

Lecture 06 – Deep Learning and CNNs

Prof. André Gustavo Hochuli

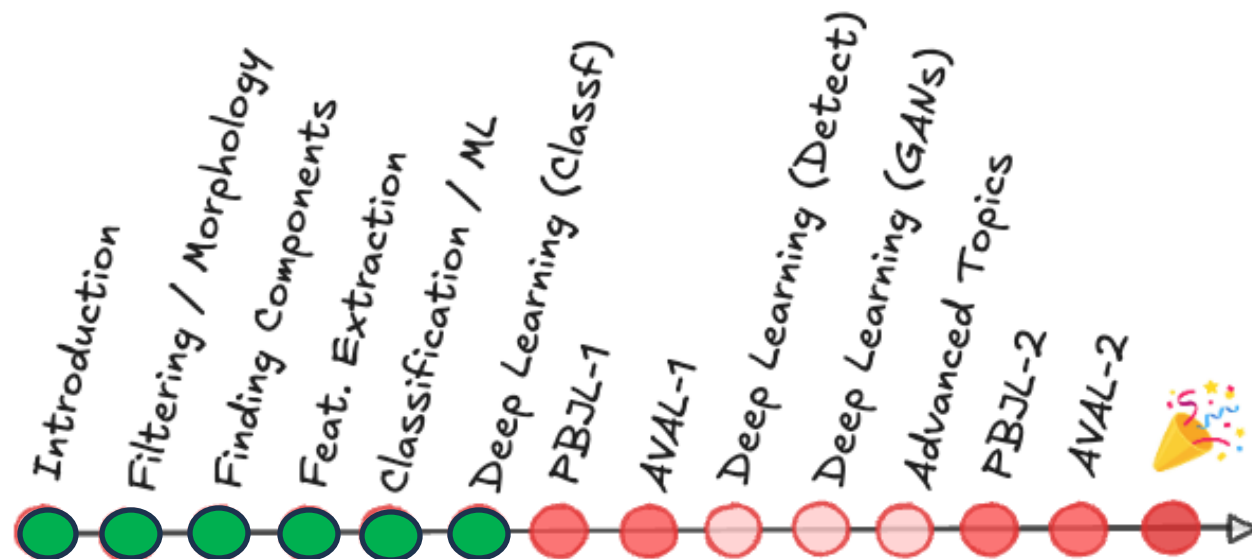
gustavo.hochuli@pucpr.br

aghochuli@ppgia.pucpr.br

Topics

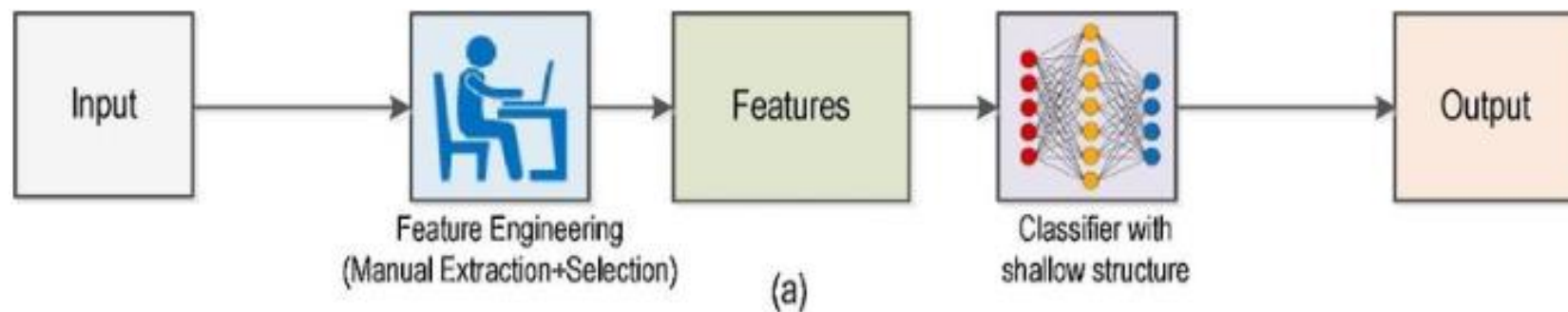
Deep Learning

- Traditional Convolutional Neural Network
 - Lenet
 - Imagenet
 - Deep Networks
- Practice

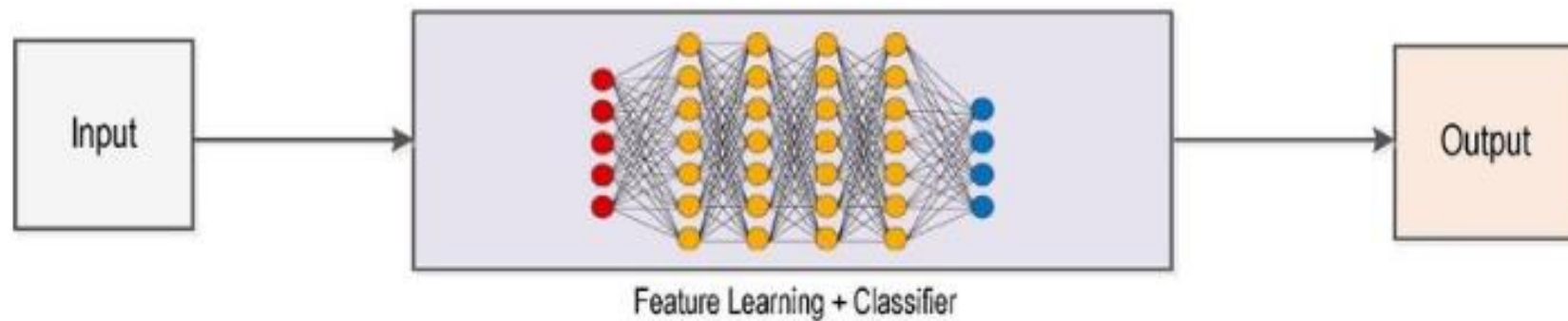


Traditional and Deep Learning

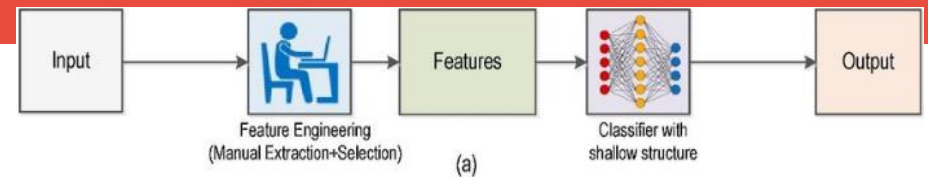
- Traditional (“Shallow”)



