(4-b) Arquiteturas de CNNs

Redes Neurais e Aprendizado Profundo

Moacir Ponti

www.icmc.usp.br/~moacir — moacir@icmc.usp.br

Agenda

Sequenciais e em blocos

Arquitetura com ramos

Arquitetura residual

Convoluções separáveis em profundidade

Agenda

Sequenciais e em blocos

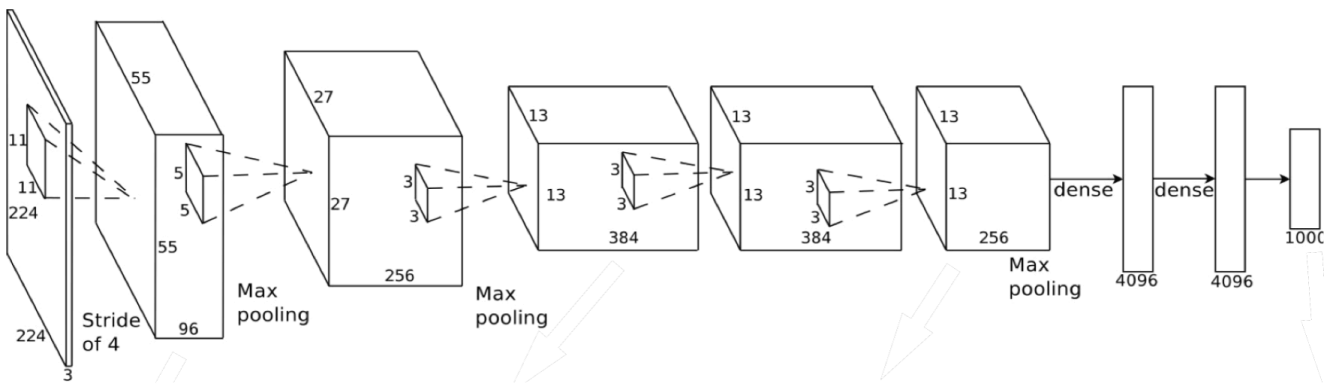
Arquitetura com ramos

Arquitetura residual

Convoluções separáveis em profundidade

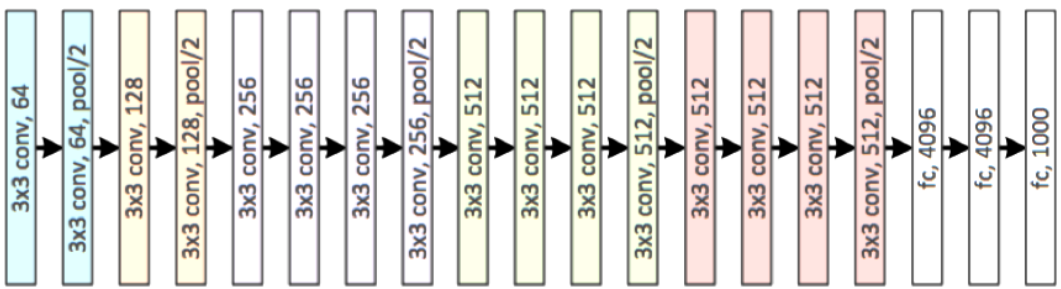
AlexNet (Krizhevsky, 2012)

- entrada 224×224
- conv1: $K = 96$ filters with $11 \times 11 \times 3$, stride 4,
- conv2: $K = 256$ filters with $5 \times 5 \times 96$,
- conv3: $K = 384$ filters with $3 \times 3 \times 256$,
- conv4: $K = 384$ filters with $3 \times 3 \times 384$,
- conv5: $K = 256$ filters with $3 \times 3 \times 384$,
- densas1, 2: $K = 4096$.



VGGNet (Simonyan, 2014)

- entrada 224×224,
- filtros: todos 3×3,
- conv 1-2: $K = 64$ + maxpool
- conv 3-4: $K = 128$ + maxpool
- conv 5-6-7-8: $K = 256$ + maxpool
- conv 9-10-11-12: $K = 512$ + maxpool
- conv 13-14-15-16: $K = 512$ + maxpool
- densas1, 2: $K = 4096$



