Convolutional NN

Introduction

PSIML 8 PRACTICAL SEMINAR IN MACHINE LEARNING 8

Uros Stegic
uros.stegic@everseen.com

Description

- Process & analyze visual signal
- Extract information from visual signal
- Perform on raw signal (pixel intensities values)

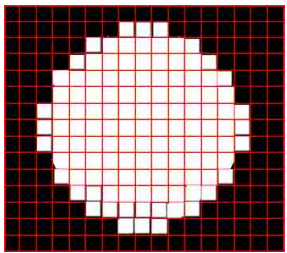
Enhance Intuition

Computer Graphics

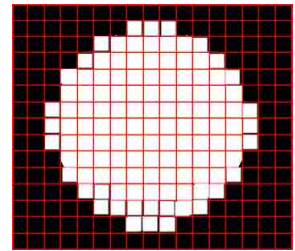
$(x_0, y_0) = (9, 8)$

$r = 7$

$\downarrow$



Computer Vision



$\downarrow$

$(x_0, y_0) = (9, 8)$

$r = 7$

Tasks in Computer Vision

- Object Recognition
- Image Retrieval
- Object Detection
- OCR
- Pose Estimation
- ...

- Tracking
- Scene Reconstruction
- Optical Flow
- Semantic Segmentation
- Image Reconstuction
- ...

Tasks in Computer Vision

- Object Recognition
- Image Retrieval
- Object Detection
- OCR
- Pose Estimation
- ...

- Tracking
- Scene Reconstruction
- Optical Flow
- Semantic Segmentation
- Image Reconstuction
- ...

Convolution Operator - Definition

Let $A, B \in \mathcal{D} \subseteq \mathbb{R}^{n \times n}$. Convolution operator, denoted as $*$ maps the space $\mathcal{D} \times \mathcal{D}$ to a field of real numbers and is defined as follows:

$$A * B = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} B_{ij}$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = 2 * 1 + 4 * 1 + 6 * 1 + 8 * 1 = 20$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = 2 * 1 + 4 * 1 + 6 * 1 + 8 * 1 = 20$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = 2 * 1 + 4 * 1 + 6 * 1 + 8 * 1 = 20$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = 2 * 1 + 4 * 1 + 6 * 1 + 8 * 1 = 20$$

Convolution Operator - Example

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = 2 * 1 + 4 * 1 + 6 * 1 + 8 * 1 = 20$$

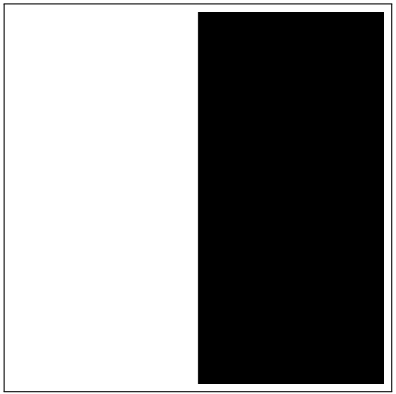
Filters

$$\begin{bmatrix} 211 & 39 & 200 & 102 & 174 & 25 & 90 & 144 \\ 138 & 44 & 184 & 110 & 193 & 30 & 92 & 136 \\ 151 & 73 & 190 & 114 & 189 & 41 & 105 & 128 \\ 129 & 101 & 123 & 181 & 201 & 169 & 117 & 191 \\ 140 & 122 & 153 & 231 & 209 & 157 & 124 & 113 \\ 221 & 115 & 77 & 244 & 198 & 149 & 156 & 247 \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

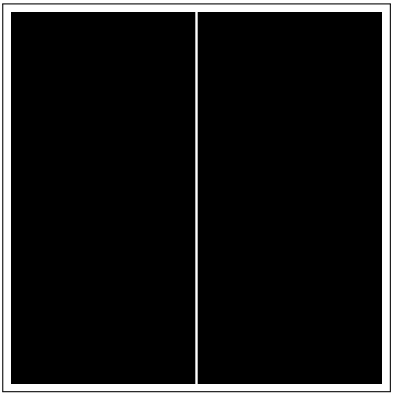
Filters - Examples

- Vertical Edge Extractor
- Horizontal Edge Extractor
- Sobel filter
- Sharpen
- Gaussian Blur