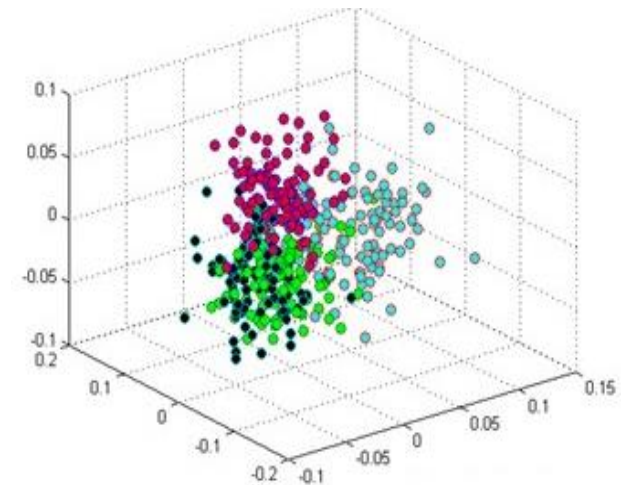
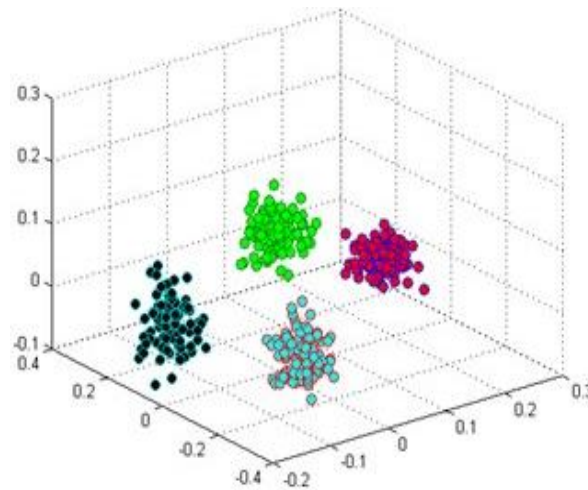
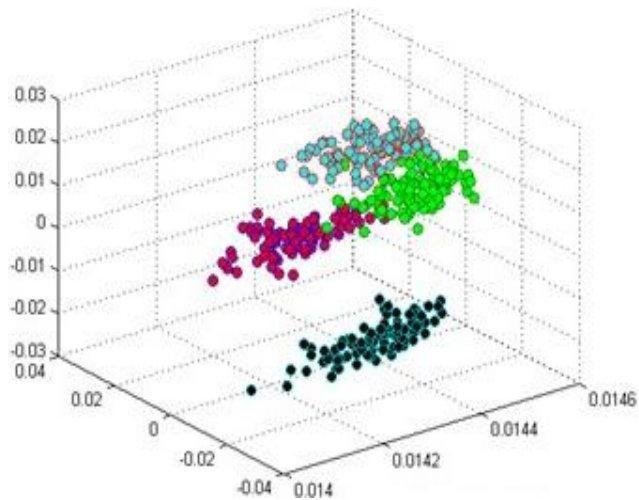
- Deep



“Shallow” Learning

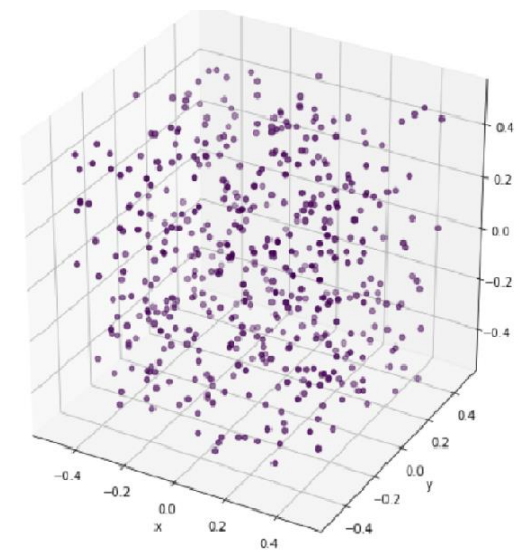
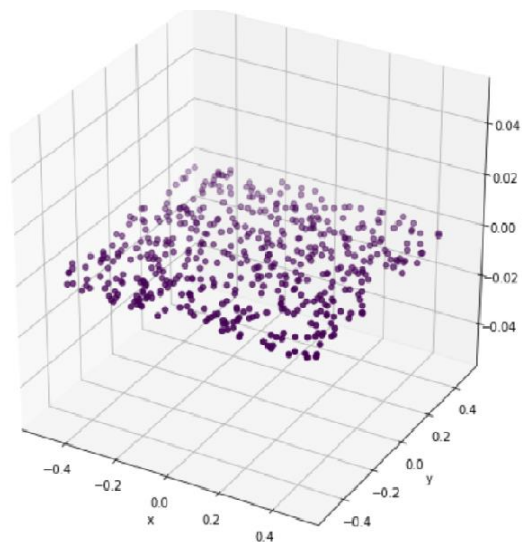
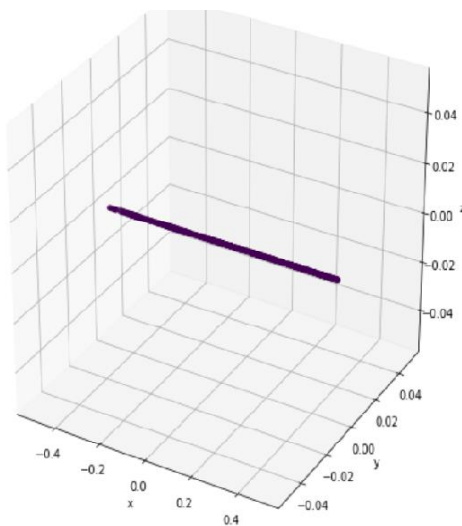


How to effectively capture discriminative features from images?



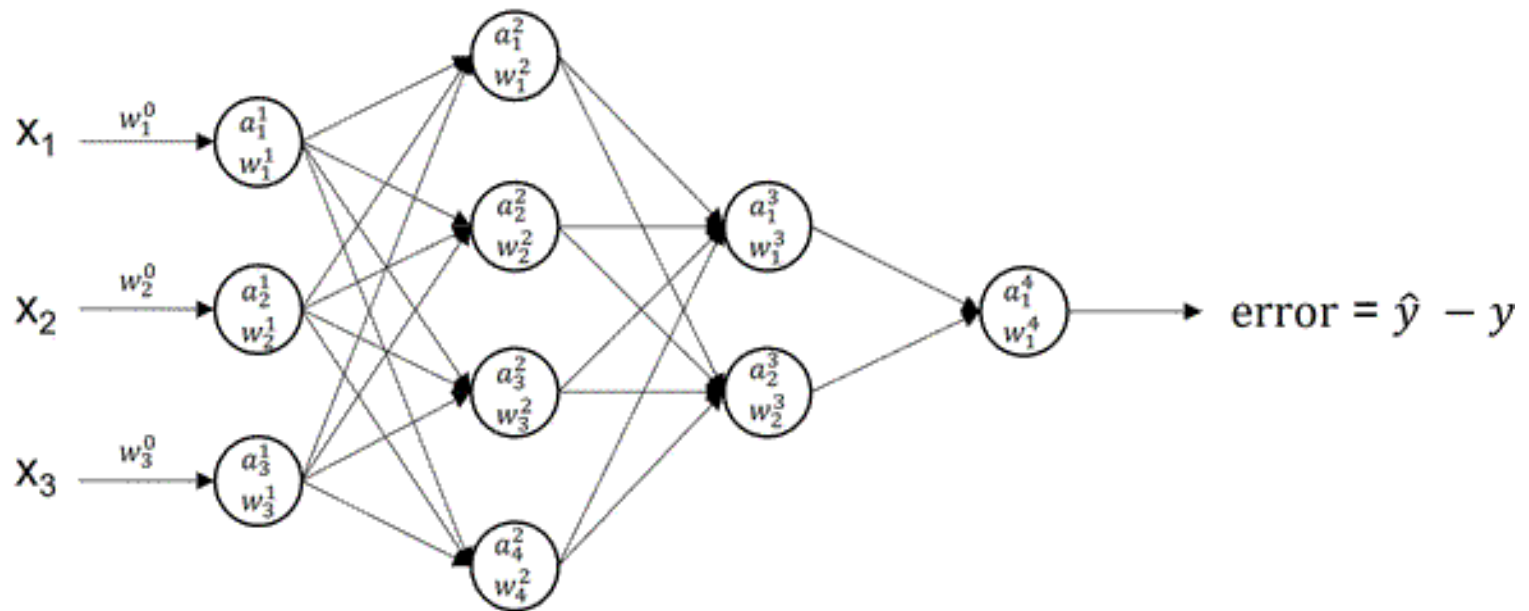
The Curse of Dimensionality

- Most of us have very reasonable intuition that more information in terms of features (dimensions) is always better. **Is this always true?**
 - **Exponential Growth of Feature Space** → sparsity increases
 - **Distance Metrics Lose Meaning** → similarity becomes unreliable
 - **More Data Needed** → exponential increase in samples
 - **Risk of Overfitting** → noise dominates patterns
 - **Mitigation** → feature selection, PCA, autoencoders