Agenda

Sequenciais e em blocos

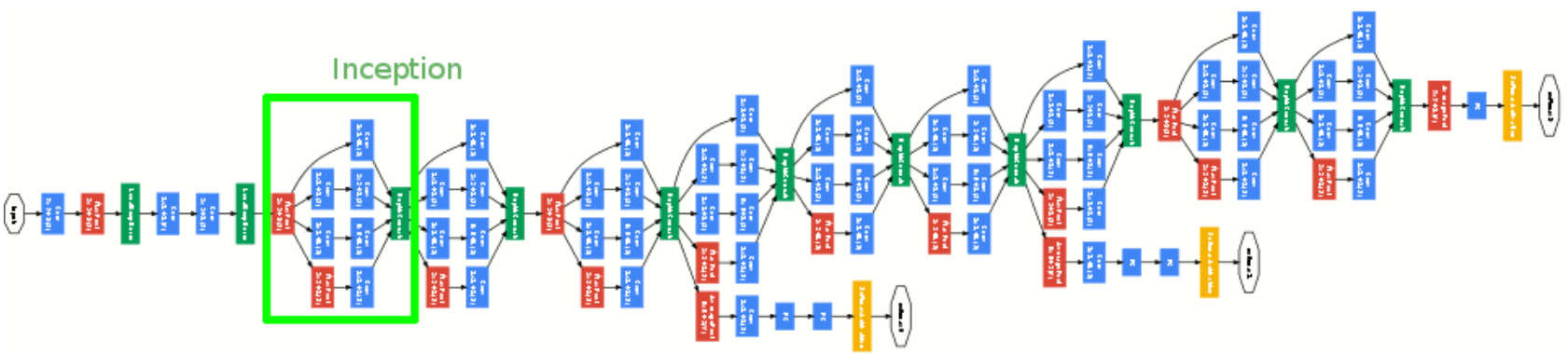
Arquitetura com ramos

Arquitetura residual

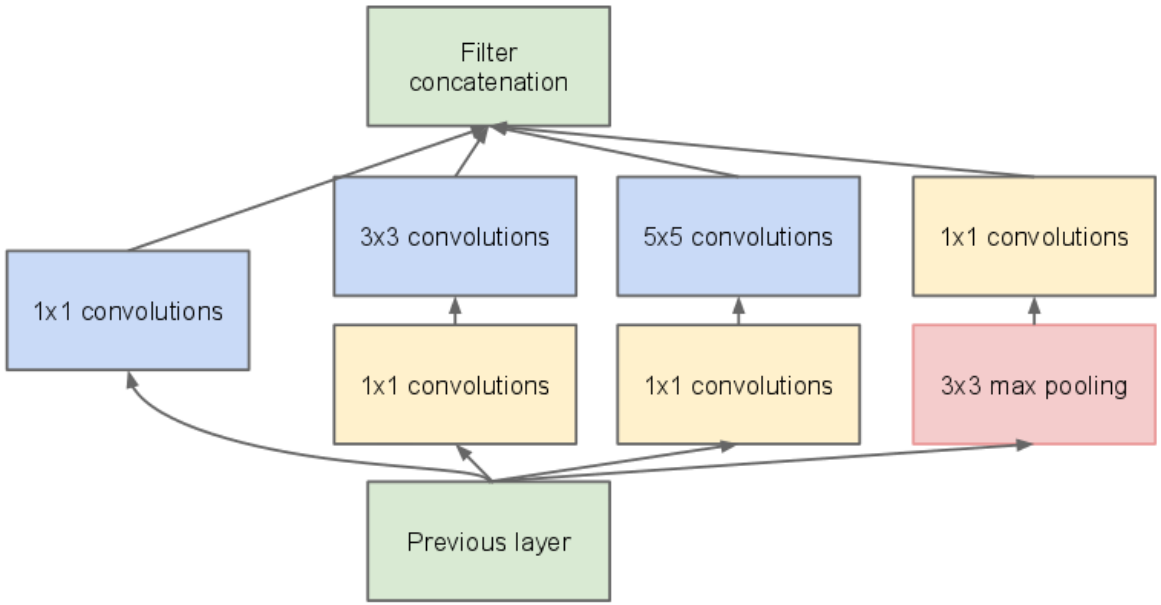
Convoluções separáveis em profundidade

GoogLeNet / Inception (Szegedy, 2014)

- 22 layers (v1)
- Começa com duas camadas convolucionais
- *Inception layer* (banco de filtros):
  - filtros 1×1, 3×3, 5×5 + max pooling 3×3;
  - controla dimensionalidade usando filtros 1×1.
  - 3 classificadores (não sequenciais)
- Azul = conv.,
- Vermelho = pool.,
- Amarelo = densa+softmax,
- Verde = concatenação.

