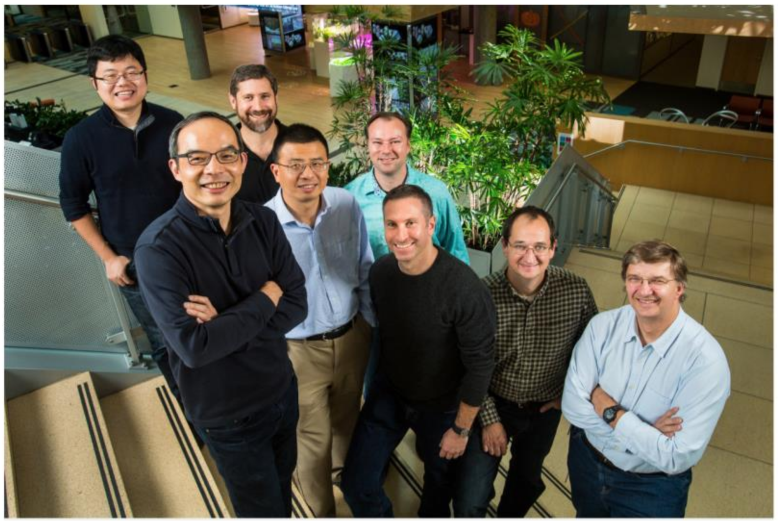
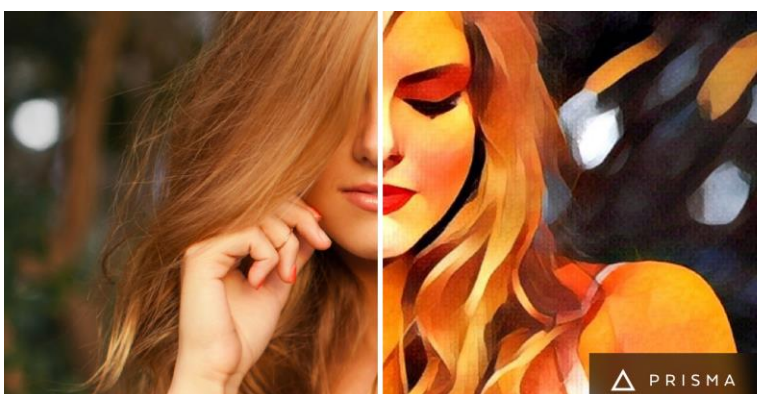
Deep Learning in Visual Recognition

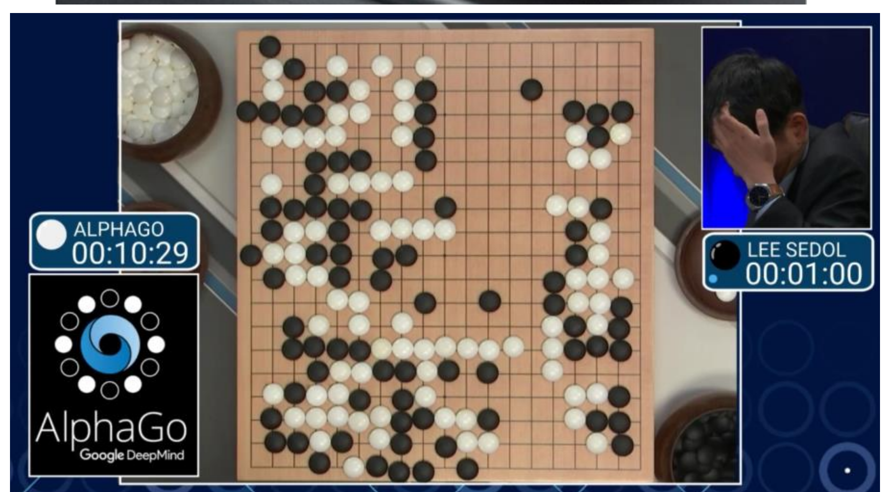
Thanks Da Zhang for the slides

Deep Learning is Everywhere

Historic Achievement: Microsoft researchers reach human parity in conversational speech recognition







Roadmap

- ❑ Introduction
- ❑ Convolutional Neural Network
- ❑ Application
    - ❑ Image Classification
    - ❑ Object Detection
    - ❑ Object Tracking
- ❑ Our Work
- ❑ Conclusion and Discussion

Roadmap

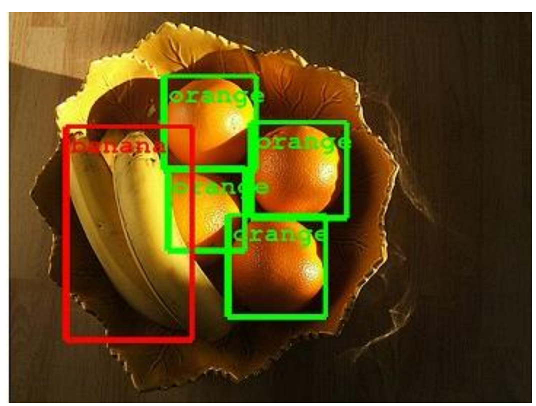
- Introduction
- Convolutional Neural Network
- Application
  - Image Classification
  - Object Detection
  - Object Tracking
- Our Work
- Conclusion and Discussion

Computer Visual Recognition

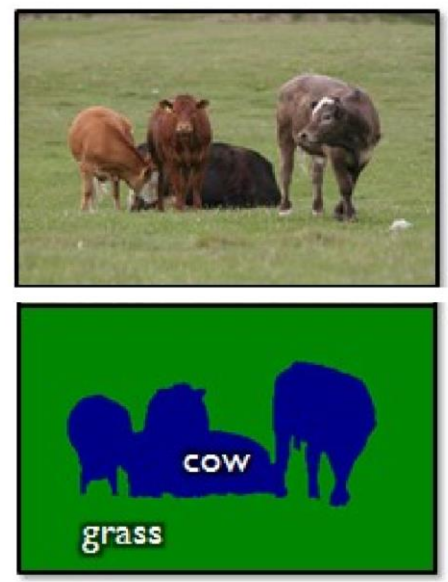
❑Definition

Visual Recognition deals with how computers can be made to gain high-level understanding from digital images or videos. It seeks to automate tasks that the human visual system can do.

Image classification



Object detection



Semantic segmentation

Computer Visual Recognition

- Semantic gap between human and computer
  - Colorful image
  - 3-D intensity matrix

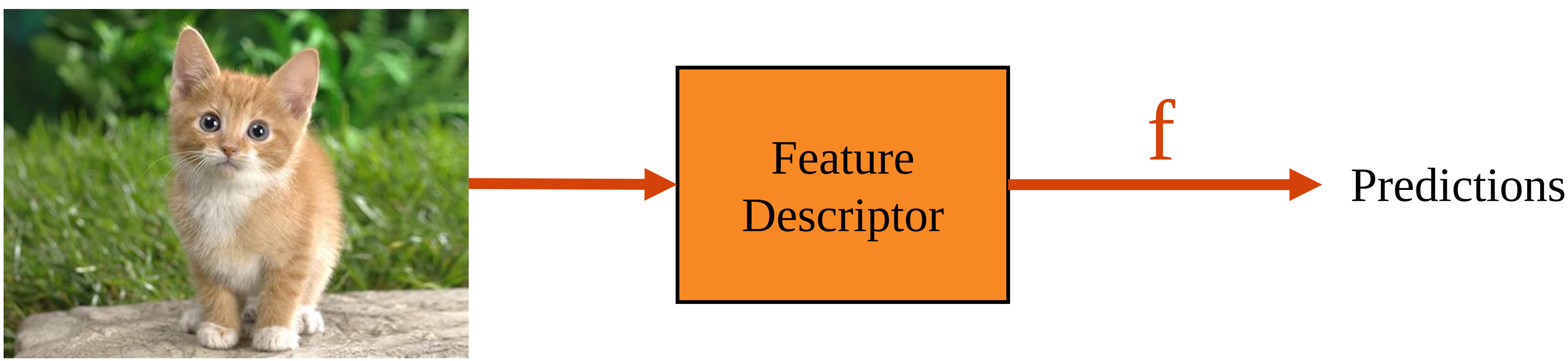
Human Vision

79	6D	80	6E	54	0C	0D	09	0A	06	04
7E	8C	73	7A	5C	1E	05	0A	0F	0E	0C
89	93	8B	83	69	43	07	0A	12	0A	0B
91	93	8C	7F	6F	5F	0B	09	12	0D	0C
30	48	62	87	71	5C	0A	08	11	0C	09
8A	5A	3D	42	76	5C	13	08	13	0F	0C
42	39	73	7D	89	46	12	06	12	12	0F
0F	22	4B	C3	A4	3F	4F	0C	18	16	0F
75	4B	AC	A1	B5	79	0C	0B	13	0F	0B
5F	3E	98	B7	B7	A3	31	11	14	0A	0D
82	70	9F	AE	AD	A5	92	16	10	07	0E

Computer Vision

Feature Descriptor

- Feature Descriptor is an algorithm which takes an image and outputs feature vectors.
- Feature Descriptors encode interesting information and can be used to differentiate one feature from another.