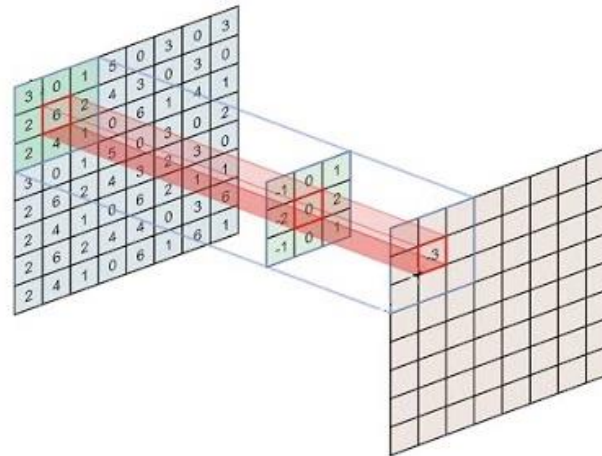
Multi-Layer Perceptron: Recap

- Features
- Weights
- FeedForward
- Error and BackPropagation

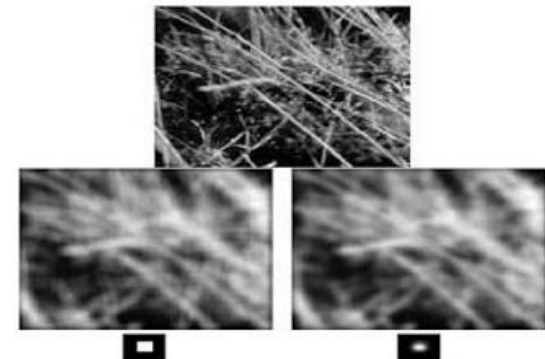


Kernel-based Descriptors: Recap

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9



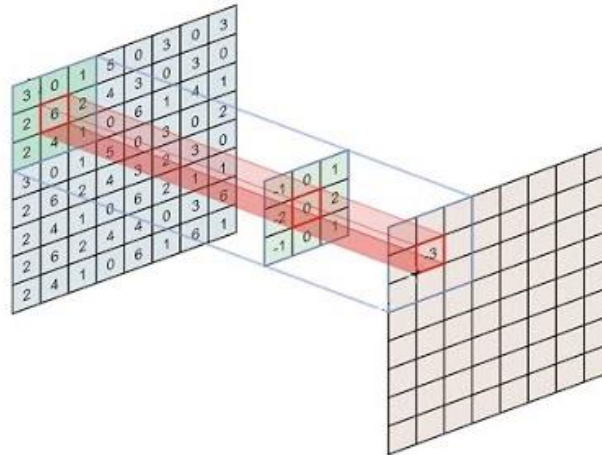
$$x * y = \sum_{i=1}^m \sum_{j=1}^n x_{(m-i)(n-j)} y_{(i)(j)}$$



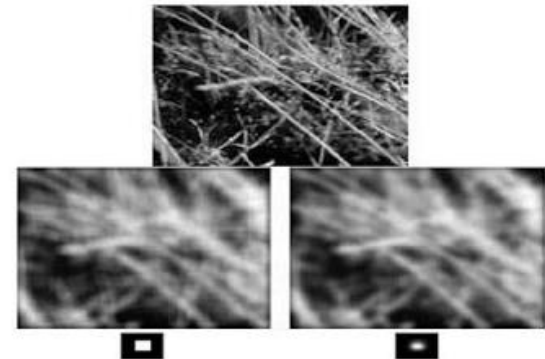
Kernel-based Descriptors: Recap

- Why not extend weight optimization to learning kernels (features)?