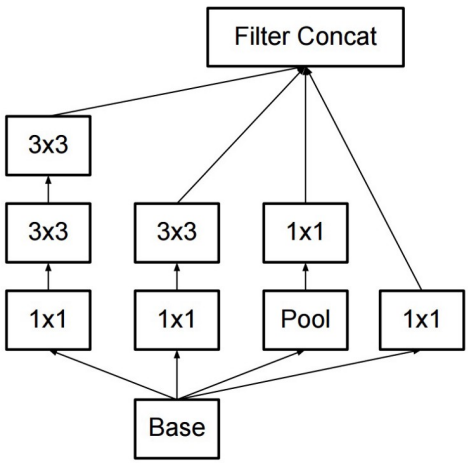


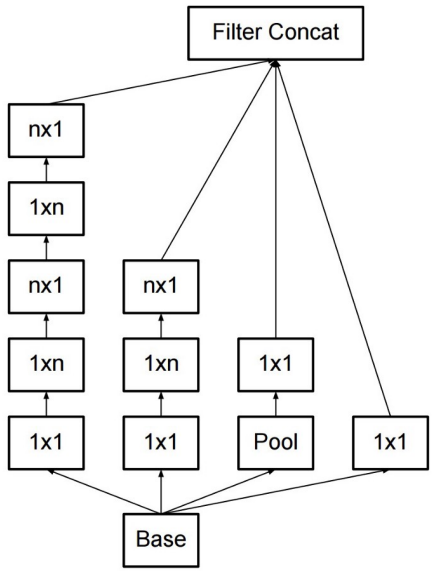
GoogLeNet: módulo inception v1

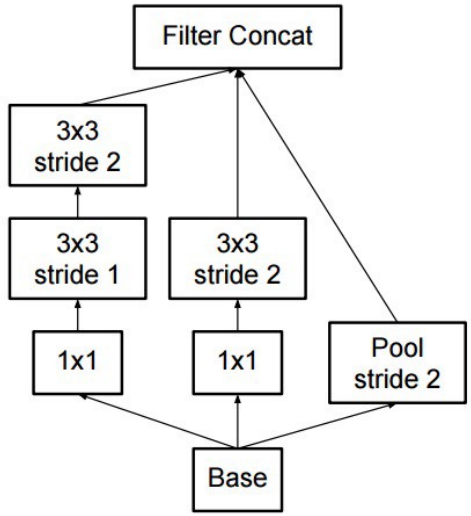


- filtro 1×1 reduz a profundidade da entrada
- concatena mapas de ativação de 3 filtros + maxpooling

Módulos inception (V2 and V3)