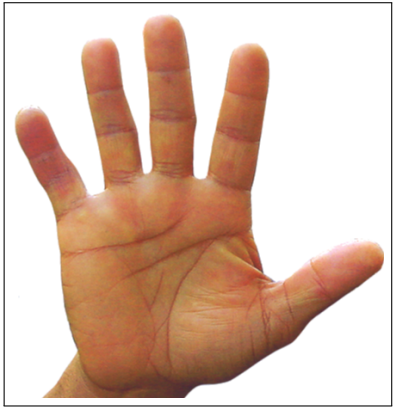
Filters - Edge Extractor



$$* \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} =$$

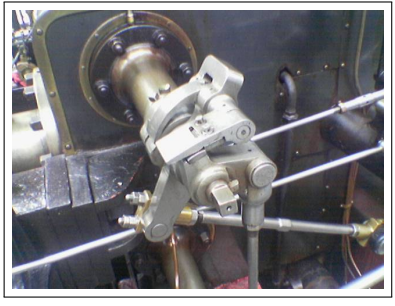


Filters - Edge Extractor

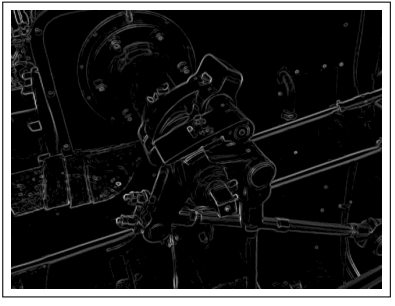


$$* \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} =$$

Filters - Sobel



$$* \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} =$$



Filters - Gaussian Blur

$$* \frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} =$$

Multiple Input Channels

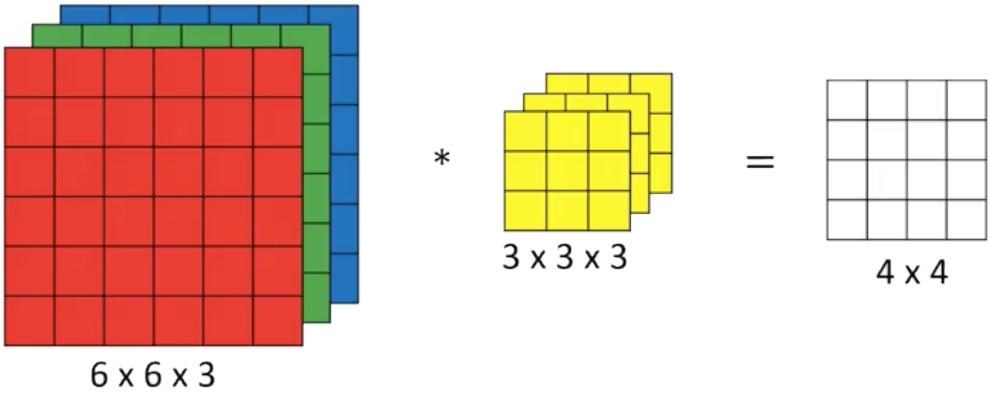
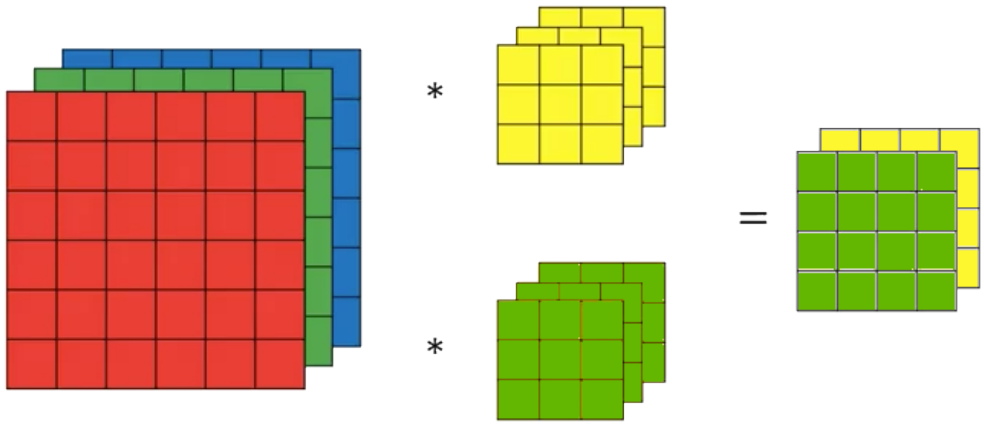