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

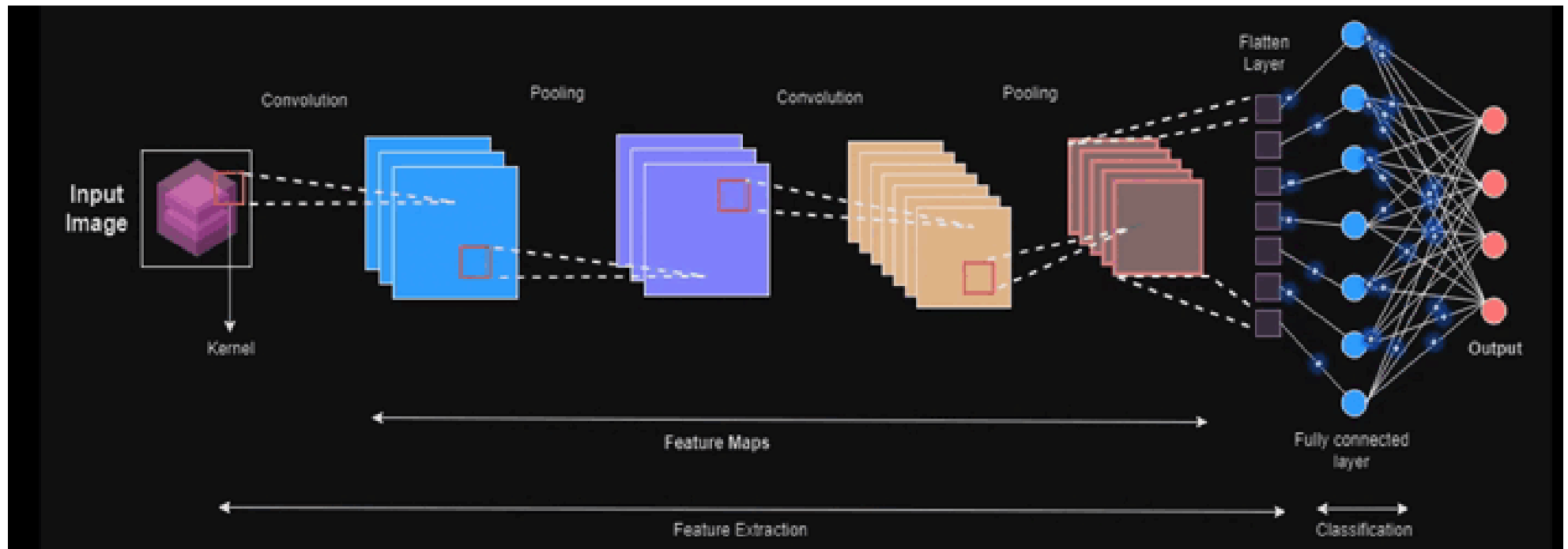


$$x * y = \sum_{i=1}^m \sum_{j=1}^n x_{(m-i)(n-j)} y_{(i)(j)}$$



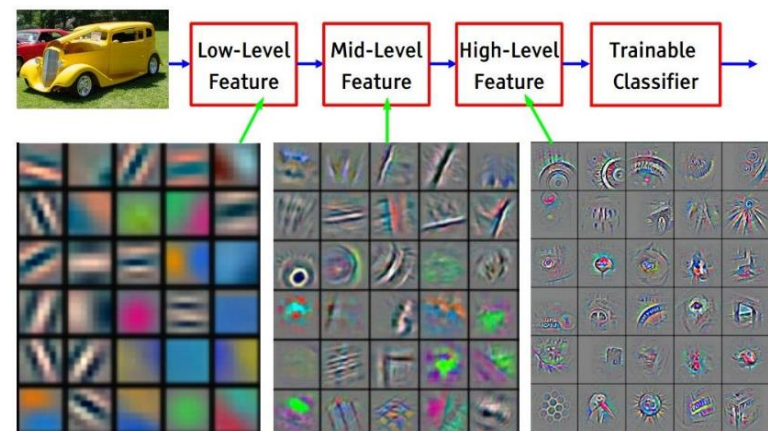
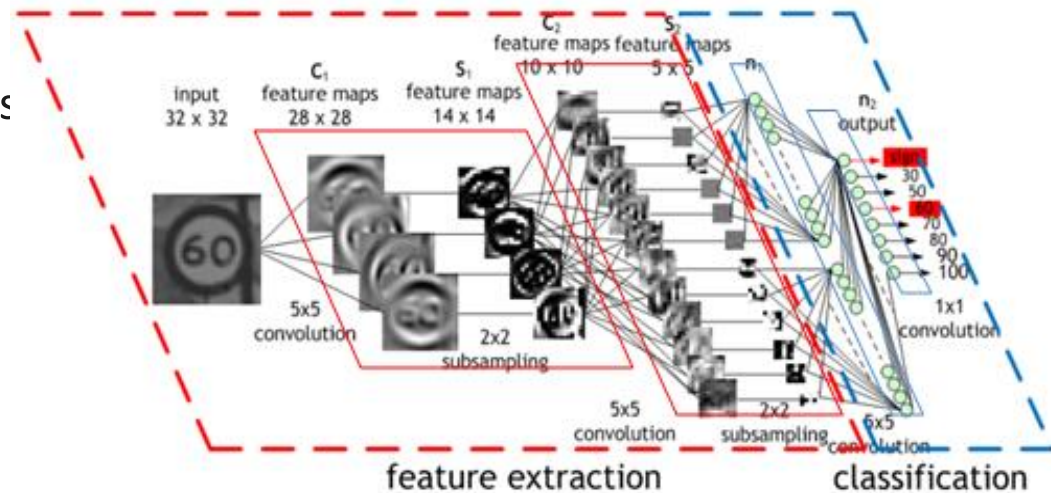
Deep Learning

- **Why not extend weight optimization to learning kernels (features)?**



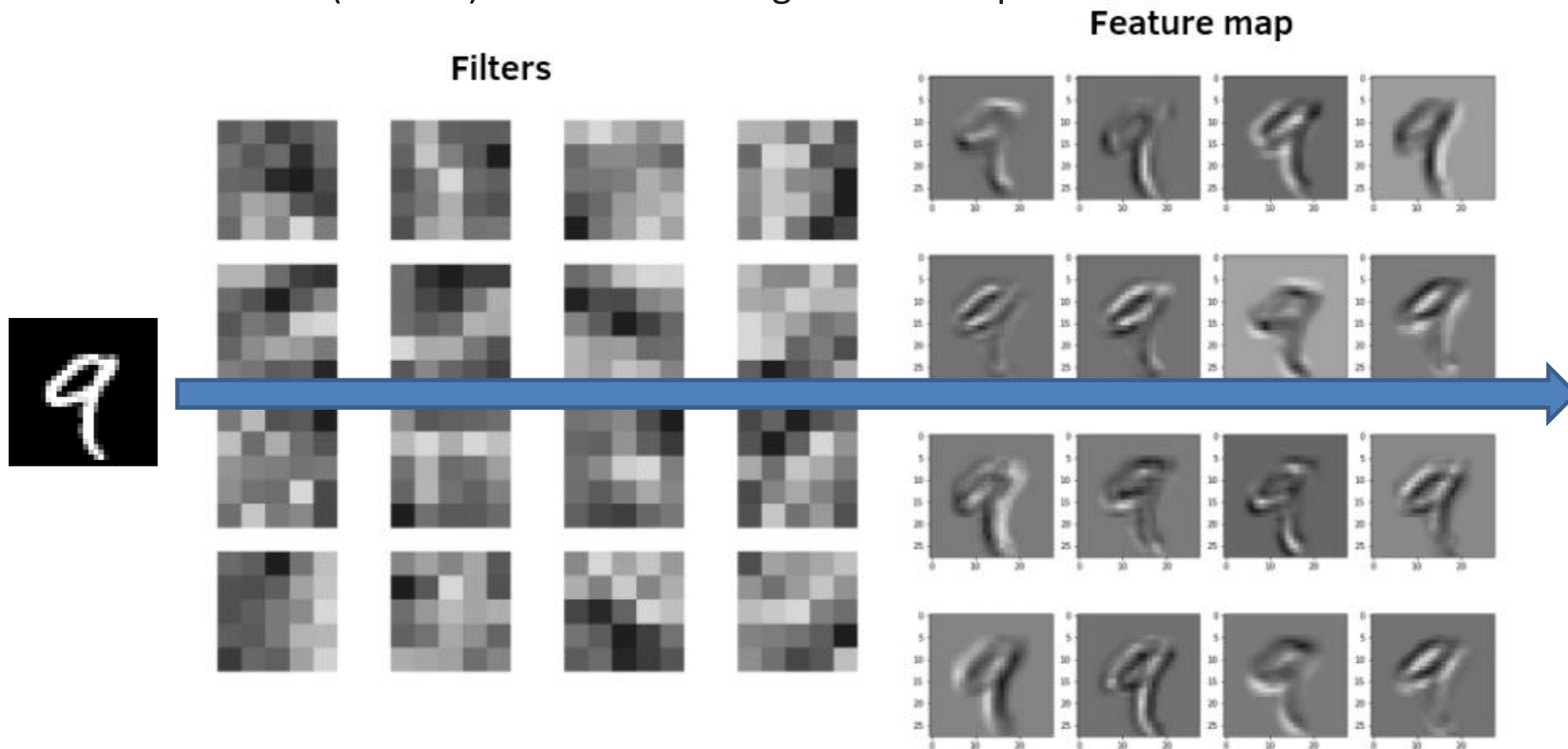
Convolutional Neural Networks

- Feature Extraction
 - Sequential Convolutional Layers
 - Learnable Filters (Kernels)
 - Activation Functions(ReLU)
 - Pooling Layers
- Classification
 - Fully-Connected Layers
 - SVM
 -