Agenda

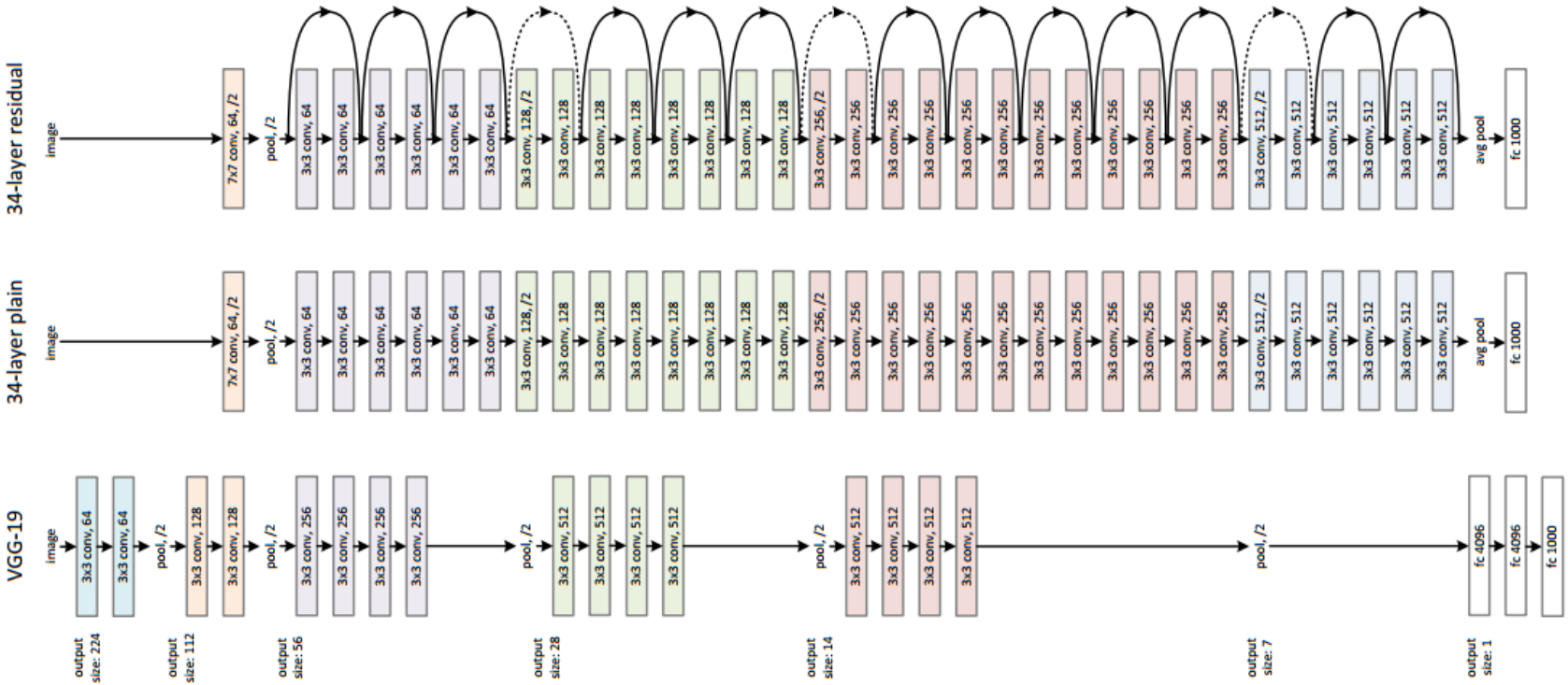
Sequenciais e em blocos

Arquitetura com ramos

Arquitetura residual

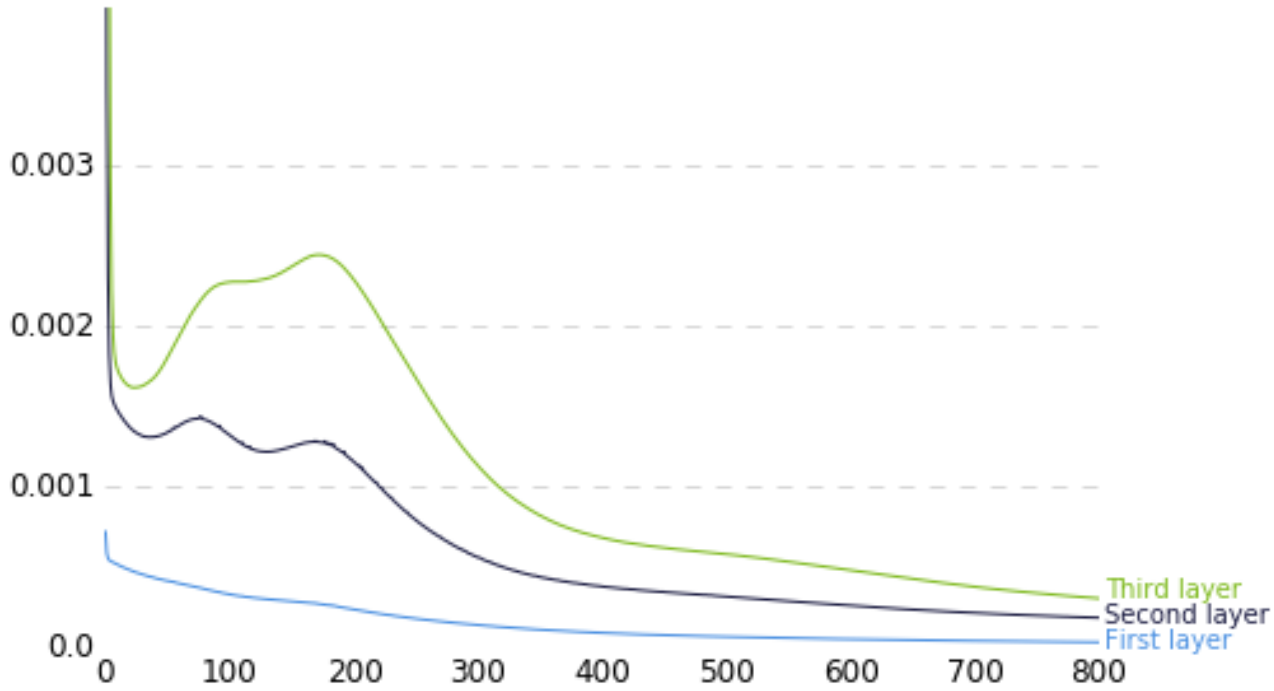
Convoluções separáveis em profundidade

Residual Network (He et al, 2015)