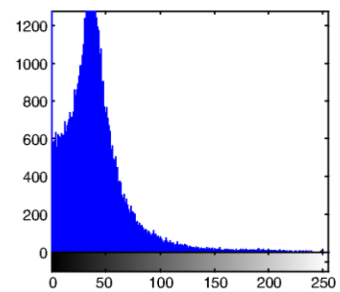


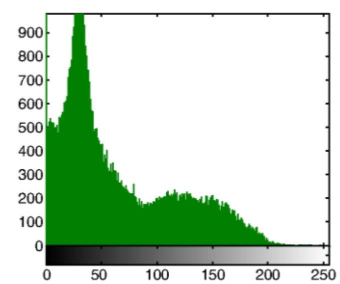
Feature Descriptor – Toy Example

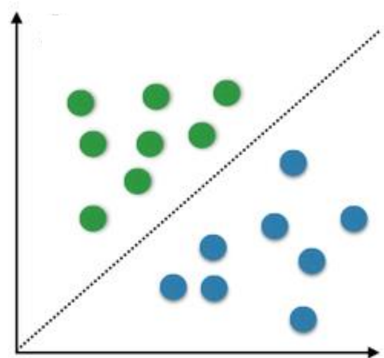
Feature Descriptor

f

Class Scores







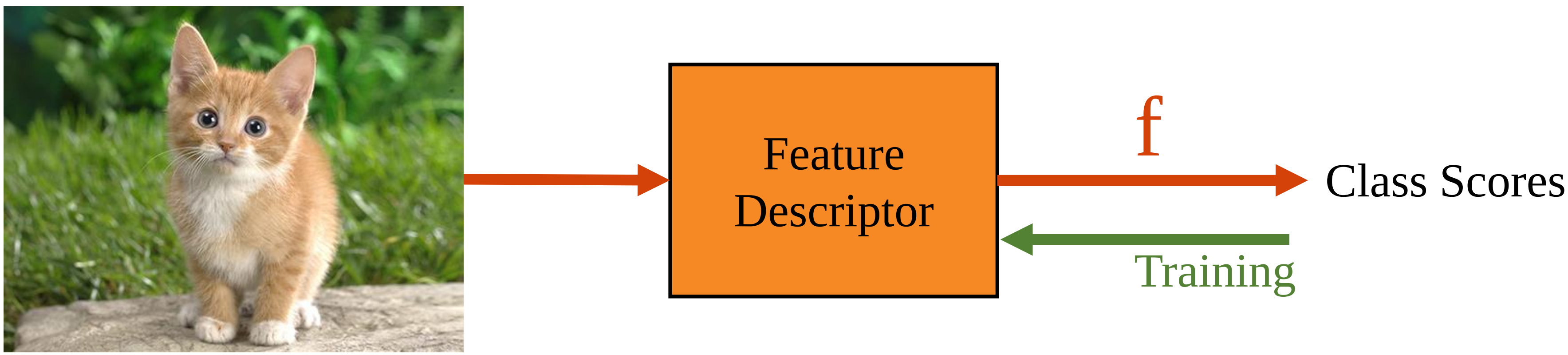
Sky

Grass

Feature Descriptor - Before Deep Learning

- Finding better features
  - Corner, blob, SIFT, HoG …
- Training better classifiers
  - Logistic regression, Random forest, Adaboost, Support Vector Machine …

From Feature Descriptor to Deep Learning



- ❑ Problems with traditional feature descriptors
  - ❑ Hand-crafted features.
  - ❑ Only machine learning model is tuned during training.
  - ❑ Features are low-level and lack generality.

- ❑ Deep Learning

  Hierarchical and general features are automatically learned through an end-to-end training process.

Roadmap

- ❑ Introduction
- ❑ Convolutional Neural Network
- ❑ Application
  - ❑ Image Classification
  - ❑ Object Detection
  - ❑ Object Tracking
- ❑ Our Work
- ❑ Conclusion and Discussion

Auto Encoder

- It is hard to train multiple layers of neurons together
- How about training layer-by-layer?
- But what should be the “merit” of each layered training? What is the cost function?
- Auto encoder: best way to reproduce inputs
- $Y = W^T X$ implies $X = W^{-T} Y = W^{-T} W^T X = X$
- In reality, $X = UZV'$, each W preserves some “common” patterns in the training sets
- Think about mapping pixels to “super pixels” and do that recursively

Auto Encoder

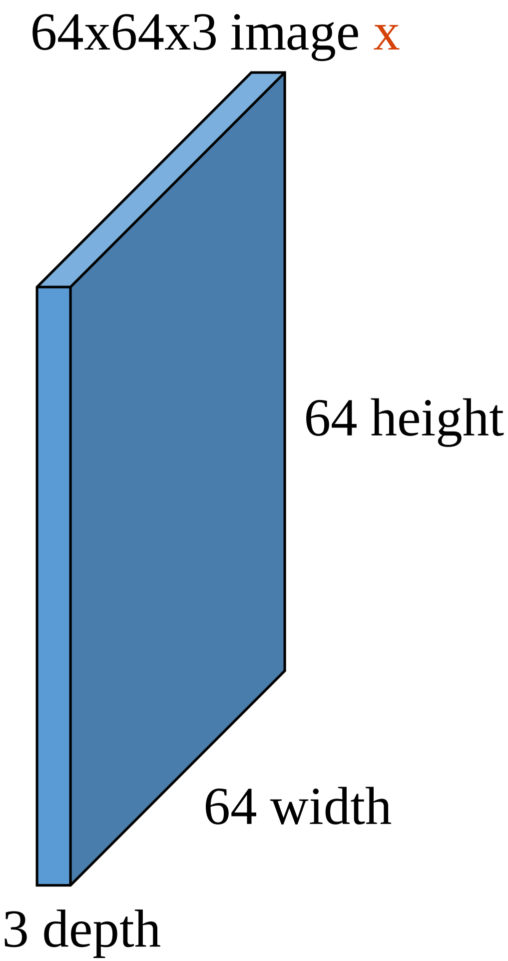
- From individual pixel
- Sample mxn neighborhood from
  - All training images
  - All neighborhoods of mxn
- Form an mxn vector
- SVD of such (mxn)xu matrix captures most common patterns
- Each hidden layer encodes on such pattern
- Auto encoder can best reproduce the input pattern by using transpose weight matrix

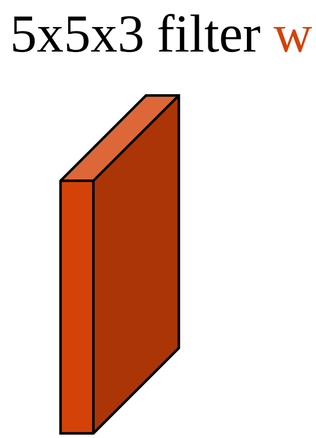
- From individual “super”-pixel
- Sample kxl neighborhood from
  - All training images
  - All mxn super pixel neighborhood
- Form an kxl vector
- SVD of such (kxl)xu matrix captures most common patterns
- Each hidden layer encodes on such pattern
- Auto encoder can best reproduce the input pattern by using transpose weight matrix

- From Pixel
  - Collect all mxn patterns in all training images and all neighbors
  - Linearize into mxn vectors
  - Find

- From Superpixel
  - Collect all mxn superpixel patterns

Convolution

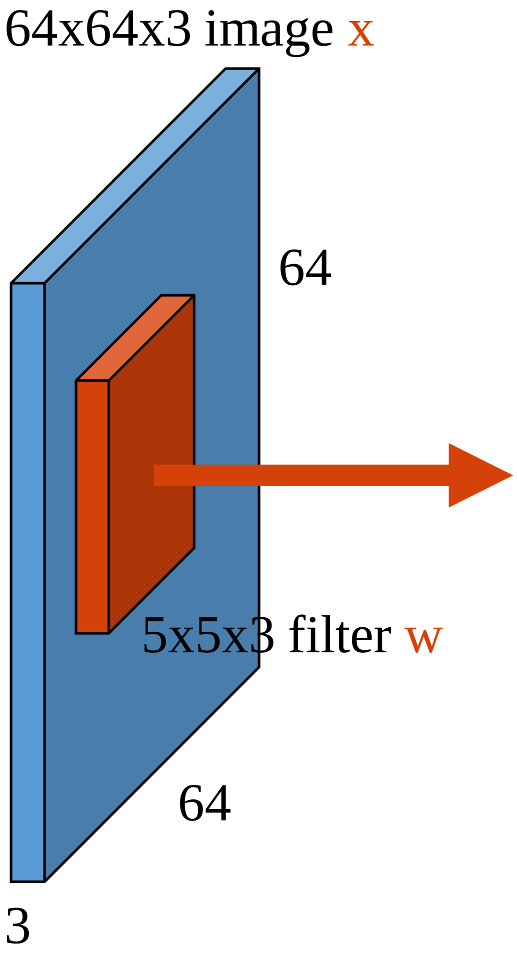




Convolve the filter with the image

"Slide over the image spatially, computing dot products"

