

# ImageNet Classification with Deep Convolutional Neural Networks

Jairo Lucas Universidade Federal do Espírito Santo - Laboratório de Computação de Alto Desempenho - LCAD

**Resumo** - Neste trabalho replicamos a arquitetura da rede neural profunda criada por Alex Krizhevsk, Ilya Sutskever e Geoffrey Hinton que se consagrou vencedora do concurso ILSVRC[3] de 2012, utilizando imagens locais.

Os resultados conseguidos por esta arquitetura foram impressionantes, sendo a mesma considerada um divisor de águas em relação as técnicas empregadas até então para problemas de classificação de imagens, demonstrando que que as redes CNN's seriam o novo padrão para problemas de classificação de imagens.

A arquitetura apresentada é composta por 8 camadas sendo 5 camadas convolucionais e 3 completamente conectadas, utilizando SoftMax. Os resultados obtidos no concurso alavancaram o estado da arte para a tarefa de classificação de imagens, conseguindo uma taxa de erro de 15.3%, muito distante do 2º. colocado, que ficou com uma taxa de 26.2%.

Em nossos experimentos utilizamos imagens locais, o Framework Caffe[2], um aplicativo em C para classificar a imagem e um modelo pré-treinado exatamente com os mesmos parâmetros da rede apresentada no ILSVRC-2012[3]. Conseguimos uma taxa de erro em torno de 14%, bastante semelhante a reportada pela literatura para a arquitetura utilizada.

**Termos** - Redes Neurais Convolucionaiss, CNN, Deep Learning, ILSVRC, Caffe, Modelos pré-treinados.

## 1.INTRODUÇÃO

As abordagens atuais para o reconhecimento de objetos fazem uso essencialmente dos métodos de aprendizagem de máquinas, sendo que estes métodos exigem um volume extremamente grande de imagens previamente rotuladas para a fase de treinamento da rede. Até recentemente, os conjuntos de dados de imagens rotuladas eram relativamente pequeno (da ordem de milhares de imagens). Este quantitativo é adequado para as tarefas de reconhecimento simples, como por exemplo, a tarefa de reconhecimento de algarismos numéricos. A taxa de erro para a base MNIST (base composta por imagens de números de 0 a

9 manuscritos) é de aproximadamente 0,3%, oque se aproximada do desempenho humano. Porém, objetos em cenários realistas exibem variabilidade considerável, sendo necessário usar conjuntos de treinamento muito maiores. Recentemente este problema foi parcialmente resolvido com a criação de conjuntos de dados rotulados com milhões de imagens, como o IMAGENet [4], que consiste de mais de 15 milhões de imagens de alta resolução rotulados em mais de 22.000 categorias.

No entanto, dada a variabilidade dos objetos no mundo real, mesmo utilizando conjunto gigantesco de dados para treinamento, precisamos de uma rede que seja capaz de aprender características das imagens, de forma que a mesma possa abstrair variações de rotação, tamanho, e iluminação.

As redes neurais convolutivas atendem esta necessidade.

### 1.1 – Redes CNN – Redes Neurais Convolucionais

As CNN's – Redes Neurais Convolucionais - foram projetadas inspiradas na arquitetura biológica do cérebro. Em 1968 Hubel e Wiesel realizaram experimentos com gatos e macacos e mostraram que o córtex visual é formado por um conjunto hierárquico de células sensíveis a pequenas sub-regiões chamadas de campos receptivos, de forma que cada célula é “especialista” em monitorar (e ser ativada) por uma pequena região. Hubel classificava estas células em categorias - simples, complexas e supercomplexas – de acordo com o padrão de estímulo que as ativam. Células simples são ativadas quando são apresentados padrões simples para o animal, como linhas. As células complexas e supercomplexas são ativadas quando padrões mais elaborados são apresentados ao animal.

A partir deste estudo surge a hipótese que uma boa representação interna para uma rede neural para reconhecimento de imagens seria uma estrutura hierárquica, onde os pixels formam arestas, as arestas formam padrões, os padrões combinados formam as partes, as partes combinadas formam os objetos e os objetos formam a cena [5].