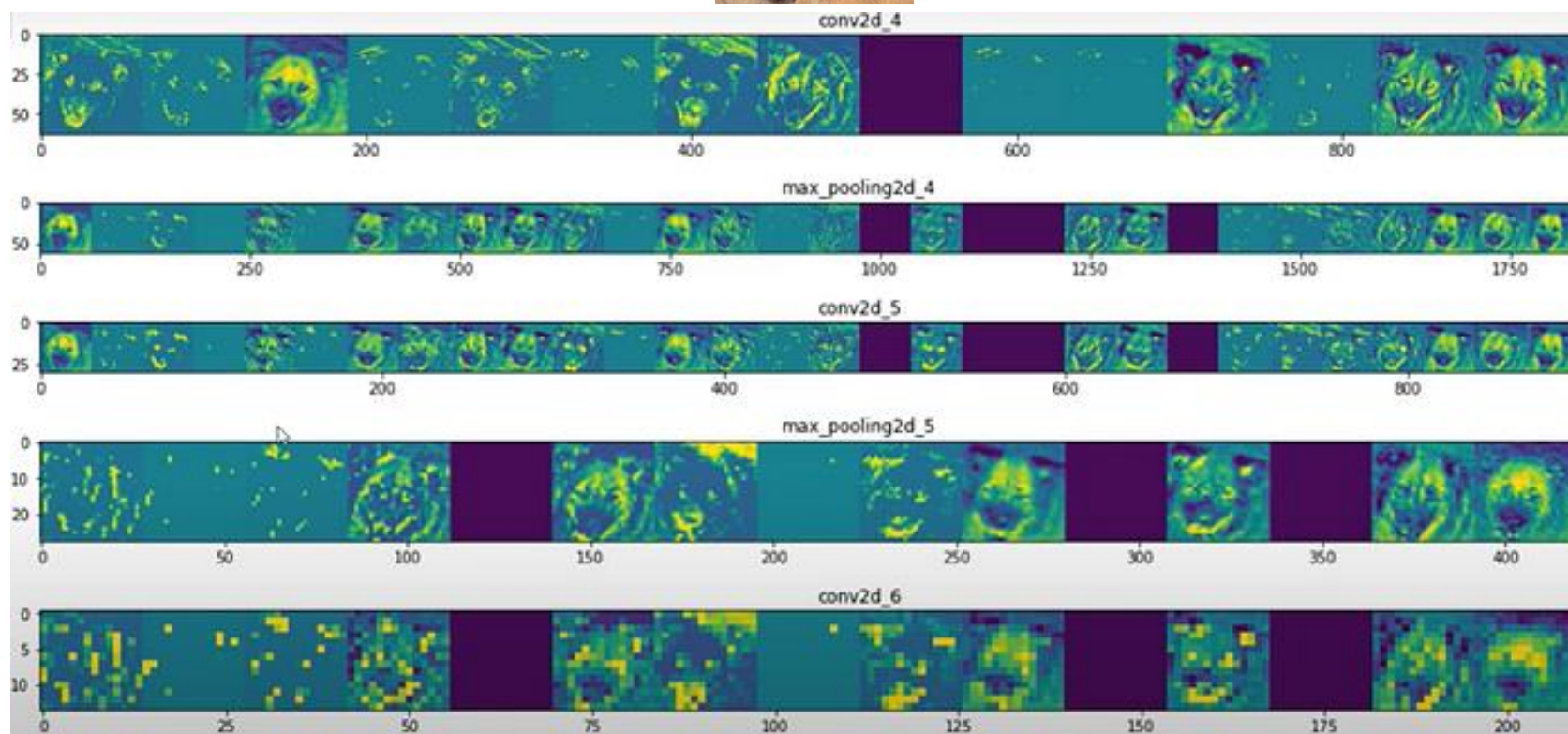
Convolutional Neural Networks

- Learned Filters (Kernels) and The Resulting Feature Map



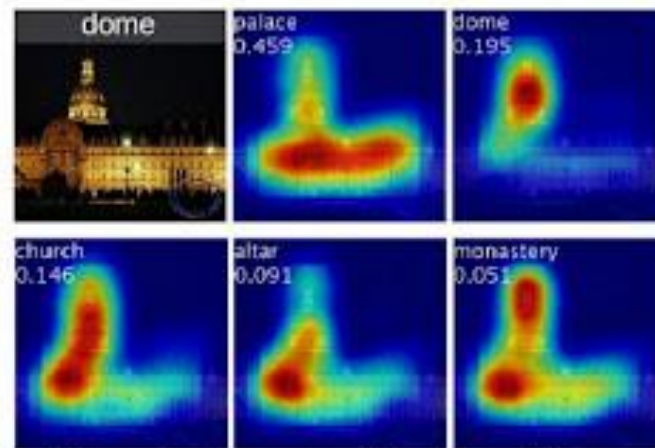
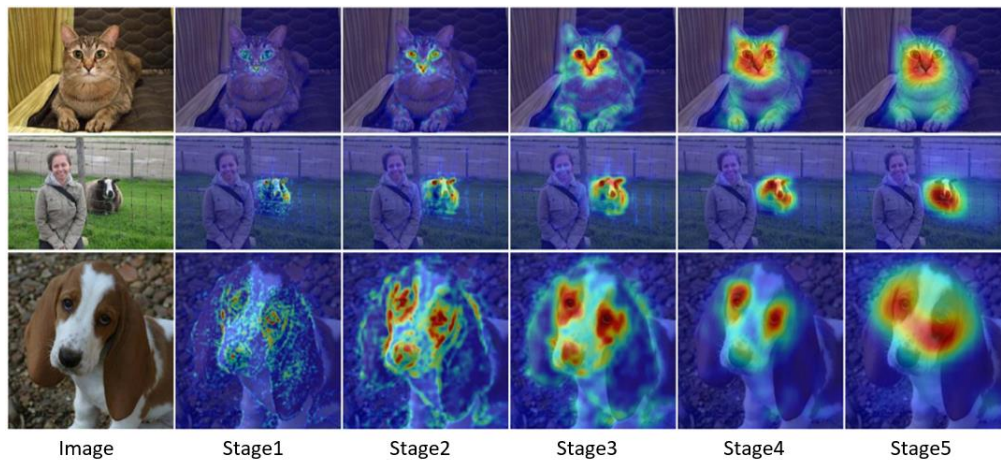
Convolutional Neural Networks