Convolution

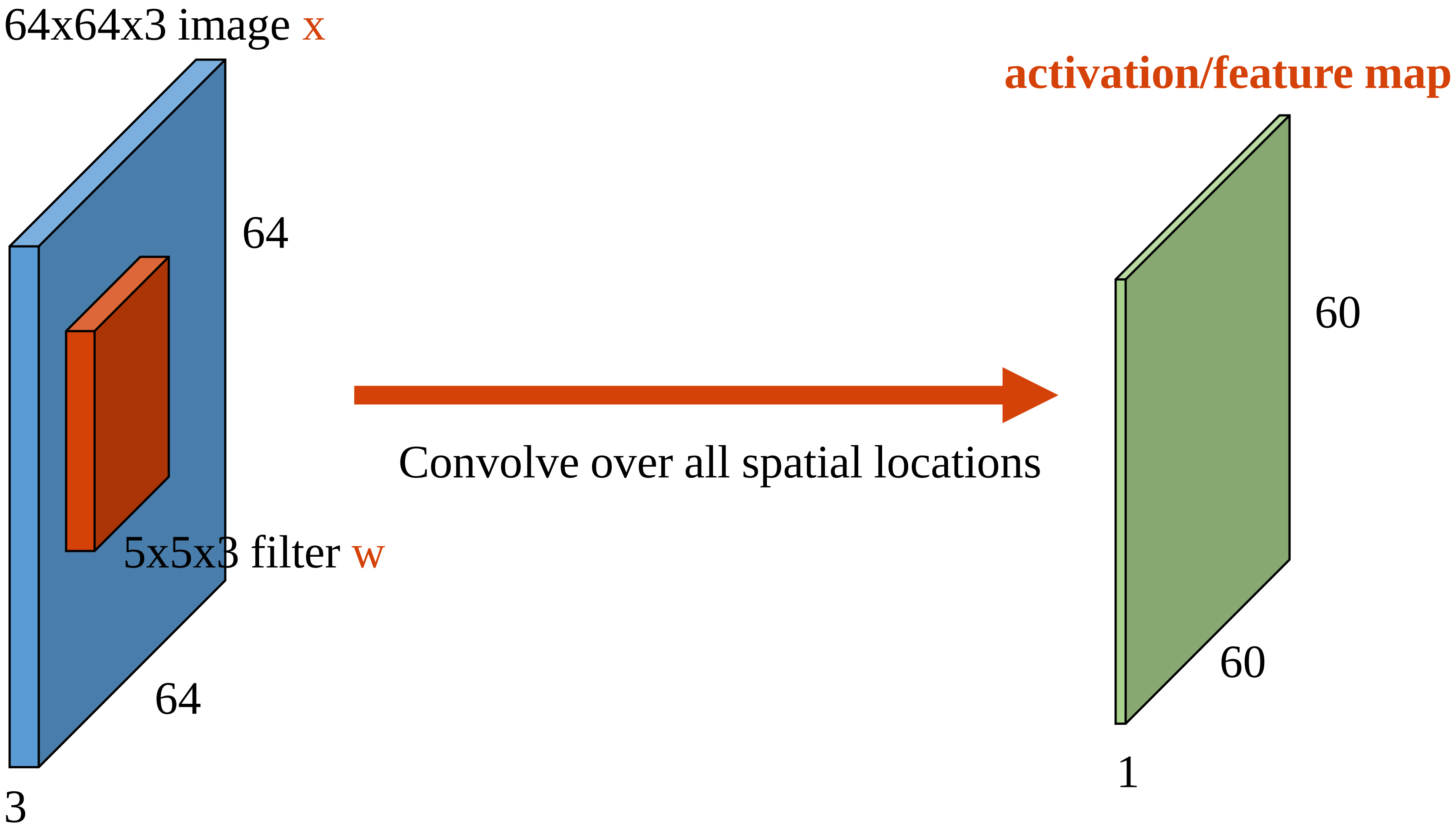


1 number:

the result of taking a dot product between the filter and a small 5x5x3 chunk of the image.

$w^T x + b$

Convolution



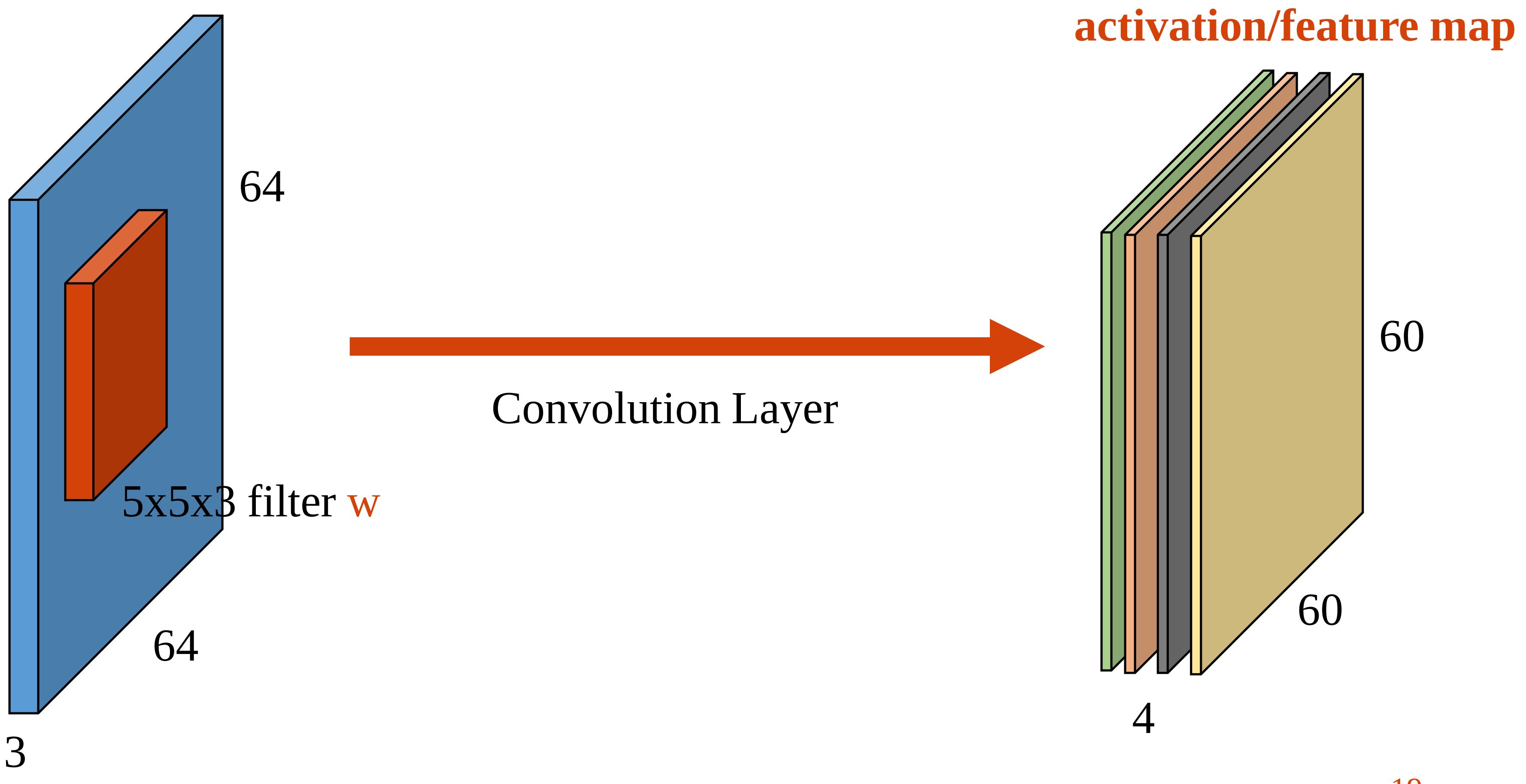
Convolution - Intuition

	0	1	0	
	1	-4	1	
	0	1	0	

- Intuition of convolution
  - Similarity measurement: image chunk and convolutional filter.
  - Looking for local patterns.
  - Patterns are not fixed and will be learned during training.