Figure: Convolution of multichannel image

Multiple Filters



Basic Concepts

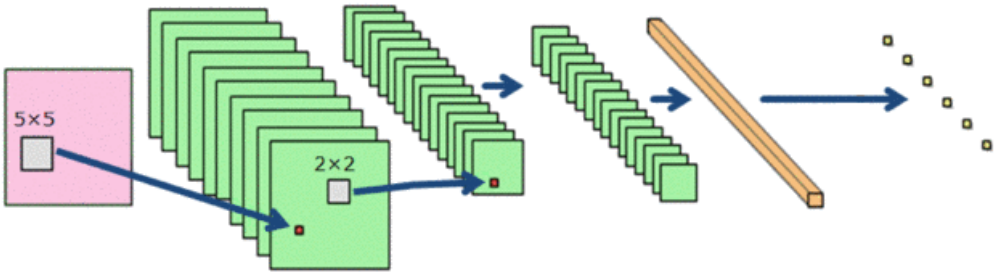


Figure: Convolutional layers stacked

Basic Concepts - Takeaway

- Image Classification
- Parameters (filters) Learning [LBD+89]
- Weight Sharing
- Feature Extraction
- Translation invariant

Basic Concepts - Feature Abstractions

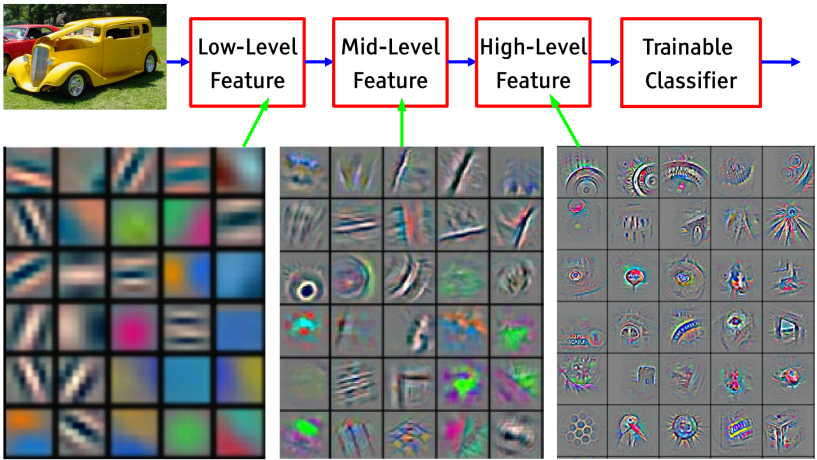


Figure: Feature Visualization [ZF13]

Basic Concepts - Pooling Layers

- Sampling important Features
- Reduce Computation Time
- Make Features Robust

Basic Concepts - Pooling Layers (Example)

Pooling Layer - Max Pooling

$$\begin{bmatrix} 9 & 2 & 4 & 1 \\ 3 & 1 & 8 & 2 \\ 4 & 5 & 9 & 2 \\ 5 & 6 & 0 & 1 \end{bmatrix} \longrightarrow \begin{bmatrix} 9 & 8 \\ 6 & 9 \end{bmatrix}$$