Esta estrutura considera que o mecanismo de reconhecimento necessita de vários estágios de treinamento empilhados uns sobre os outros, um para cada nível de hierarquia [5]. As redes CNN's seguem este conceito, representando arquiteturas multi-estágios capazes de serem treinadas.

## 2. ILSVRC – IMAGENET LARGE SCALE VISUAL RECOGNITION CHALLENGE

A competição ILSVRC consiste em 3 desafios, onde as equipes podem competir em um ou mais deles:

- **Classificação:** Dado uma imagem, o algoritmo deve produzir uma lista ordenada com 5 rótulos de classes.
  - Caso o rótulo verdadeiro da imagem não esteja na lista o erro é igual a 1, caso contrário 0.
  - A pontuação geral é dado pelo erro médio.
- **Localização (Single-object localization):** Dado uma imagem, o algoritmo deve produzir uma lista dos objetos presentes na cena e demarcar os mesmos.
  - Caso um objeto encontrado na cena combinar com o rótulo e a demarcação do objeto estiver de acordo com as demarcação fornecida o erro é igual a 0, caso contrário 1.
  - A caixa delimitadora encontrada deve ter uma intersecção mínima de 50% com a caixa delimitadora fornecida.
  - A imagem de treino é fornecida com o rótulo verdadeiro e com as coordenadas da caixa delimitadora de todas as instâncias do objeto rotulado.
- **Detecção (Object detection):** Dado uma imagem, o algoritmo deve localizar e demarcar todas as instâncias de todos os objetos localizados na imagem.
  - A qualidade da detecção é avaliada considerando o *Recall*, *Precision* e *Average Precision*

A arquitetura descrita neste paper concorreu somente no desafio de **classificação**.

### 2.1 – A Base de Dados ImageNet

ImageNet é um conjunto de dados de mais de 15 milhões de imagens de alta resolução rotuladas em cerca de 22.000 categorias. As imagens foram coletadas da web e rotuladas por pessoas usando a ferramenta de crowd-sourcing da Amazon Turk.

ILSVRC usa um subconjunto do ImageNet com aproximadamente 1000 imagens em cada uma das 1000 categorias. Ao todo, existem cerca de 1,2 milhão de imagens de treino, 50.000 imagens de validação e 150.000 imagens de teste.

## 3. A ARQUITETURA DA REDE

A arquitetura da rede é composta por oito camadas de aprendizado, sendo cinco convolucionais e três totalmente conectadas. A figura 1 mostra a estrutura da rede.

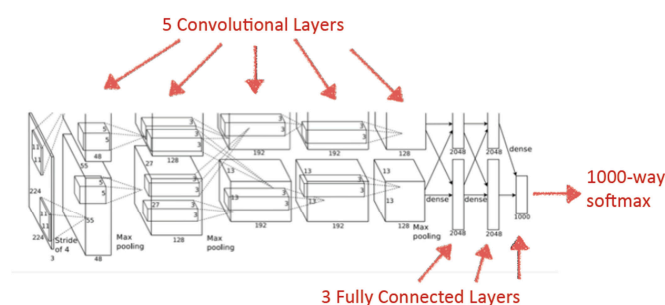


Fig 1 – Estrutura da rede

### 3.1 – ReLU Nonlinearity

Outra inovação apresentada pelo trabalho é o uso ReLu como função de ativação. A maneira padrão de modelar a saída de um neurônio  $f$  como uma função de sua entrada  $x$  é dada por  $f(x) = \tanh(x)$  ou  $f(x) = (1 + e^{-x})^{-1}$ . Dado a complexidade matemática, estas funções são exigem muito poder computacional.

A função de ativação não linear ReLu (Unidades Lineares Retificadas) utiliza a função  $f(x) = \text{Max}(0; x)$ , que é várias vezes mais rápida que as citadas anteriormente. A figura 2 mostra a comparação entre o número de iterações necessárias utilizando ReLU e uma função sigmoide e para alcançar um erro de treinamento de 25% no conjunto de dados CIFAR para uma rede de 4 camadas.