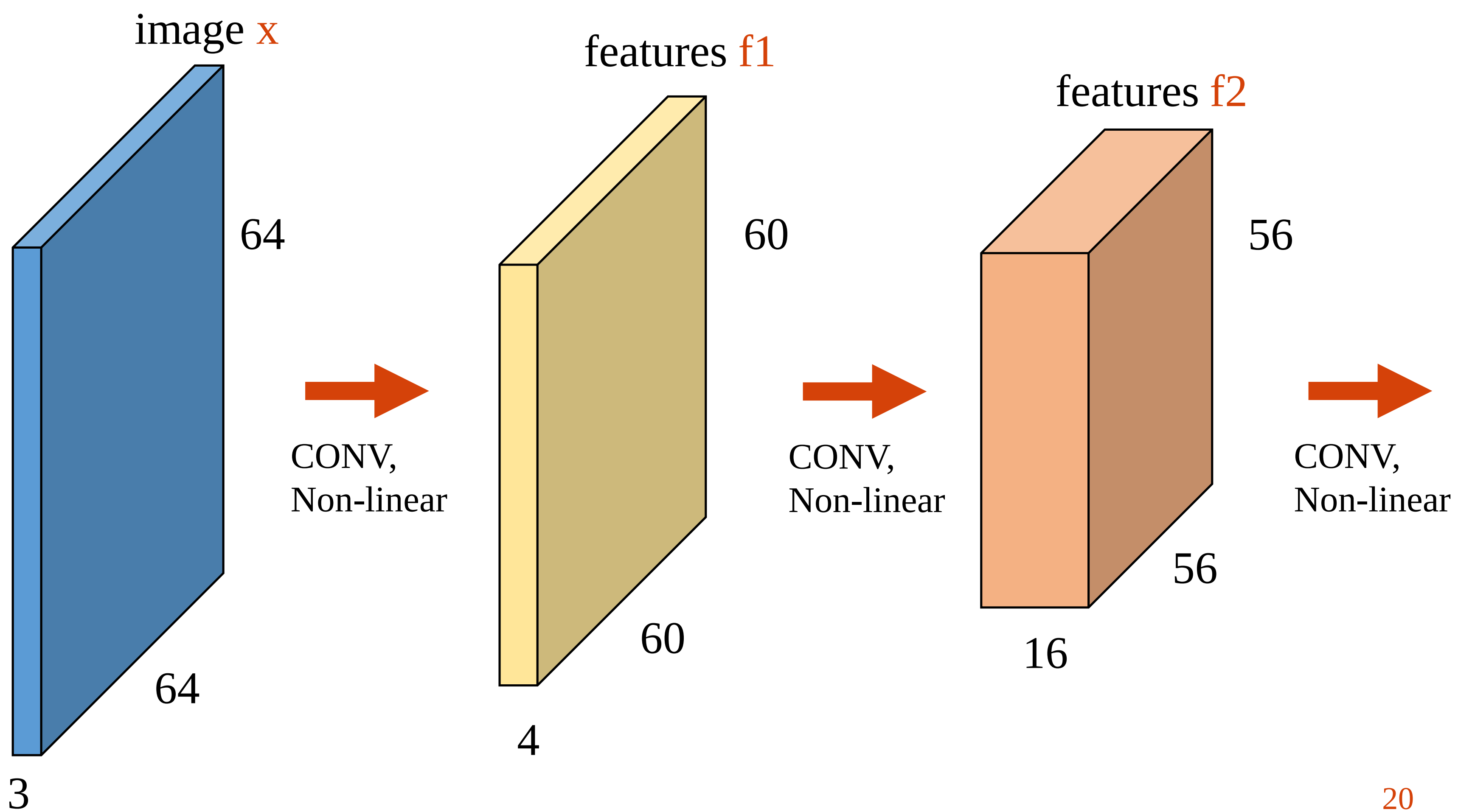
Convolutional Layer

- Stacking multiple convolutions in one layer



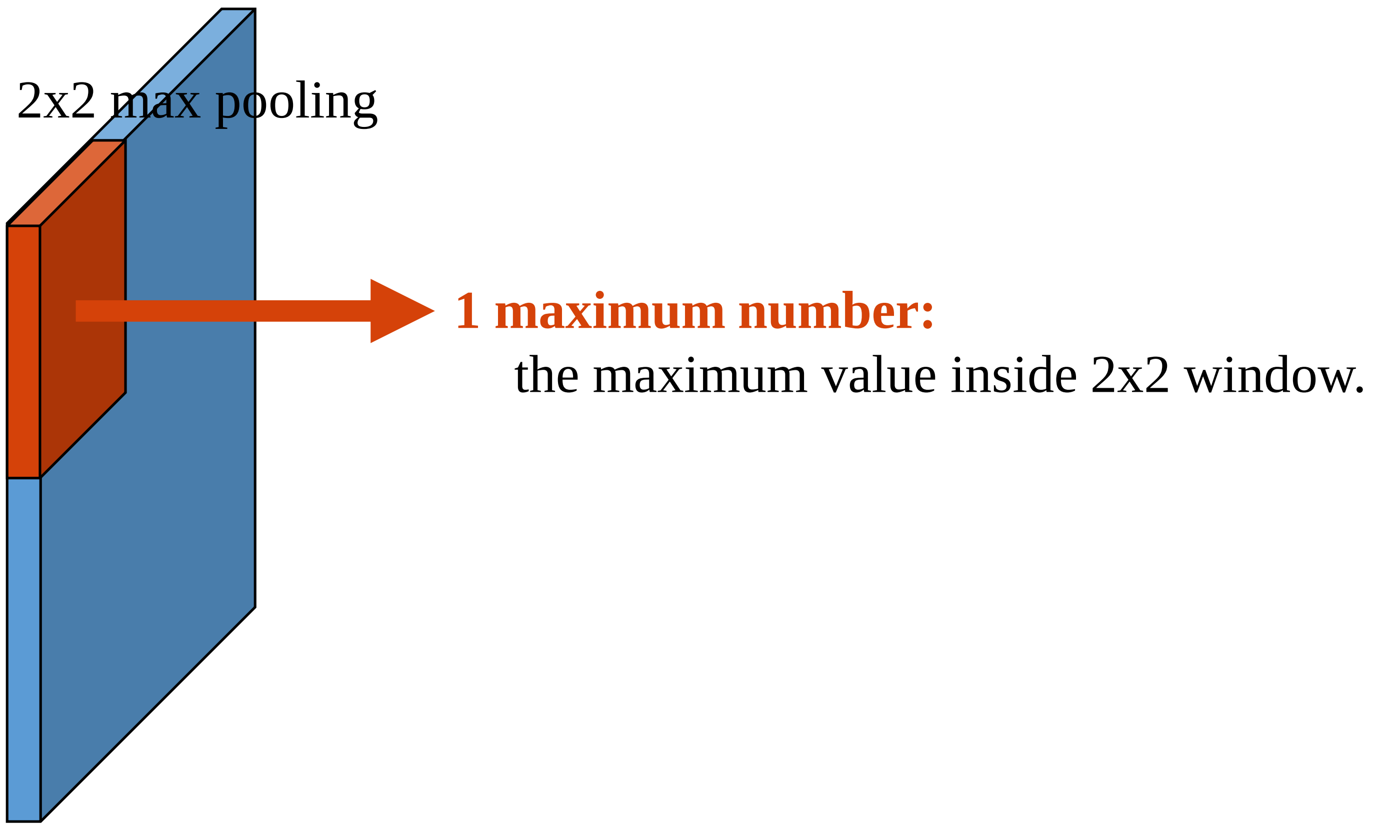
Convolutional Feature Maps

❑ A feature hierarchy by stacking convolutional layers



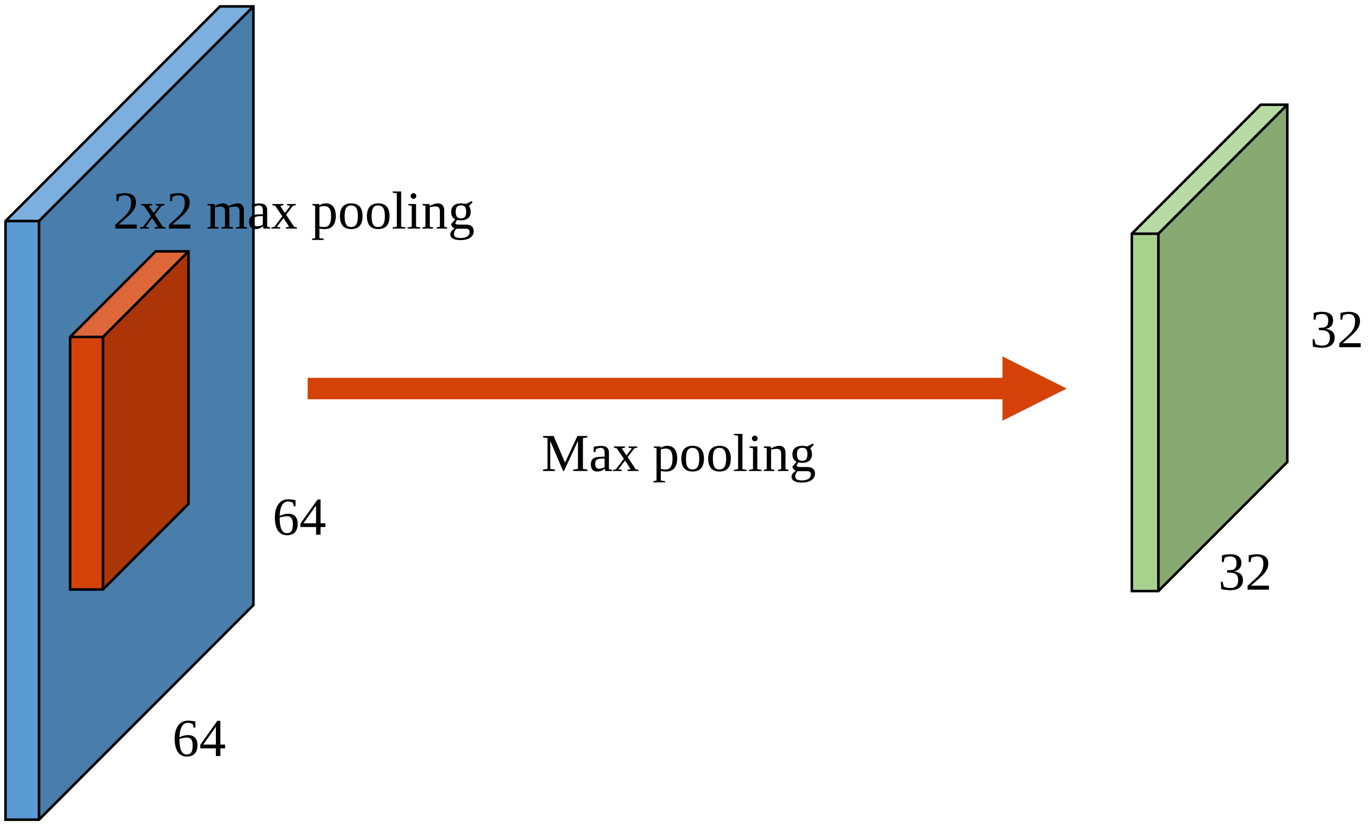
Max Pooling

- Max pooling operation
  - Slides a small non-overlapping window.
  - Picks the maximum value inside each window.