Basic Concepts - Architecture

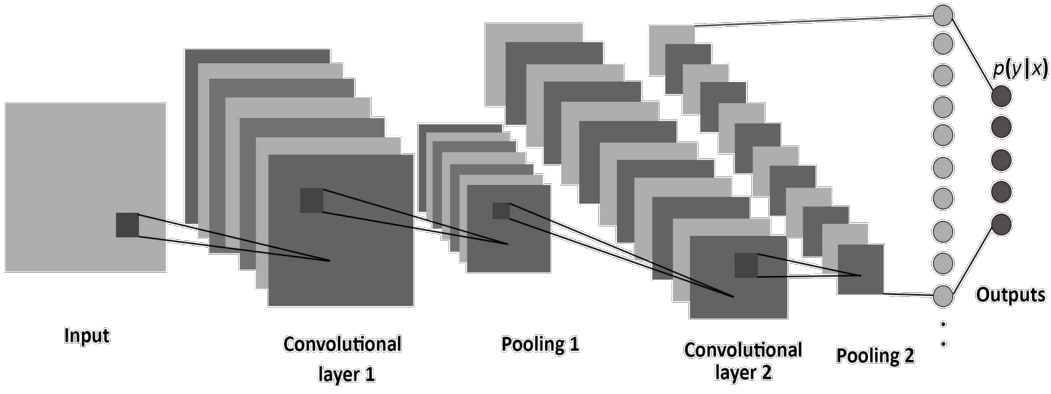


Figure: Convolutional Neural Network - Example

CNN Architecture - Lenet-5

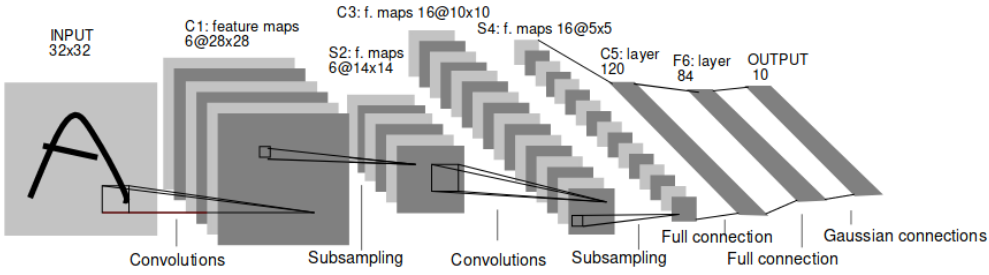


Figure: Lenet-5 Architecture [LBBH98]

CNN Architecture - AlexNet

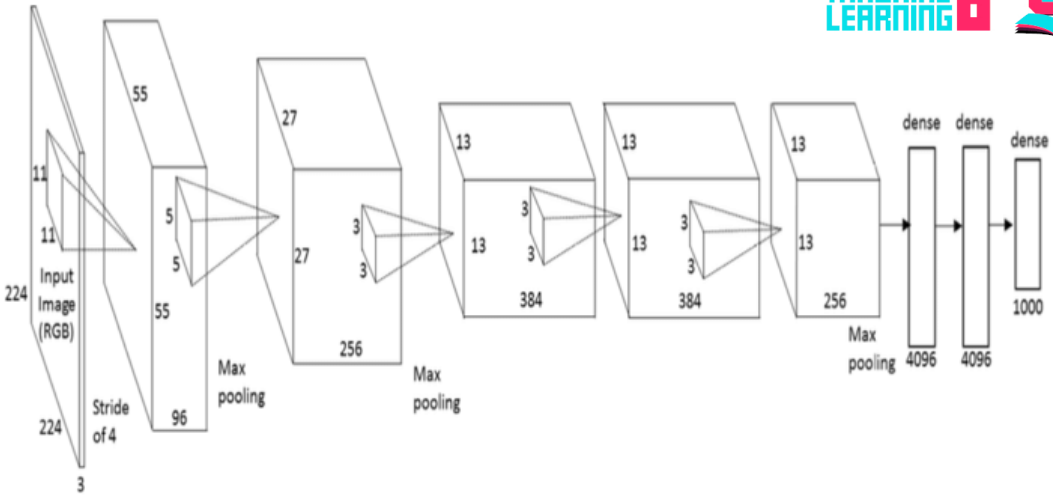


Figure: AlexNet Architecture [KSH12]