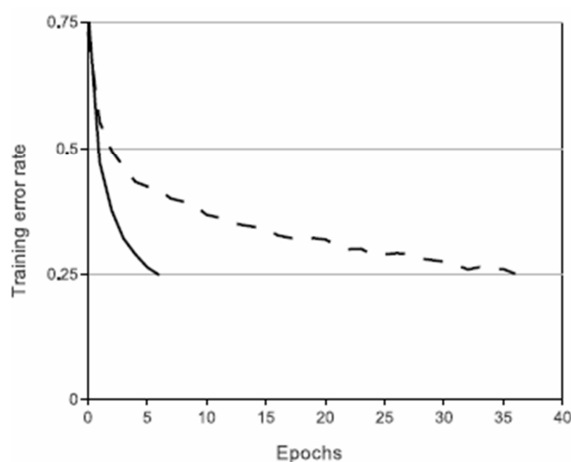


Figure 2: A four-layer convolutional neural network with ReLUs (solid line) reaches a 25% training error rate on CIFAR-10 six times faster than an equivalent network with tanh neurons (dashed line). The learning rates for each network were chosen independently to make trainings fast as possible. No regularization of any kind was employed. The magnitude of the effect demonstrated here varies with network architecture, but networks with ReLUs consistently learn several times faster than equivalents with saturating neurons.

### 3.2 – Treinamento em múltiplas GPUS

As GPU's atuais são particularmente adequadas para a paralelização de GPU cruzadas, pois elas podem ler e gravar diretamente das memórias das GPU's sem a necessidade de passar pela memória principal do host. Considerando isso, a rede foi treinada espalhando-se a mesma em duas GPU's cruzadas. O esquema empregado coloca metade dos Kernels em cada GPU, sendo que as GPU's se comunicam somente em determinadas camadas. Por exemplo, a camada três recebe como entrada todos os mapas do kernel da camada 2, de ambas as GPU's. Já a camada quatro, recebe como entrada somente os mapas do kernel da camada três residente na própria GPU.

### 3.3 Overlapping Pooling

As camadas de agrupamento nas CNN resumem as saídas dos grupos vizinhos de neurônios no mesmo mapa do kernel. Tradicionalmente, os grupos resumidos por unidades de pooling adjacentes não se sobrepõem. Para ser mais preciso, uma camada de pooling pode ser pensada como consistindo em um grid de unidades de pooling separados com  $s$  pixels, cada uma resumindo vizinhos de tamanho  $Z \times Z$  centrado no local da unidade de pooling. Se definirmos  $s = z$ , obtemos agrupamentos

locais tradicionais como comumente empregados nas CNNs. Se configurarmos  $s < z$ , obtemos agrupamentos sobrepostos. Este padrão de sobreposição foi utilizado em toda a rede, com  $s = 2$  e  $z = 3$ . Este esquema reduz as taxas de erro top-5 em 0,3% em comparação com o esquema não sobreposto ( $s = 2$ ;  $z = 2$ ).

### 3.4 Redução de Overfitting

A arquitetura da rede proposta utilizou dois métodos para redução do overfitting, que são descritos a seguir.

#### 3.4.1 Aumento Artificial dos Dados

O método mais fácil e mais comum para reduzir a overfitting nos dados da imagem é aumentar artificialmente o conjunto de dados usando transformações e preservando o rótulo da imagem. Na rede em questão foram empregadas duas formas distintas de aumento de dados, que permitem que imagens transformadas sejam produzidas a partir das imagens originais com muito pouca computação.

- A primeira forma de aumento de dados consiste em gerar translações e reflexões horizontais da imagem. Para isso, extrai-se “pedaços” aleatórios de  $224 \times 224$  (e suas reflexões horizontais) das imagens e treina-se a rede com os “pedaços” extraídos. No momento do teste, a rede faz uma previsão extraindo cinco “pedaços”  $224 \times 224$  (quatro pedaços dos cantos e o patch central), bem como as suas reflexões horizontais (dez pedaços no total). O resultado é a média das previsões feitas pela camada softmax da rede nos dez pedaços.
- A segunda forma de aumento de dados consiste em alterar as intensidades dos canais RGB nas imagens de treinamento. É executado o PCA no conjunto de valores de pixels RGB em todas as imagens de treinamento da ImageNet. Para cada imagem de treinamento, são adicionados múltiplos dos principais componentes encontrados com magnitudes proporcionais aos autovalores correspondentes vezes uma variável aleatória adquirida de uma gaussiana com média zero e desvio padrão 0,1. Este esquema captura uma propriedade importante das imagens naturais, que é o fato da identidade do objeto ser invariante para mudanças na intensidade e na cor da iluminação. Esse esquema reduz a taxa de erro top 1 em mais de 1%.