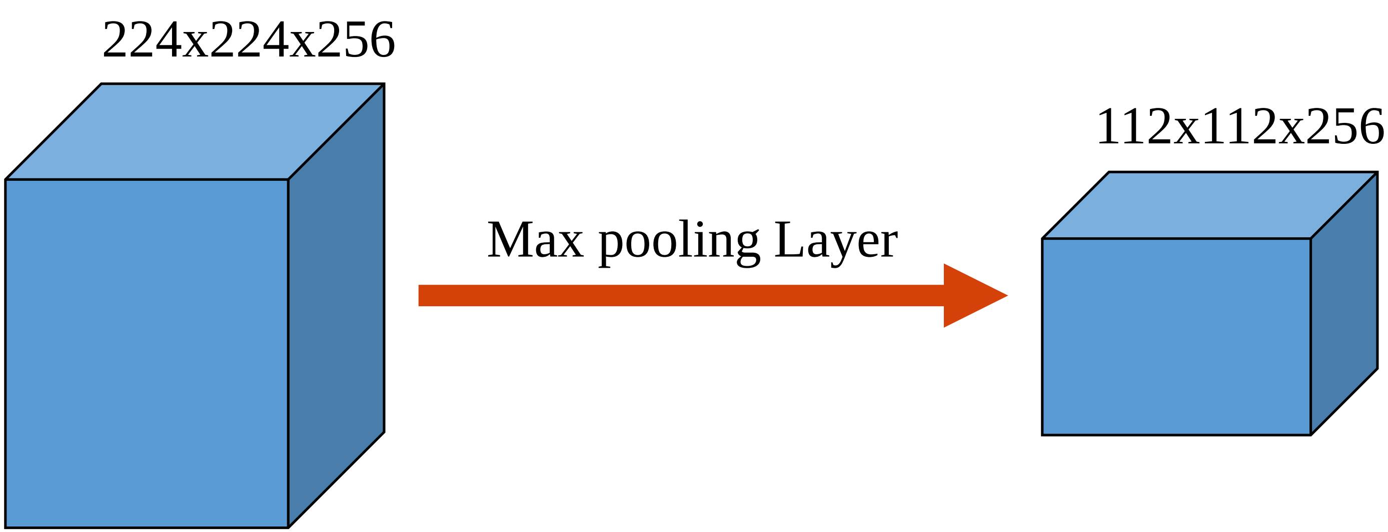
Max Pooling

- Max pooling operation
  - Slides a small non-overlapping window.
  - Picks the maximum value inside each window.



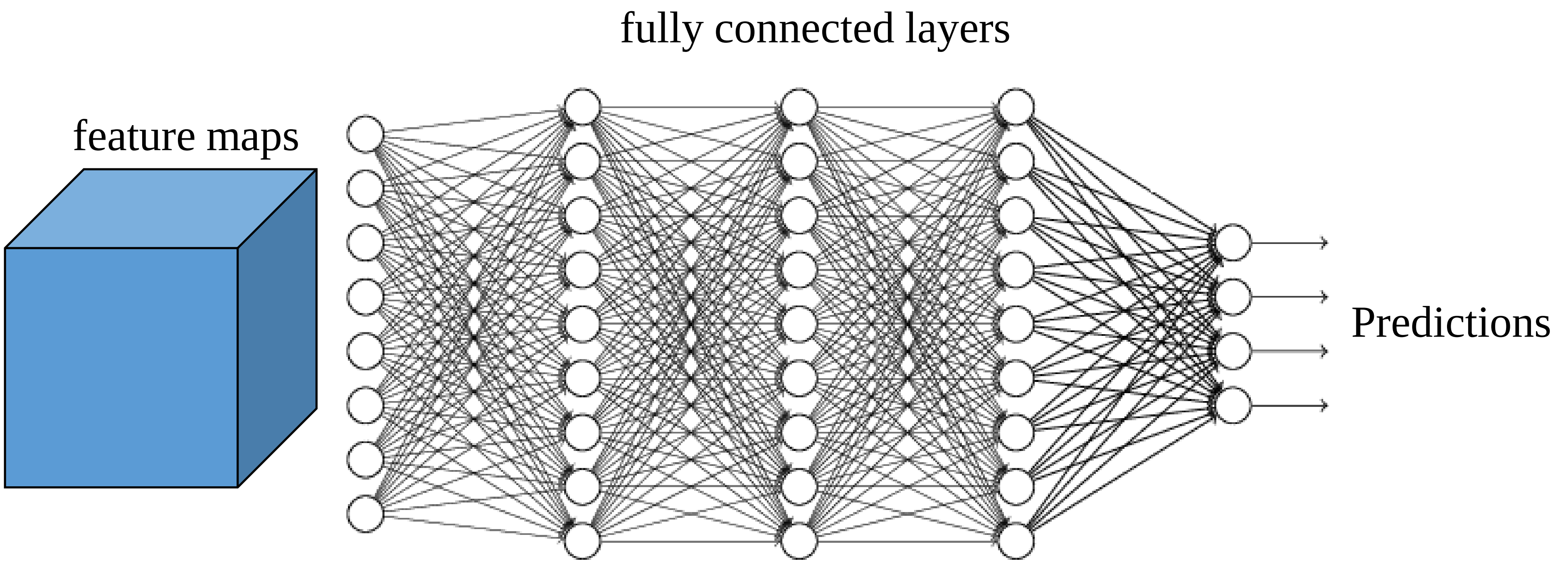
Max Pooling Layer

- Advantages
  - Reduces the redundancy of convolutions.
  - Makes the representations smaller and more manageable.
  - Reduces the number of parameters, controls overfitting.
  - Invariant to small transformations, distortions and transitions.



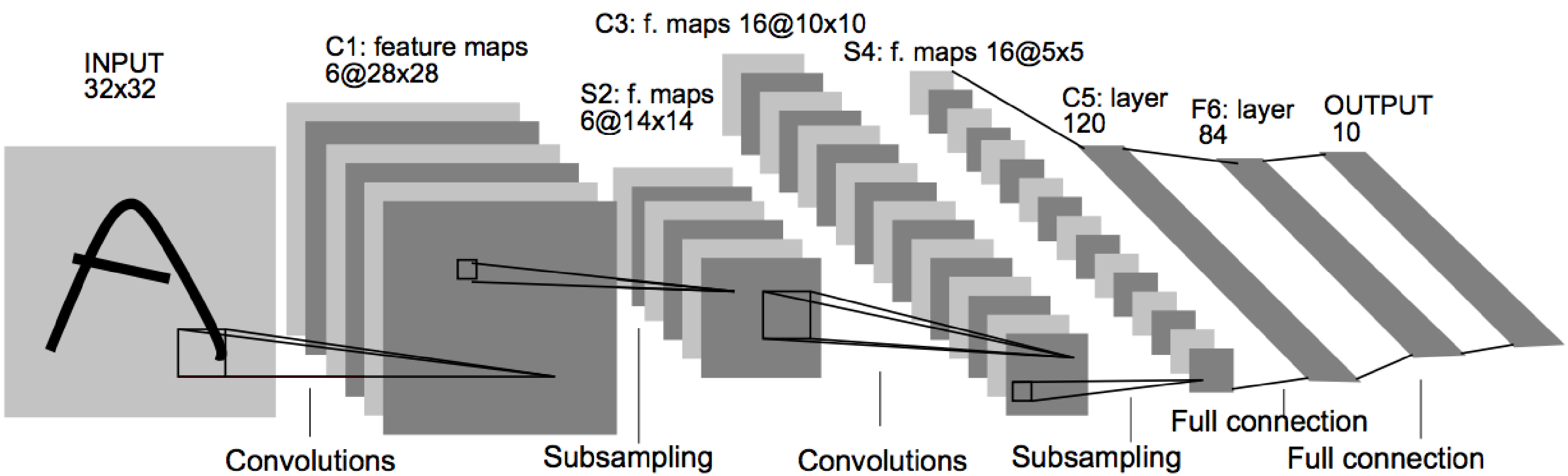
Fully Connected Layer

- A classifier on top of feature maps
  - Maps high-dimensional matrix to predictions (1-D vector)



Convolutional Neural Network

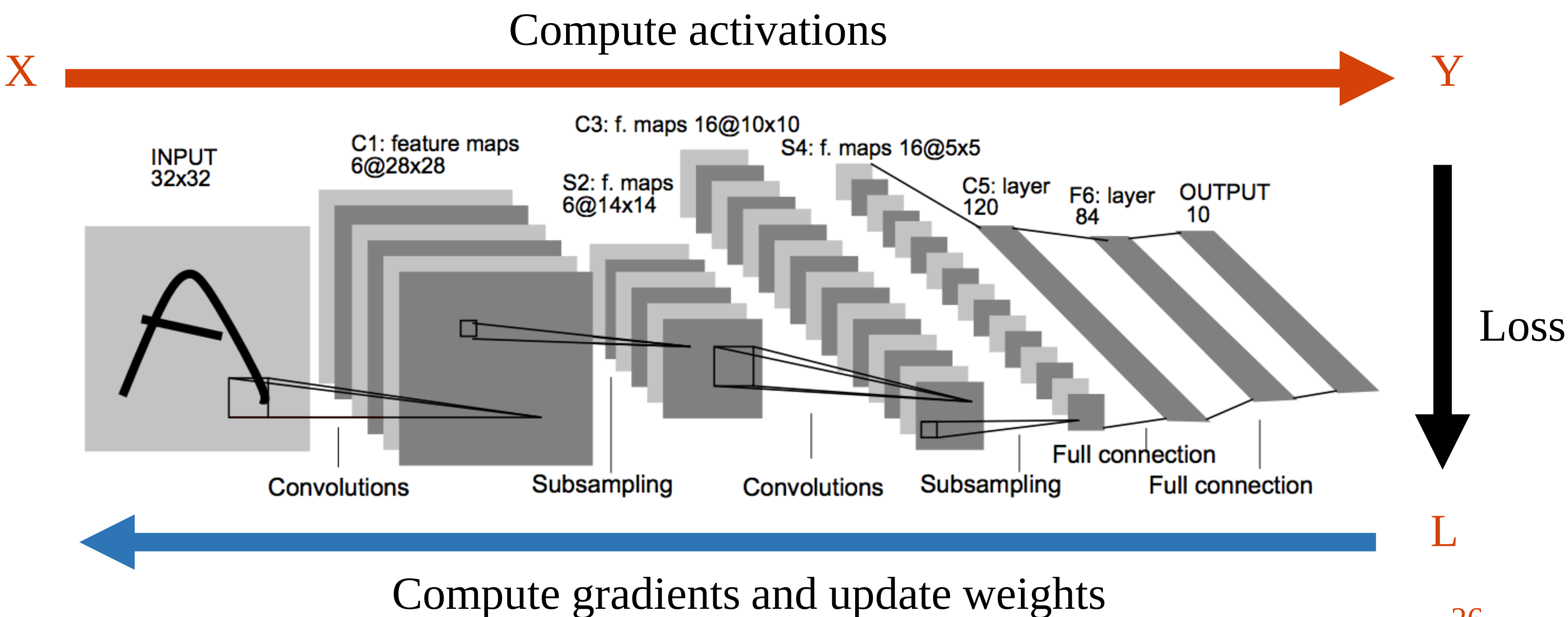
- First CNN trained with backpropagation
  - Three different layers: convolution, subsampling, fully-connected
  - Training CNN: gradient-decent optimization (backpropagation)