O gradiente não se comporta igual em todas as camadas

Gradiente medido em cada camada

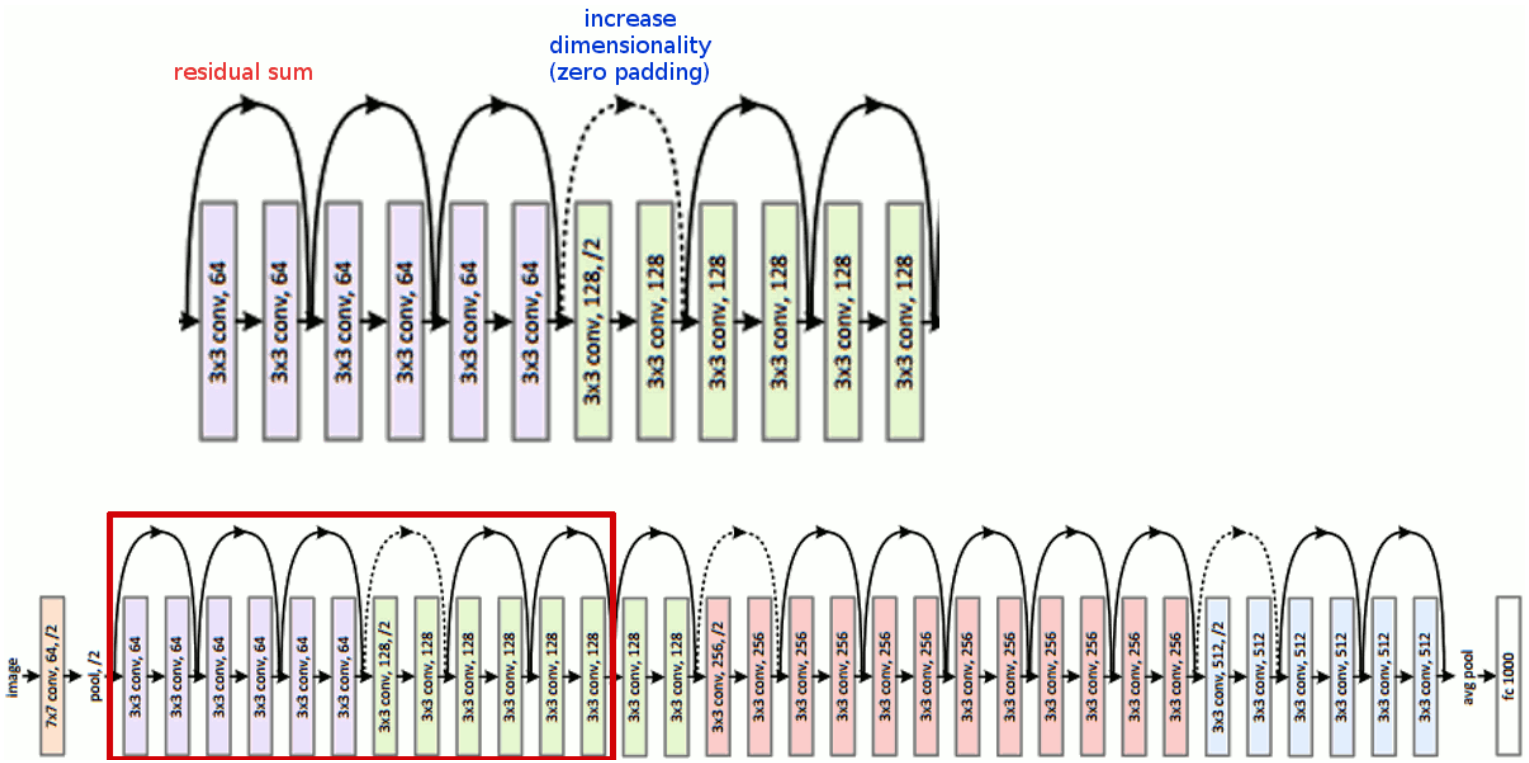


Agradecimentos a Harini suresh (`http://harinisuresh.com`) pelos gráficos

Residual Network — ResNet (He et al, 2015)

Pular camadas (skip layers) permite empilhar mais camadas (de 34 a $\sim$ 1000).

Arquitetura residual: adiciona resultado anterior preservando gradiente.