### 3.4.3 DropOut

A arquitetura proposta utiliza a técnica chamada DropOut (abandono), que consiste em ajustar para zero a saída de uma determinada quantidade de neurônios.

Os neurônios que são Dropped Out (abandonados) não contribuem para a passagem para frente (Forward) e não participam da backpropagation. Desta forma, cada vez que uma entrada é apresentada, a rede neural experimenta uma arquitetura diferente, mas todas essas arquiteturas compartilham pesos.

Esta técnica reduz co-adaptações complexas de neurônios, uma vez que um neurônio não pode confiar na presença de outros neurônios em particular sendo forçado a aprender recursos mais robustos que são úteis em conjunto com muitos subconjuntos aleatórios diferentes dos outros neurônios. O DropOut foi utilizado nas duas primeiras camadas completamente conectadas.

### 3.5. Arquitetura geral da rede

Conforme ilustrado na figura 2, a arquitetura geral da rede implementada conta com as seguintes características:

- A rede contém oito camadas com pesos; As cinco primeiras são convolucionais e as três restantes estão totalmente interligadas. A saída da última camada totalmente conectada é alimentada para uma softmax 1000-way que produz uma distribuição sobre os rótulos de 1000 classes.
- Os núcleos das segunda, quarta e quinta e camadas convolucionais estão conectados apenas aos mapas do kernel na camada anterior que residem na mesma GPU (veja a Figura 2). O kernel da terceira camada convolucional esta conectado a todos os kernel do mapa da segunda camada. Os neurônios nas camadas totalmente conectadas estão conectados a todos os neurônios na camada anterior.
- ReLU é aplicada à saída de todas as camadas convolucionais e das totalmente conectadas.
- A primeira camada convolucional utiliza um filtro de  $224 \times 224 \times 3$  para imagens de entrada com 96 kernels de tamanho  $11 \times 11 \times 3$  com um stride de 4 pixels.
- A segunda camada convolucional tem como entrada a saída (resposta-normalizada e agrupada) da primeira camada convolucional e 256 kernels de tamanho  $5 \times 5 \times 48$ .

- A terceira, quarta e quinta camadas estão conectadas entre si sem quaisquer camadas intermediárias ou de normalização.
- A terceira camada convolucional possui 384 kernels de tamanho  $3 \times 3 \times 256$  conectadas a saída (normalizada e resumida) da segunda camada.
- A quarta camada possui 384 kernels de tamanho  $3 \times 3 \times 192$
- A quinta camada possui 256 kernels de tamanho  $3 \times 3 \times 192$ .
- As camadas totalmente conectadas possuem 4096 neurônios cada.

## 4. Experimentos e Resultados

Foram efetuados testes para reproduzir o resultado da rede descrita no paper utilizando imagens locais, capturadas através de um celular e sem nenhum tipo de tratamento especial, apenas o redimensionamento da imagem para o tamanho esperado pela rede.

Para o experimento utilizamos o Framework Caffe [2], a rede pré-treinada *bvlc\_alexnet.caffemodel* que possui exatamente a mesma arquitetura e foi treinado exatamente com os mesmos parâmetros descritos no paper.

Como o Framework Caffe não permite efetuar o teste de uma imagem individual diretamente pela linha de comando, utilizamos uma adaptação de um programa em C fornecido como exemplo pela própria Caffe para esta tarefa. O programa utilizado é o *classification.c*

### 4.1 – Base de dados e Resultados

Para os testes efetuados utilizamos 28 imagens de objetos do cotidiano adquiridas por um celular. A figura 3 mostra algumas dessas imagens:

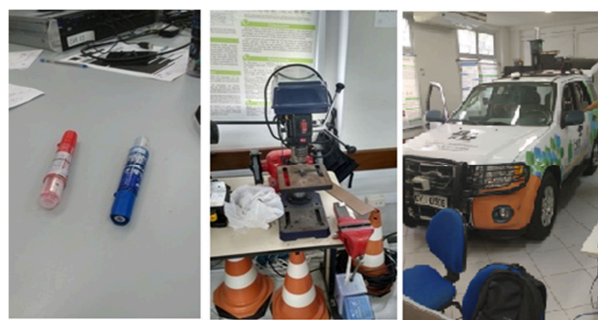


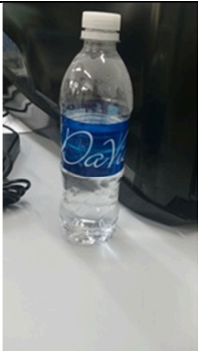







Fig. 3 – Exemplo de imagens de testes

Das 28 imagens apresentadas a rede conseguiu reconhecer corretamente 24 imagens, apresentando um erro de classificação em torno de 14%.

O resultado conseguido está de acordo com o reportado na literatura para a rede em questão. No concurso ILSVR2012, onde a mesma foi apresentada, o resultado reportado foi um erro de 15.3%. A figura 3 mostra alguns exemplos de classificação.

	
studio couch, day bed quilt, comforter, puff cradle computer keyboard, laptop, laptop computer"	desktop computer printer photocopier screen, noteboor
	
water bottle cocktail shaker perfume, essence nipple pop bottle, soda bottle	folding chair steel drum solar dish, furnace rocking chair, rocker

	
lipstick, lip rouge plunger, plumber's helper bucket, pail soap dispenser toilet seat	cab, hack, taxi, taxicab desk tow truck, tow car, race car, racing car garbage truck, dustcart
	
Band Aid rubber eraser, pencil digital clock envelope harmonica, harp	cassette switch, electric switch, syringe modem rubber eraser, pencil eraser

## 5. CONCLUSÕES

Este relatório replicou e testou o trabalho de Alex e Hinton descritos no paper '*ImageNet Classification with Deep Convolutional Neural Networks*', utilizando uma rede pré-treinada com a mesma arquitetura e com os mesmos parâmetros descritos no paper. A replicação foi testada com uma base formada por imagens de objetos locais.

Os resultados conseguidos demonstram que a rede consegue alcançar a performance descrita no paper, com uma margem de erro de aproximadamente 14%.

Pela análise das imagens é possível inferir que apesar da boa performance, a rede ainda erra bastante, muitas vezes em objetos simples do ponto de vista humano, como a placa informativa na mostrada na figura 4. É provável que este tipo de erro seja em função das poucas amostras de objetos desta classe apresentadas para a rede na fase de treinamento.

Na época em que foi apresentada, em 2012, a rede testada elevou drasticamente o estado da arte para a tarefa de reconhecimento de objetos. Porém, nos últimos anos a mesma foi superada por outras redes neurais convolucionais, como a rede proposta por Hu [7] que apresenta uma margem de erro de menos de 3% para a base ILSVRC.

## 6. BIBLIOGRAFIA

- [1] Alex Krizhevsky Ilya Sutskever e Geoffrey E. Hinton , **ImageNet Classification with Deep Convolutional Neural Networks** - NIPS'12 - Conference on Neural Information Processing Systems - 2012
- [2] Jia, Yangqing and Shelhamer, Evan and Donahue, Jeff and Karayev, Sergey and Long, Jonathan and Girshick, Ross and Guadarrama, Sergio and Darrell, Trevor; *Caffe: Convolutional Architecture for Fast Feature Embedding*; arXiv preprint arXiv:1408.5093; 2014
- [3] A. Berg, J. Deng, and L. Fei-Fei. *Large scale visual recognition challenge* 2010. [www.image-net.org/challenges](http://www.image-net.org/challenges). 2012.
- [4] Olga RussakovskyEmail author, Jia Deng ,Hao Su and at – **ImageNet Large Scale Visual Recognition Challenge** - December 2015, Volume 115, Issue 3, pp 211–252 |
- [5] LECUN, Yann et al. Convolutional networks and applications invision. In: ISCAS. 2010. p. 253-256.
- [6] A. Krizhevsky. Convolutional deep belief networks on cifar-10. Unpublished manuscript, 2010.
- [7] Jie Hu, Li Shen, Gang Sun - *Squeeze-and-Excitation Networks* - <https://arxiv.org/abs/1709.01507v1> – September 2017