[1] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.

Training Convolution Neural Network

- Gradient Descent
  - Randomly initialize weights at each layer.
  - Compute a forward pass and calculate the cost function.
  - Calculate the with respect to each weight during a backward pass.
  - Update weights.



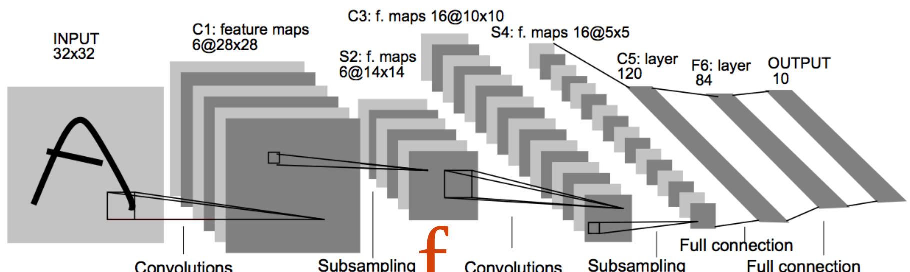
Traditional vs. Deep Learning

Feature Descriptor

f

Predictions

Training



f

Predictions

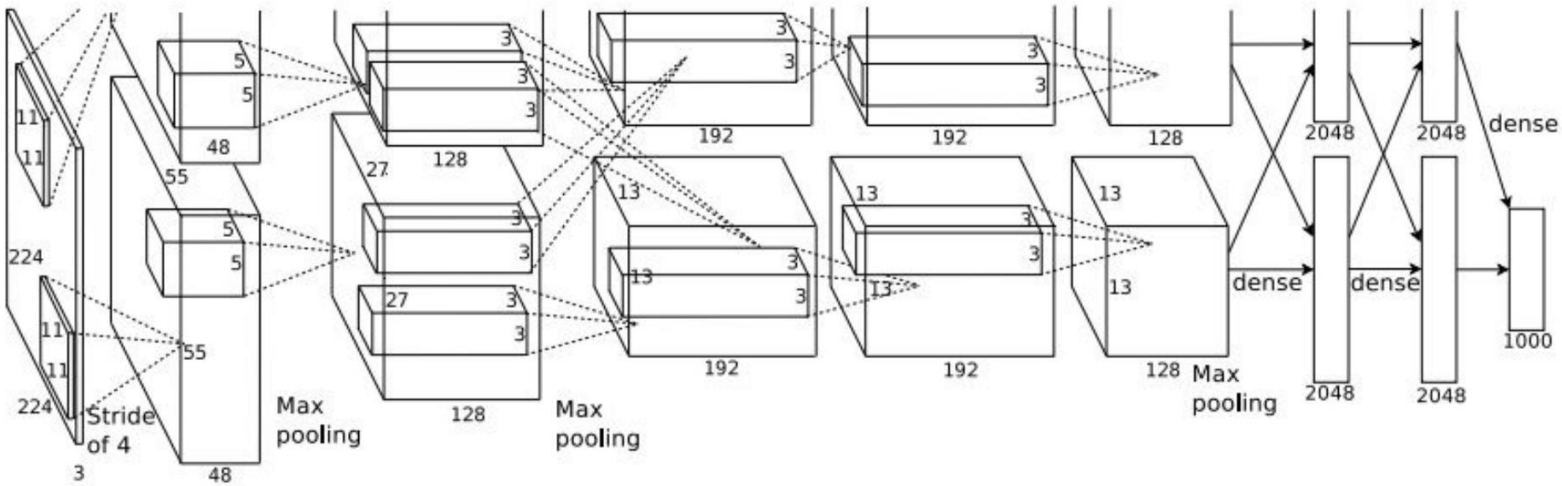
Training

Roadmap

- Introduction
- Convolutional Neural Network
- Application
  - Image Classification
  - Object Detection
  - Object Tracking
- Our Work
- Conclusion and Discussion

AlexNet

- 2012 IMAGENET classification task winner

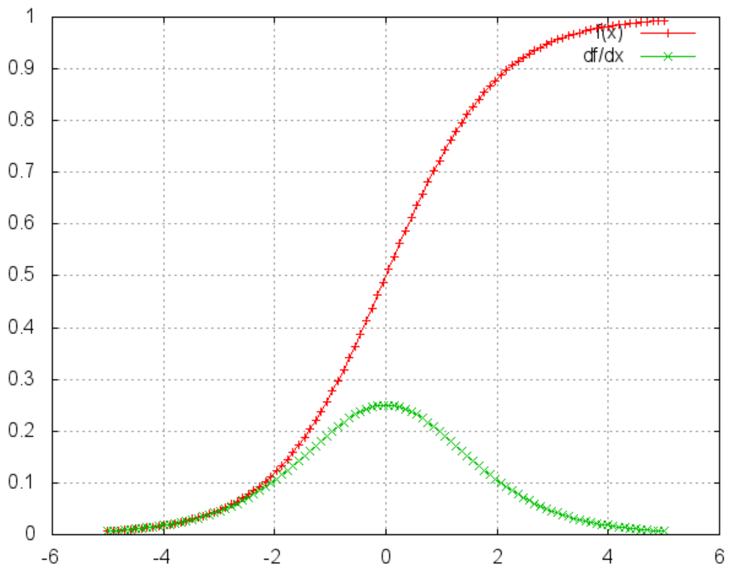


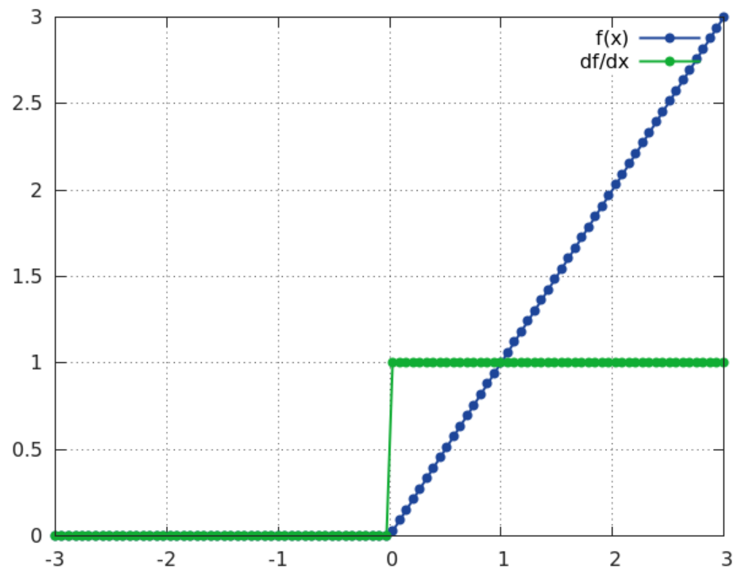
- Advances in AlexNet
  - ReLU activation function
  - Dropout regularization
  - GPU Implementation and dataset benchmark

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.