- Feature Maps

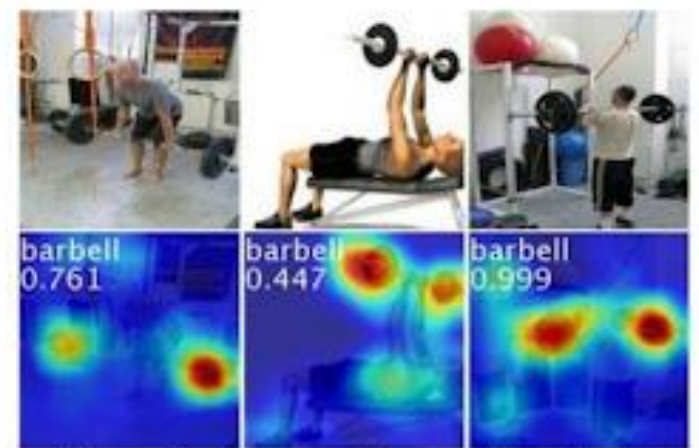


Convolutional Neural Networks

- Activations Maps



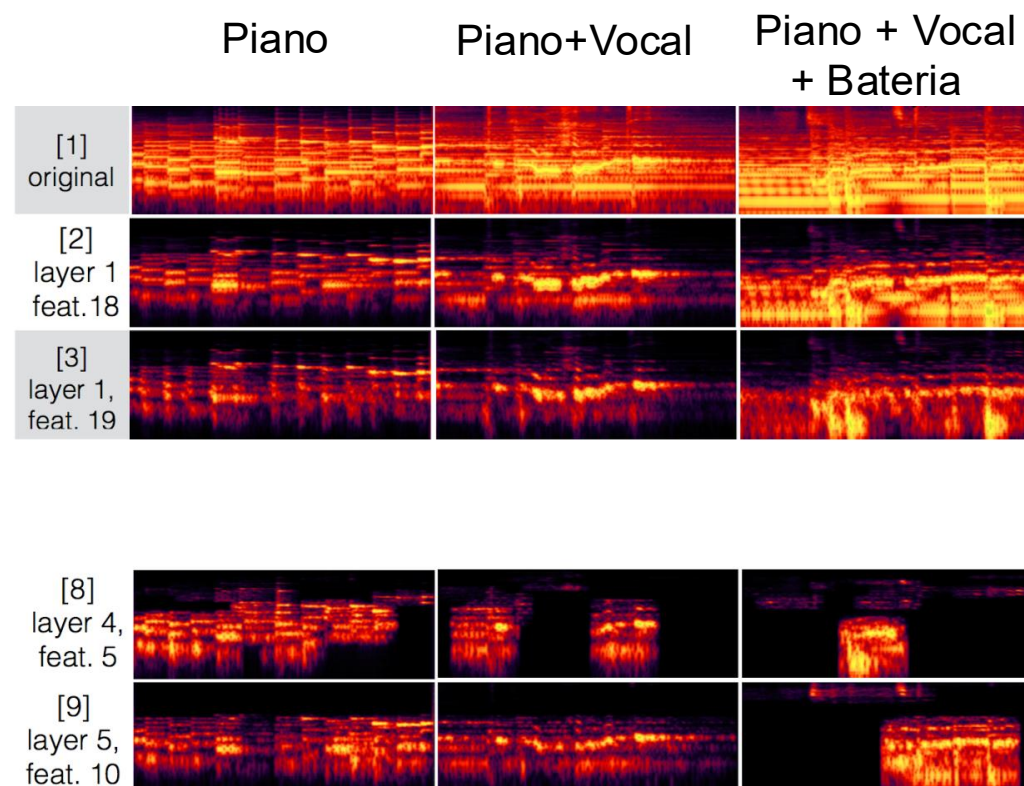
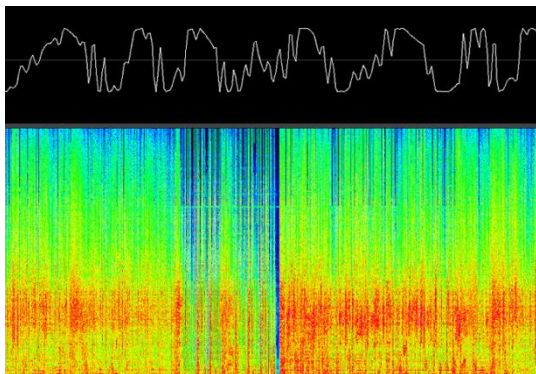
Class activation maps of top 5 predictions



Class activation maps for one object class

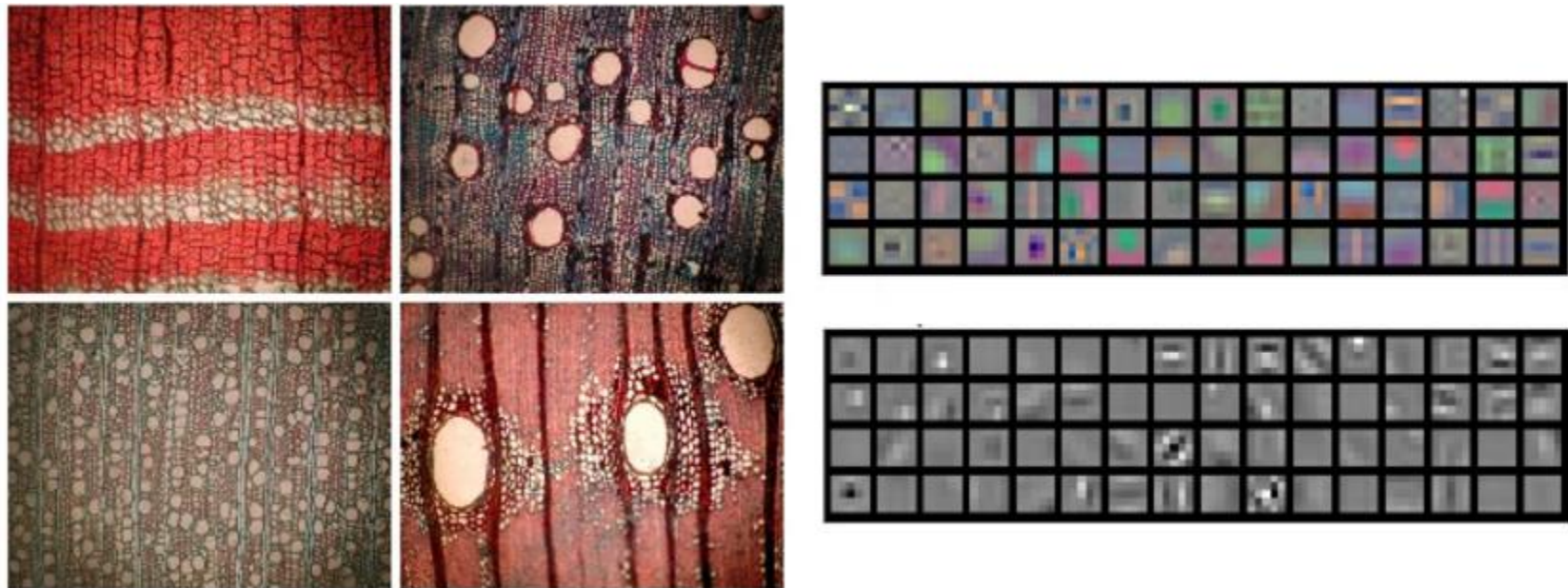
Convolutional Neural Networks

- Wide applied in the Computer Vision area (audio, images, video processing, etc.).