CNN Architecture - VGG

Figure: VGG Architecture

CNN - Problems

- Vanishing Gradient
- Exploding Gradient
- Computational Complexity

Residual Block

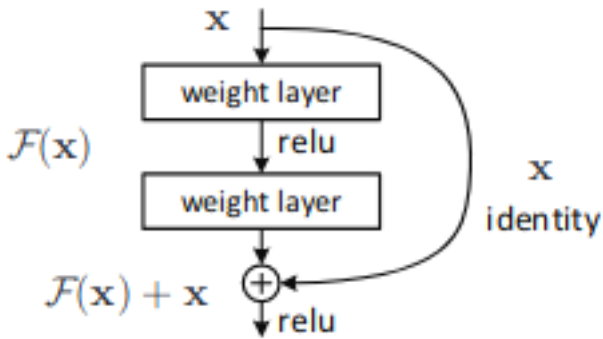


Figure: Residual Block (Skip Connection) [HZRS15]

Residual Network

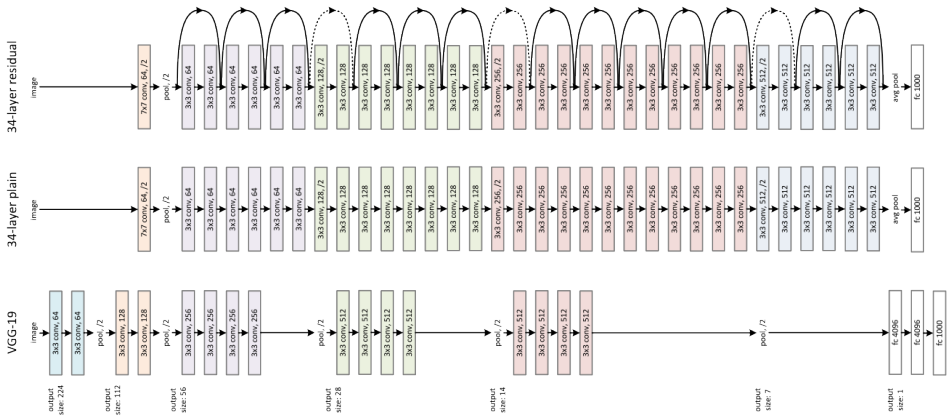


Figure: CNN Architecture - ResNet-34 [HZRS15]

1x1 Convolution

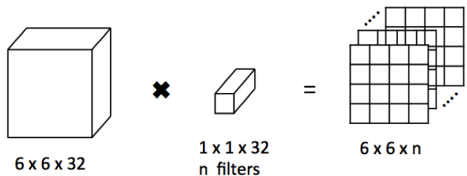


Figure: 1x1 Convolution [LCY13]