Agenda

Sequenciais e em blocos

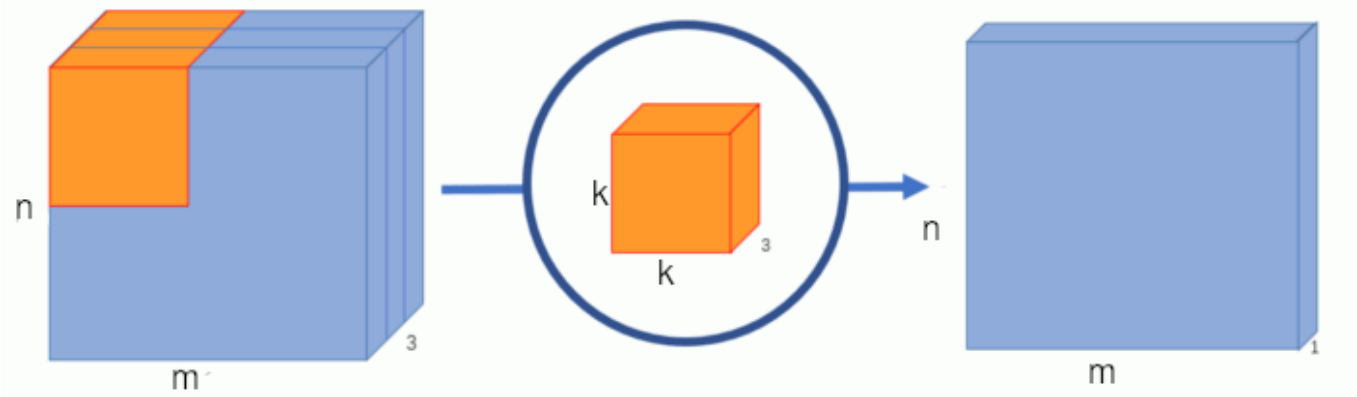
Arquitetura com ramos

Arquitetura residual

Convoluções separáveis em profundidade

Depthwise separable convolutions - Xception, Mobilenet

Convolução convencional

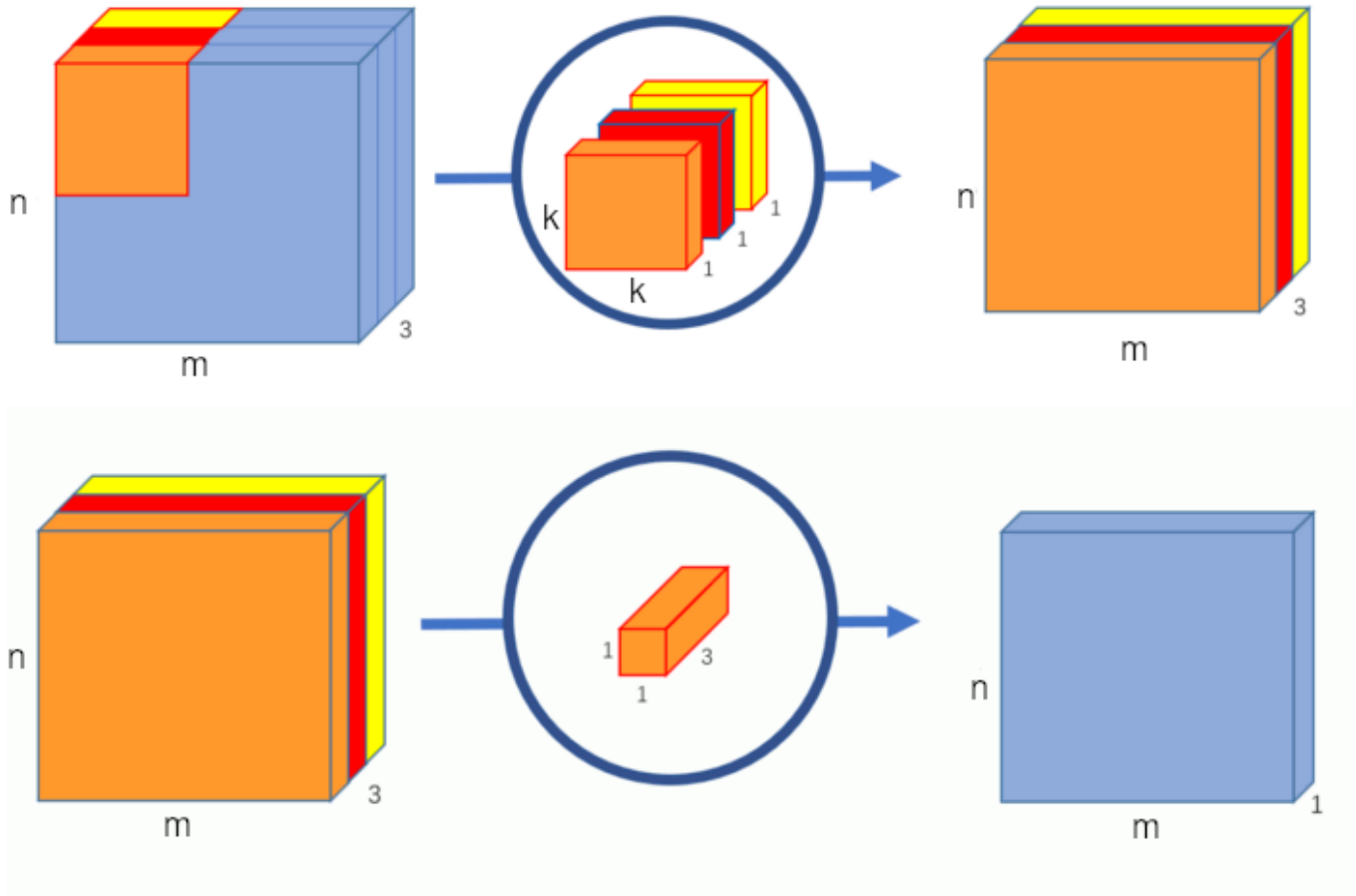


Para obter 128 mapas de características com filtros 3×3 temos:

- $3 \times 3 \times 3 \times 128 = 1638$ parâmetros.
- em imagem com $100 \times 100 \times 3$ pixels, movemos cada filtro 10000 vezes para multiplicar os valores locais, total em multiplicações: $10000 \times 1638 = 16.380.000$