Convolutional Neural Networks

- Tissue Classification
- Medical Images



Convolutional Neural Networks

- Face

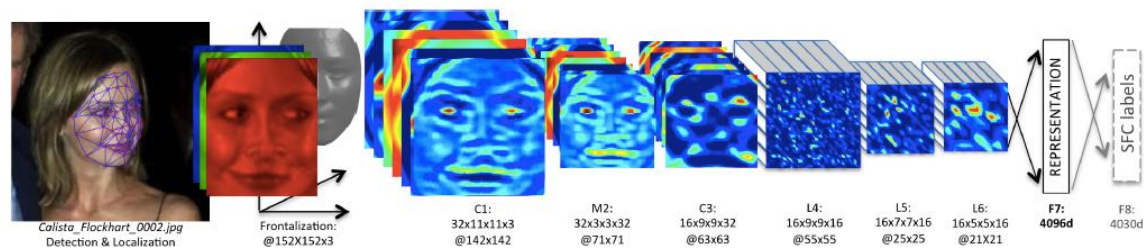


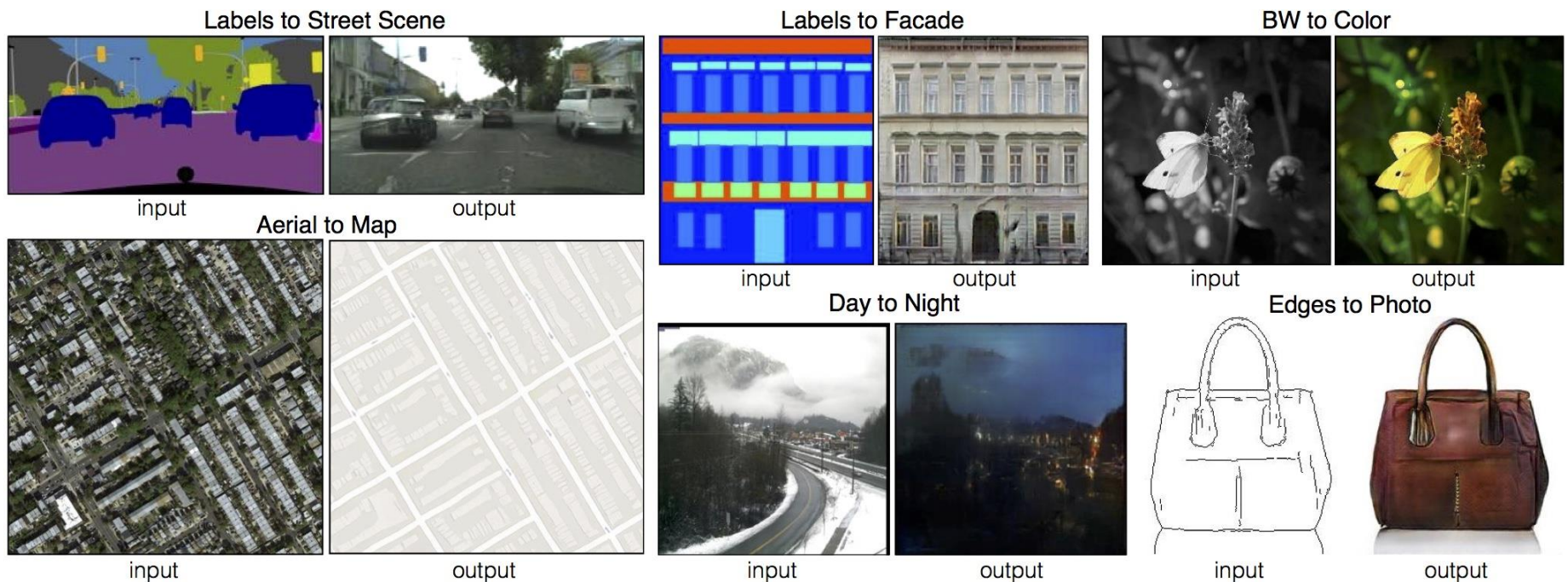
Figure 2. Outline of the *DeepFace* architecture. A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate outputs for each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

- PKLot



Convolutional Neural Networks

- Image Translation



Convolutional Neural Networks

- Deep Fakes

Animating Faces

A single model animates all images given only a single source image



<https://www.youtube.com/watch?v=mUfJOQKdtAk>

Convolutional Neural Networks

Pros

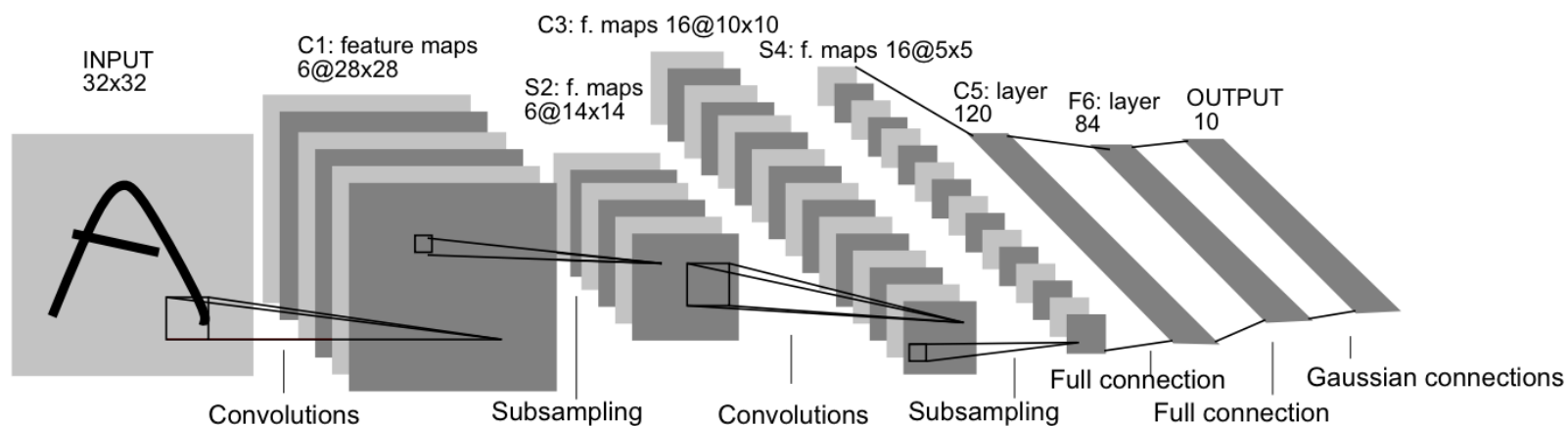
- Enables learning of features rather than hand tuning
- Impressive performance gains on
 - Computer vision
 - Speech recognition
 - Some text analysis
- Potential for much more impact

Cons

- Computationally really expensive
- Requires a lot of data for high accuracy
- Extremely hard to tune
 - Choice of architecture
 - Parameter types
 - Hyperparameters
 - Learning algorithm
 - ...
- Computational + so many choices = incredibly hard to tune

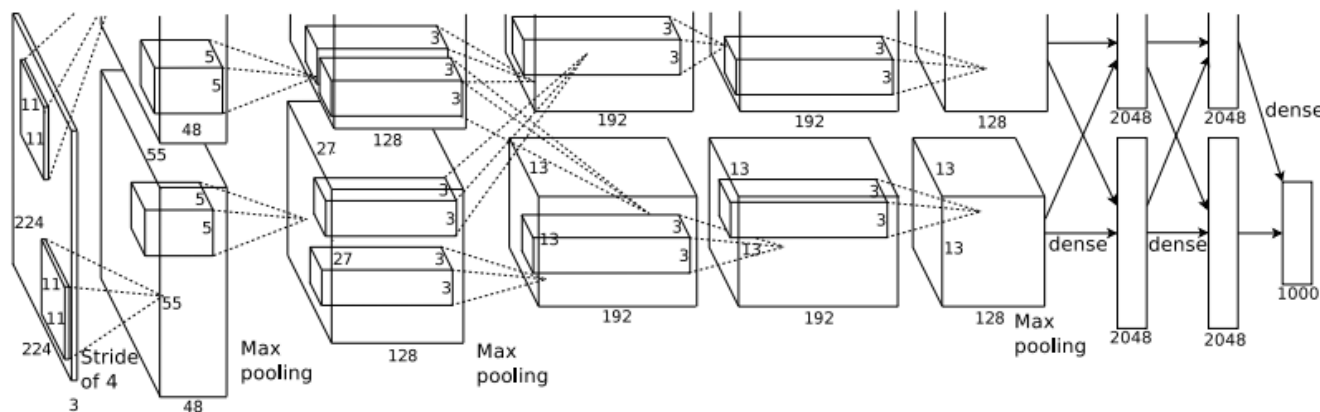
Convolutional Neural Networks

- Lenet
 - Yan Lecun – 90 's (Bell Labs / IBM / FACEBOOK)
 - Handwritten Digits
 - ~60 K Parameters
 - ~345 K Connections