Inception Module - Idea

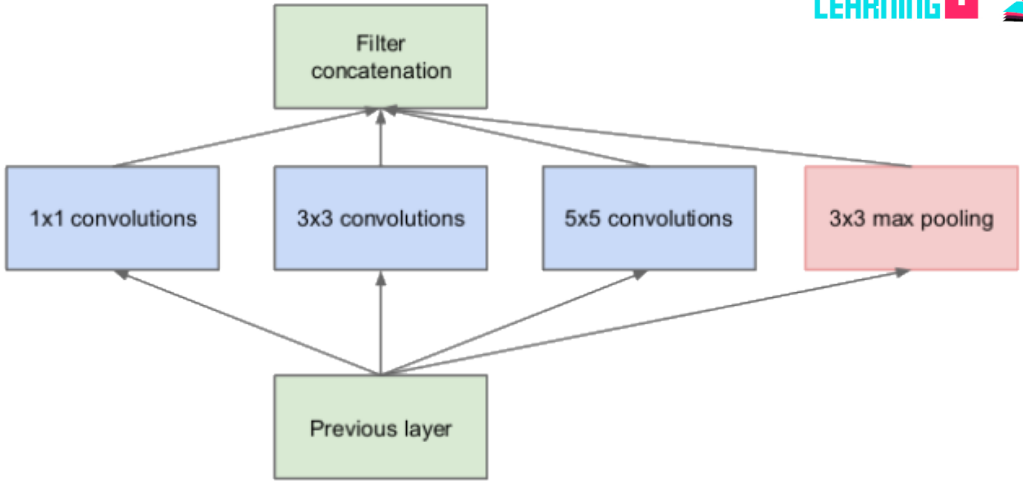


Figure: Inception Module Naive Version [SLJ+14]

Inception Module - Redone

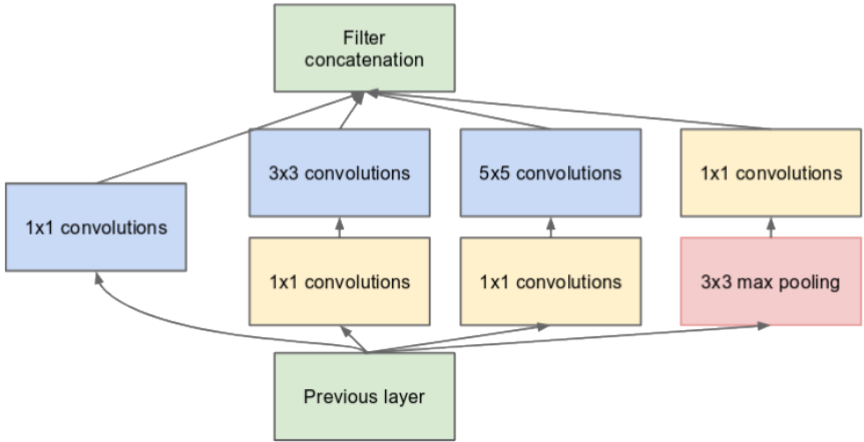


Figure: Inception Module With Dimension Reduction [SLJ+14]

Inception Network

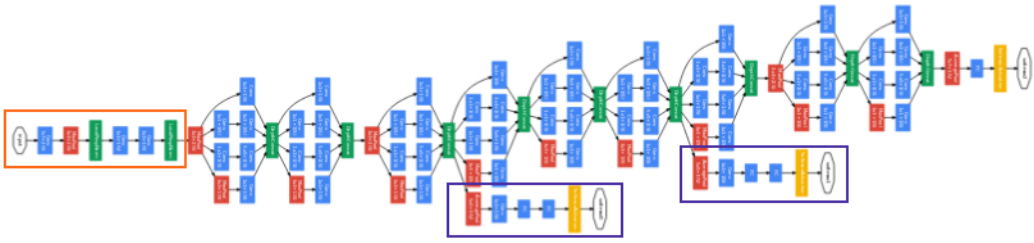


Figure: Inception Network (GoogLeNet) [SLJ+14]

CONVERGENCE

References I

- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, Deep residual learning for image recognition, CoRR **abs/1512.03385** (2015).
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 25 (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), Curran Associates, Inc., 2012, pp. 1097–1105.
- Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, IEEE (1998), 2278–2324.
- Yann Lecun, Bernhard Boser, John Denker, Don Henderson, R E. Howard, W.E. Hubbard, and Larry Jackel, Backpropagation applied to handwritten zip code recognition, Neural Computation **1** (1989), 541–551.
- Min Lin, Qiang Chen, and Shuicheng Yan, Network in network, CoRR **abs/1312.4400** (2013).

References II

- Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, Going deeper with convolutions, CoRR **abs/1409.4842** (2014).
- Matthew D. Zeiler and Rob Fergus, Visualizing and understanding convolutional networks, CoRR **abs/1311.2901** (2013).