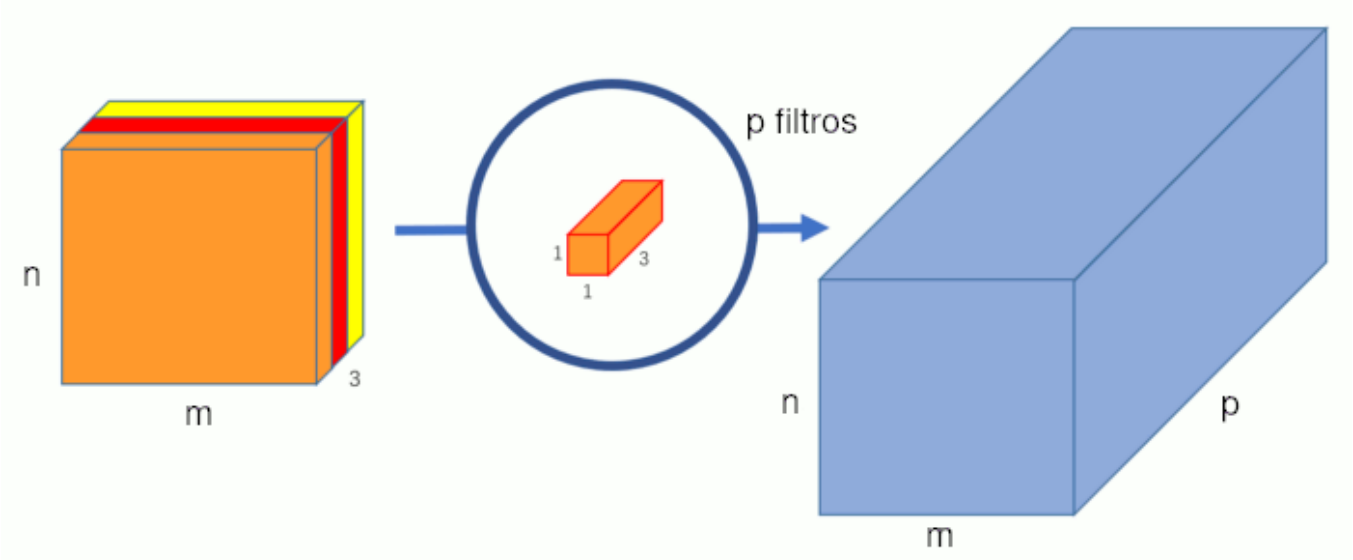
Depthwise separable convolutions - Xception, Mobilenet

Separável: primeiro lateral, depois em profundidade

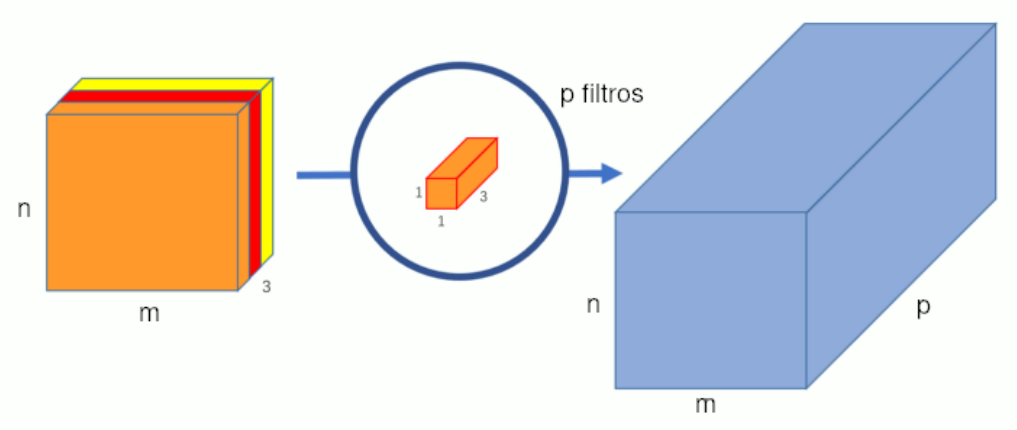


Depthwise separable convolutions - Xception, Mobilenet

Lateral executada uma única vez, produzindo um único volume, depois p filtros $1 \times 1 \times 3$ produzirão p mapas de ativação