Rectified Linear Unit (ReLU)

- Calculate gradient with backpropagation

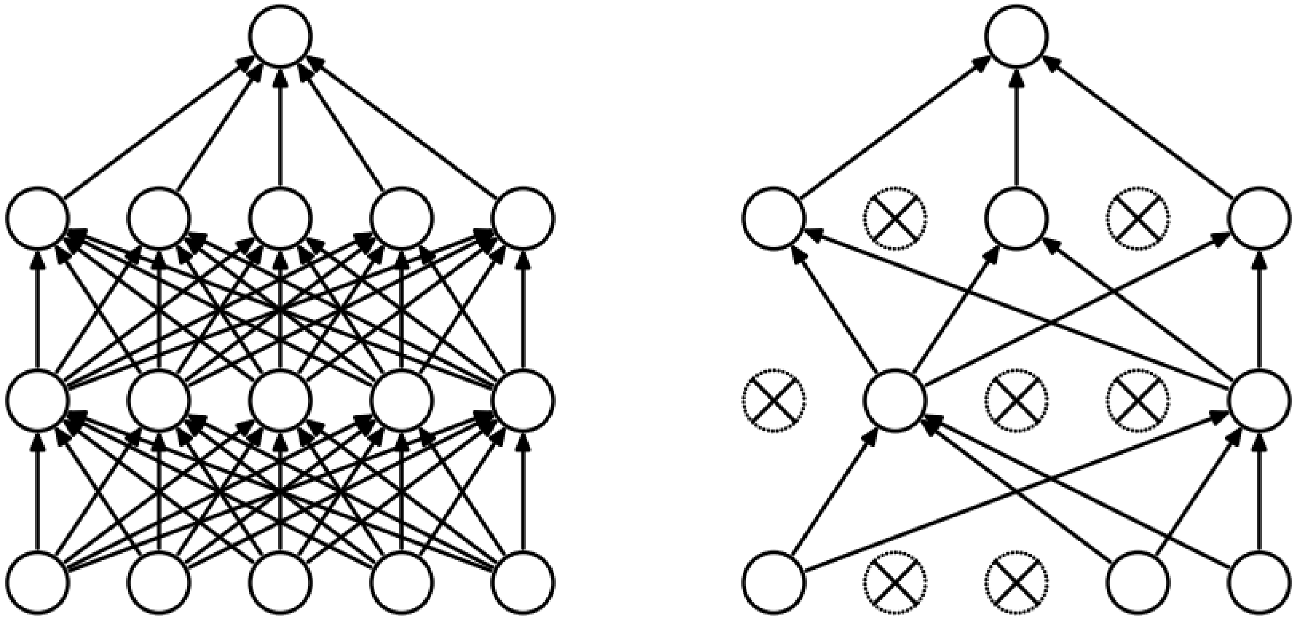




- Advantages of ReLU
  - Fast convergence.
  - Better for image data.

Dropout Regularization

- Randomly set some neurons to zero in the forward pass.



- Interpretation
  - Hidden units cannot co-adapt to other units.
  - Hidden units must be more generally useful.

Benchmark and GPU Computation

- ❑Dataset benchmarking

IMAGENET

COCO Common Objects in Context

VOT visual object tracking challenge

- ❑GPU Computation

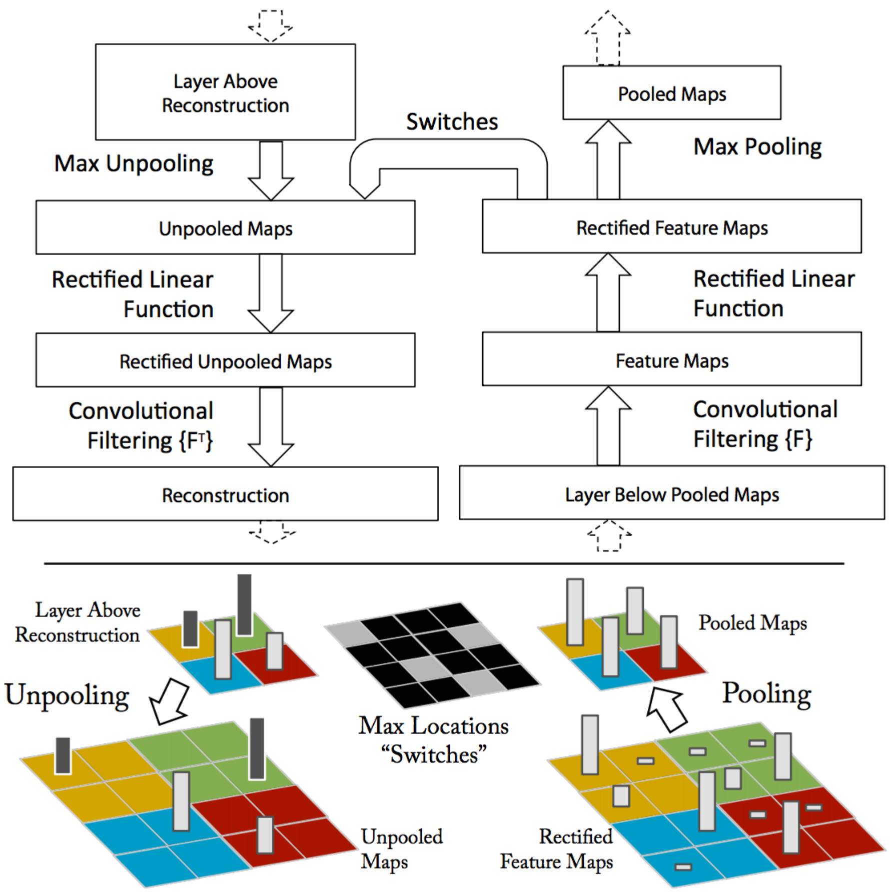
NVIDIA CUDA

Visualizing Convolutional Networks

- **Deconvolutional Network**

  Map activations back to the input pixel space, show what input pattern originally caused a given activation.

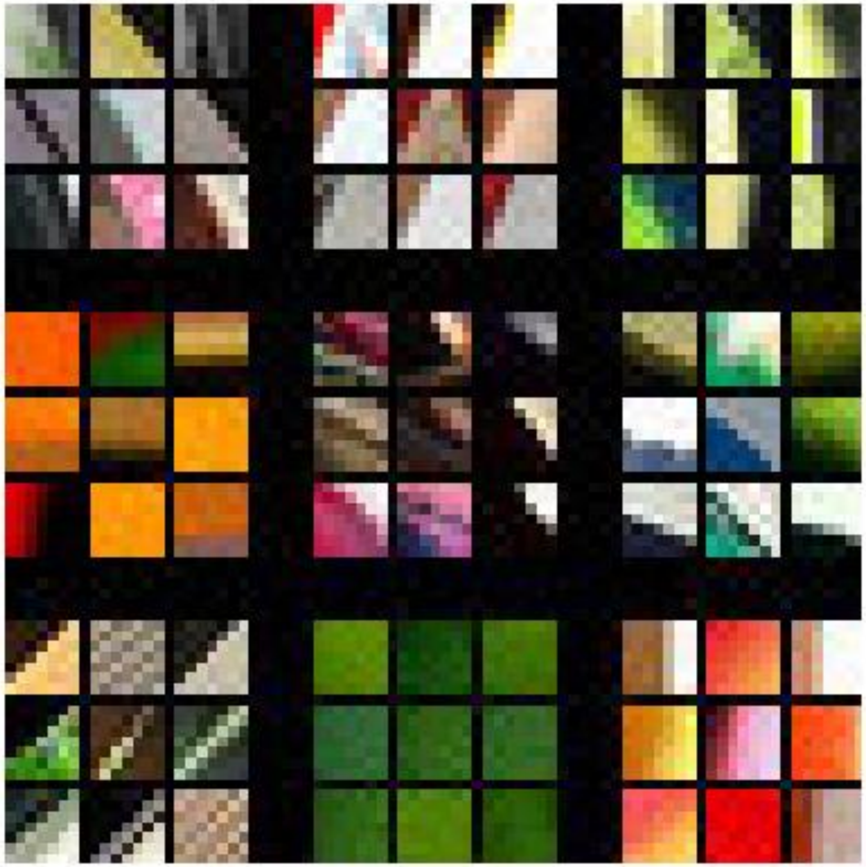
- **Steps**
  - Compute activations at a specific layer.
  - Keep one activation and set all others to zero.
  - Unpooling and deconvolution
  - Construct input image



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Visualizing Convolutional Networks