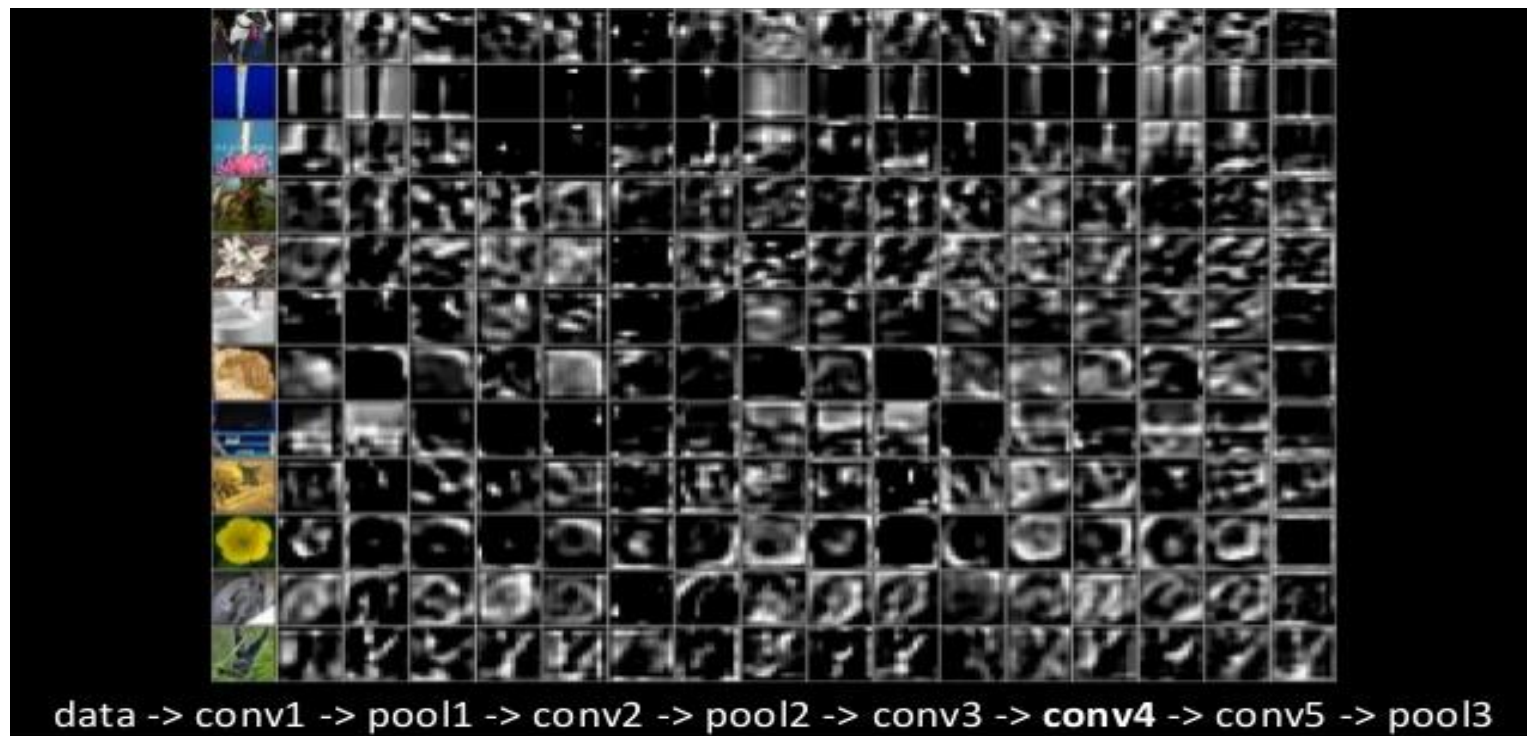
Convolutional Neural Networks

- AlexNet
 - Alex Krizhevsky – 2012 (Krizhevsky Net)
 - Imagenet 2012 Challenge (1000 classes)
 - 1.2 M Train, 50K Val, 150K Test
 - 2012 Winner (15.3% Error - Top 5)
 - 2^o SIFT Based (26.2%)



Convolutional Neural Networks

- Alexnet

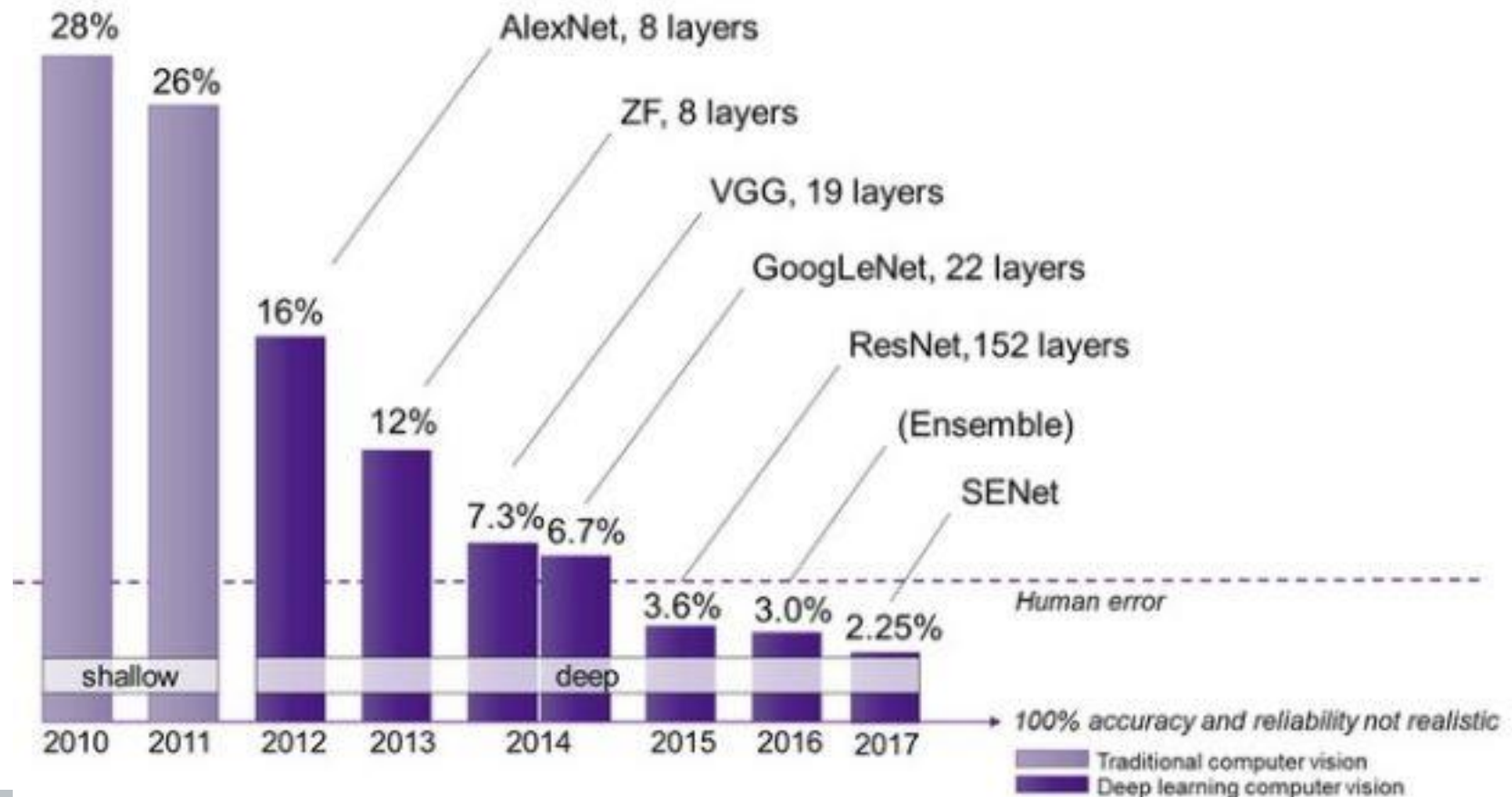


Imagenet Challenge

- Imagenet Challenge (Classification)
 - 1000 classes
 - 1.2 M Train
 - 50K Val
 - 150K Test



ImageNet Large Scale Visual Recognition Challenges



Let's Code

- [\[LINK\]](#)