Depthwise separable convolutions - Xception, Mobilenet

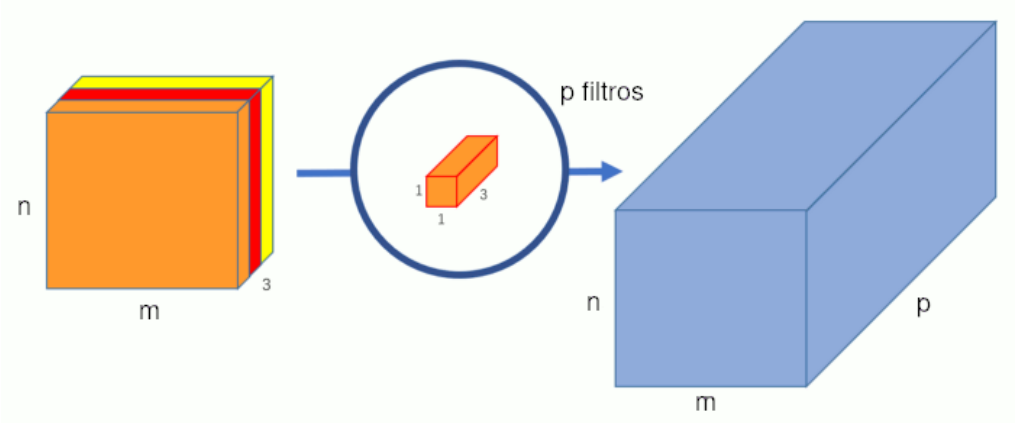


Para obter 128 mapas de características com filtros 3×3 temos:

- lateral: $3 \times 3 \times (3) = 9$ parâmetros.
- profundidade: $1 \times 1 \times 3 \times (128) = 384$ parâmetros

Agradecimentos a Chi-Feng Wang (@reina.wang) pelas belas ilustrações

Depthwise separable convolutions - Xception, Mobilenet

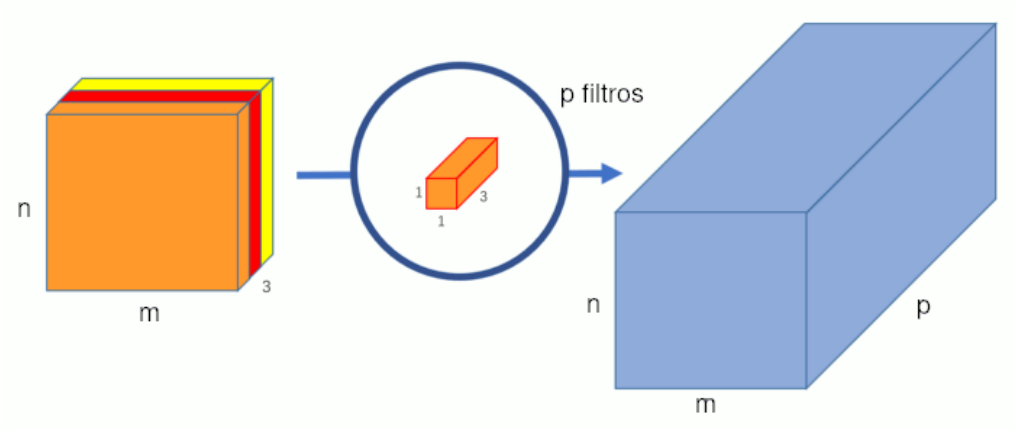


Para obter 128 mapas de características com filtros 3×3 temos:

- lateral: $3 \times 3 \times (3) = 9$ parâmetros.
- profundidade: $1 \times 1 \times 3 \times (128) = 384$ parâmetros
- na imagem $100 \times 100 \times 3$, são 10000 posições,
  - fase 1: $10000 \times 9 = 90.000$ multiplicações
  - fase 2: $10000 \times 384 = 3.840.000$ multiplicações
  - total $= 3.930.000$

Agradecimentos a Chi-Feng Wang (@reina.wang) pelas belas ilustrações

Depthwise separable convolutions - Xception, Mobilenet



Para obter 128 mapas de características com filtros 3×3 temos:

- lateral: $3 \times 3 \times (3) = 9$ parâmetros.
- profundidade: $1 \times 1 \times 3 \times (128) = 384$ parâmetros
- na imagem $100 \times 100 \times 3$, são 10000 posições,
  - fase 1: $10000 \times 9 = 90.000$ multiplicações
  - fase 2: $10000 \times 384 = 3.840.000$ multiplicações
  - total = 3.930.000 contra **16.3 milhões** da convencional

Agradecimentos a Chi-Feng Wang (@reina.wang) pelas belas ilustrações

Bibliography I

Moacir A. Ponti, Fernando dos Santos, Leo Ribeiro, Gabriel Cavallari. **Training Deep Networks from Zero to Hero: avoiding pitfalls and going beyond**.
SIBGRAPI, 2021. Tutorial.
`https://arxiv.org/abs/2109.02752`

Moacir A. Ponti, Leo Ribeiro, Tiago Nazaré, Tu Bui, John Collomosse. **Everything You Wanted to Know About Deep Learning for Computer Vision but were Afraid to Ask**.
SIBGRAPI-T, 2017. Tutorial.