantidade de dados para a etapa de treinamento. Nossa rede foi re-treinada com mais de 30 mil imagens e foi capaz de alcançar um desempenho de 90% de acerto, o que é um bom resultado se comparado com a literatura. Alguns trabalhos indicam que o reconhecimento de placas de trânsito não será necessário num futuro próximo pois as informações estarão disponíveis para consulta no mapa, entretanto enquanto esta informação ainda não se encontra disponível de forma precisa, o emprego de sistemas de reconhecimento de placas de transito se faz necessário para aumentar a segurança.

I. TRABALHOS RELACIONADOS

Tanto quanto possível pesquisar na literatura da área, o melhor resultado de reconhecimento de placas de transito encontrado ficou em 99,46% de acerto utilizando uma rede deep-learning Multi-Column Deep Neural Network [1]. Uma técnica que se aproxima desse valor Extreme Learning Classifier with Deep Convolutional Features [2] que atinge 99,4% de acerto, com um menor grau de complexidade se comparada com outras técnicas. Um trabalho interessante que desconsidera a informação de cor da imagem [3] e consegue uma taxa de acerto de 98%, se mostrou bastante interessante, pois simplifica o trabalho com a imagem e reduz o consumo de memória. Dados os diversos métodos computacionais propostos para resolver o problema de reconhecimento de placas de transito é interessante compara-los com a capacidade de motoristas humanos de reconhece-los como proposto por [4], em que segundo a pesquisa realizada a média de acertos ficou em 98,84%.

II. METODOLOGIA

O treinamento de redes neurais profundas é computacionalmente caro e dura um tempo considerável; dependendo do tamanho da rede podem ser necessárias várias semanas para realizar o treinamento. Como forma de reduzir o esforço necessário para o treinamento, surge a ideia de se utilizar um modelo pré-treinado de rede neural. Neste caso, um modelo de rede neural treinado para algum problema tem os pesos ajustados para se adequar a um novo problema, um processo conhecido como fine-tuning (sintonia fina). No

início do processo de treinamento, a rede recebe os pesos do modelo pré-treinado, ao invés de pesos aleatórios como geralmente ocorre no treinamento de redes neurais. Isso permite que o número de iterações necessários para o treinamento seja reduzido.

Para a realização deste trabalho foi utilizado o caffè [6], um framework elaborado especialmente para facilitar o desenvolvimento de rede neurais profundas. Os caffè possui uma vasta gama de modelos pré-treinados e a implementação permite execução rápida otimizada para GPUs.

O modelo pre-treinado escolhido foi o da rede NIN-Imagenet (Network in Network Imagenet Model) [7] que é o modelo de uma rede de 4 camadas treinado com a base de dados imagenet. Esse modelo obteve 59.36% de acurácia no conjunto de validação da base imagenet. O tempo de treinamento desse modelo também é bastante reduzido quando comparado com outros modelos. A utilização de outros modelos de redes pré-treinadas foi pensada, porém, devido a limitações de hardware para realizar o treinamento, não obtiveram sucesso.

A base de dados utilizadas foi a disponibilizada pelo The German Traffic Sign Benchmark (GTSRB) [8]. Nessa base de dados as placas estão divididas em 43 classes, como pode ser visto na Figura 1. São disponibilizadas 39209 imagens para treinamento e 12630 para teste. Durante o treinamento, do total das 39209 imagens de treinamento, 90% foram utilizadas para de fato treinar a rede e 10% foram utilizadas para validação. O conjunto de teste ficou separado para ser usado apenas no teste final.

A rede pré-treinada escolhida foi adaptada. Para o problema, a última camada, que possuía 1000 saídas, foi eliminada e, foi adicionada uma nova camada com 43 saídas, correspondendo ao número de classes de placas. Como o treinamento começa a partir dos pesos do modelo pré-treinado, a taxa de aprendizagem pode ser menor do que no caso de um treinamento ‘do zero’.

III. EXPERIMENTOS

Antes do treinamento ser de fato iniciado algumas medidas foram tomadas, as imagens foram separadas em conjuntos de treino e validação e redimensionadas para 256x256px para evitar o redimensionamento durante o treinamento, o que poderia aumentar o tempo. Scripts e códigos auxiliares foram criados para auxiliar nesse processo.

Com as bases de dados já preparadas o treinamento pode começar. O treinamento foi feito com o auxílio de uma GPU Tesla C2050, porém, como dito anteriormente durante primeiros testes efetuados, com modelos de rede maiores, o treinamento falhava devido a limitações técnicas. Com uma

rede menor (NIN-Imagenet) o treinamento pode então ser realizado.

O treinamento foi encerrado quando a acurácia na validação começou a oscilar em cerca de 90% (sem variar muito abaixo ou acima desse patamar). O número de iterações foi de 40000, após cerca de 23 horas de treinamento, e a acurácia na validação foi de 90,36%.



Figura 1. Exemplos de placas das 43 classes da base de dados GTSRB.

IV. RESULTADOS E DISCUSSÕES

Os resultados alcançados são considerados bons para o tipo de rede utilizada, com mais tempo de treinamento a acurácia tende a aumentar, porém a rede tende a chegar se sobre-ajustar aos dados de validação (Overfitting) e perder capacidade de generalização e, diminuindo a acurácia no teste final. Outros parâmetros da rede e do treinamento poderiam ser modificados, e talvez outros modelos pré-treinados. De fato, a utilização de um modelo pré-treinado mostrou resultado e funcionou como esperado, demonstrando que as redes neurais profundas podem ser utilizadas para solucionar diversos problemas e ajudar a entender melhor o funcionamento do cérebro humano.

Os códigos utilizados, o modelo treinado para o reconhecimento de placas e mais instruções de como utilizá-lo podem ser encontrados no endereço:

https://github.com/luanfrj/trab_cogvis.

BIBLIOGRAFIA

- [1] Cireşan, D., Meier, U., Masci, J., & Schmidhuber, J. (2012). Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32, 333-338.
- [2] Zeng Y, Xu X, Fang Y, Zhao K. Traffic sign recognition using extreme learning classifier with deep convolutional features. In *The 2015 international conference on intelligence science and big data engineering (ISIDE 2015)*, Suzhou, China 2015 Jun 14.
- [3] Fleyeh H. Traffic sign recognition without color information. In *Colour and Visual Computing Symposium (CVCS)*, 2015 2015 Aug 25 (pp. 1-6). IEEE.
- [4] Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. (2012). Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32, 323-332. H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.
- [5] Jia, Yangqing, et al. "Caffe: Convolutional architecture for fast feature embedding." *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014.
- [6] Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." *arXiv preprint arXiv:1312.4400* (2013).
- [7] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, pages 1453-1460. 2011.
- [8] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.