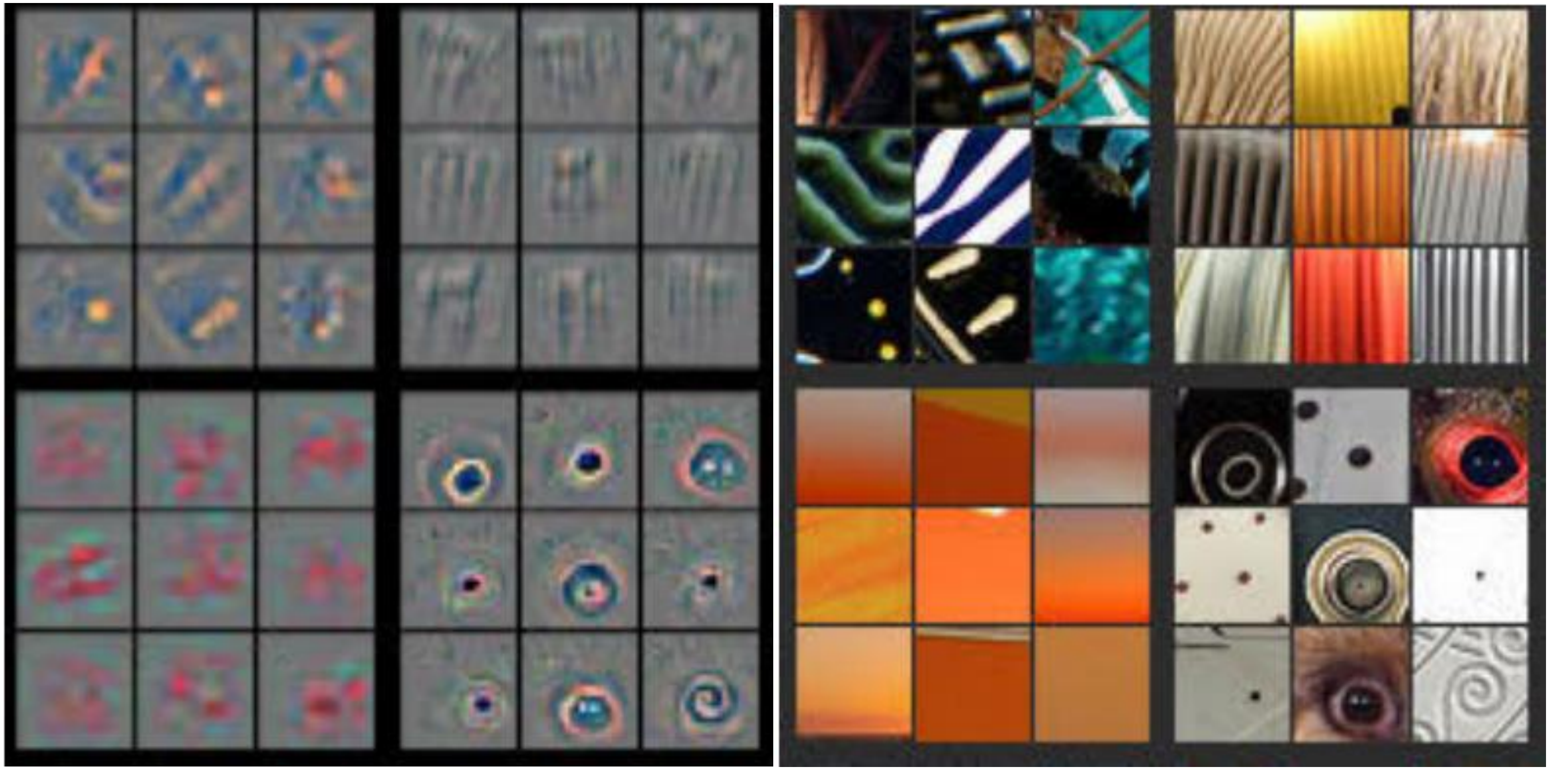
- Layer1



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Visualizing Convolutional Networks

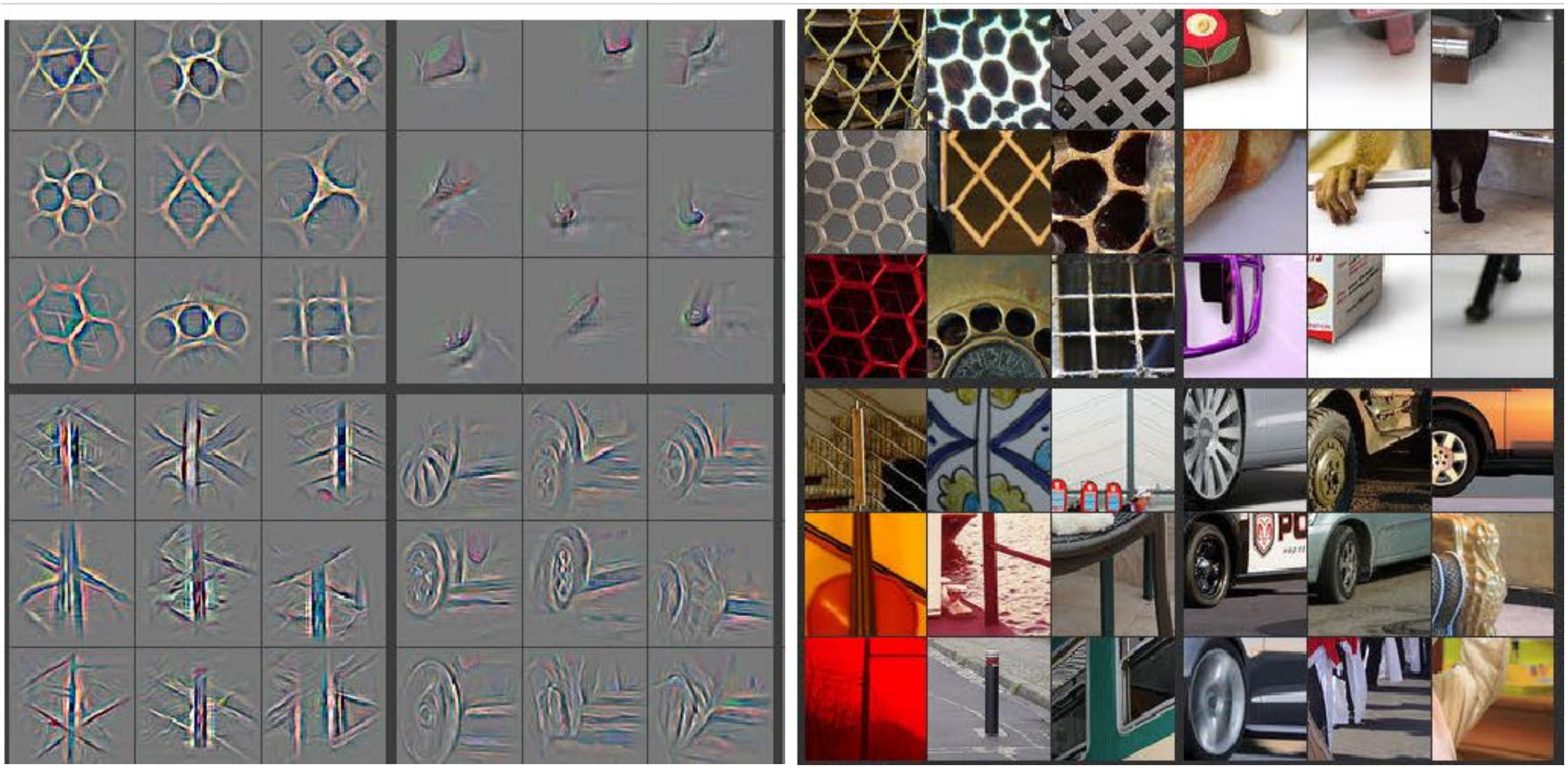
- Layer2



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Visualizing Convolutional Networks

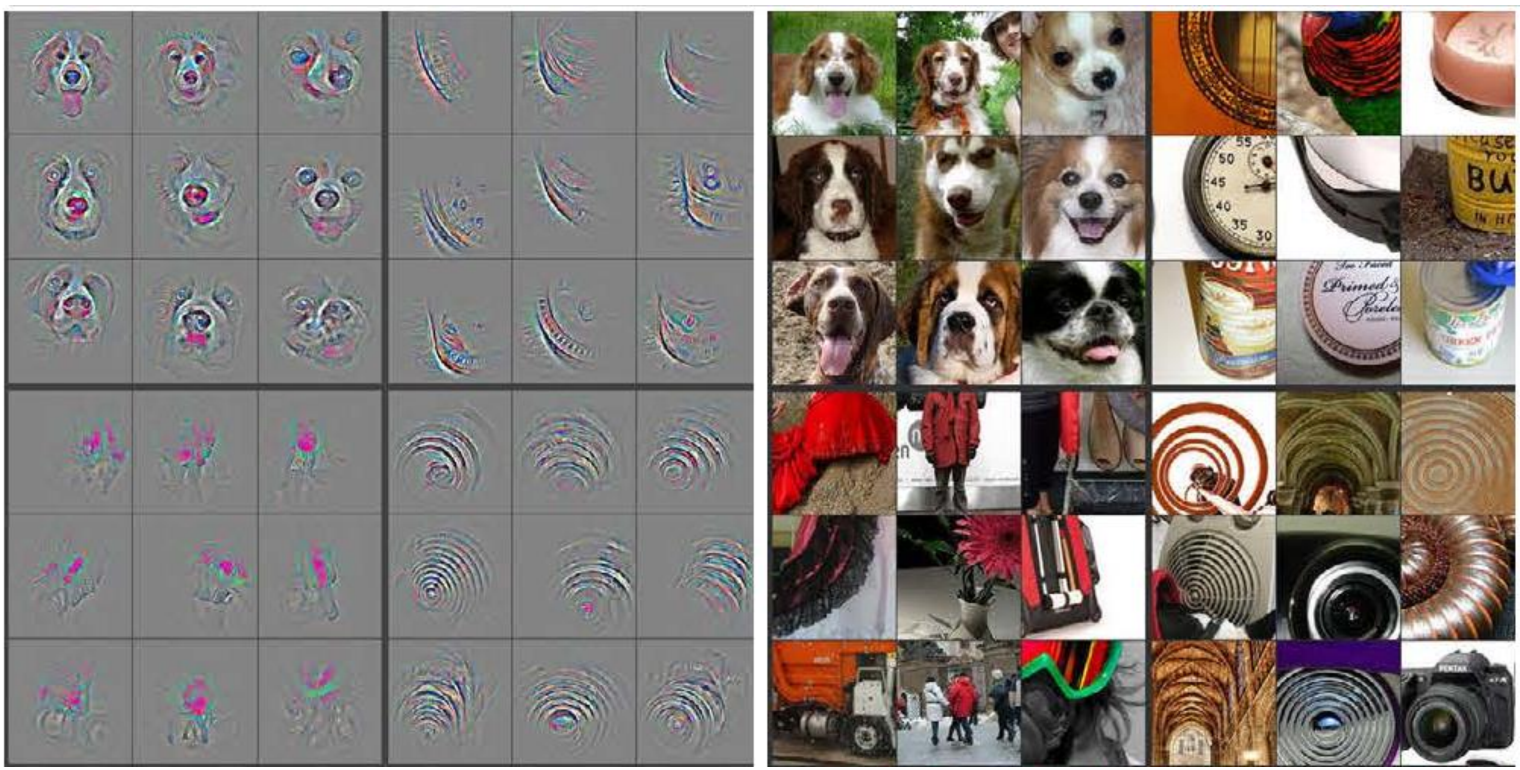
- Layer3



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Visualizing Convolutional Networks

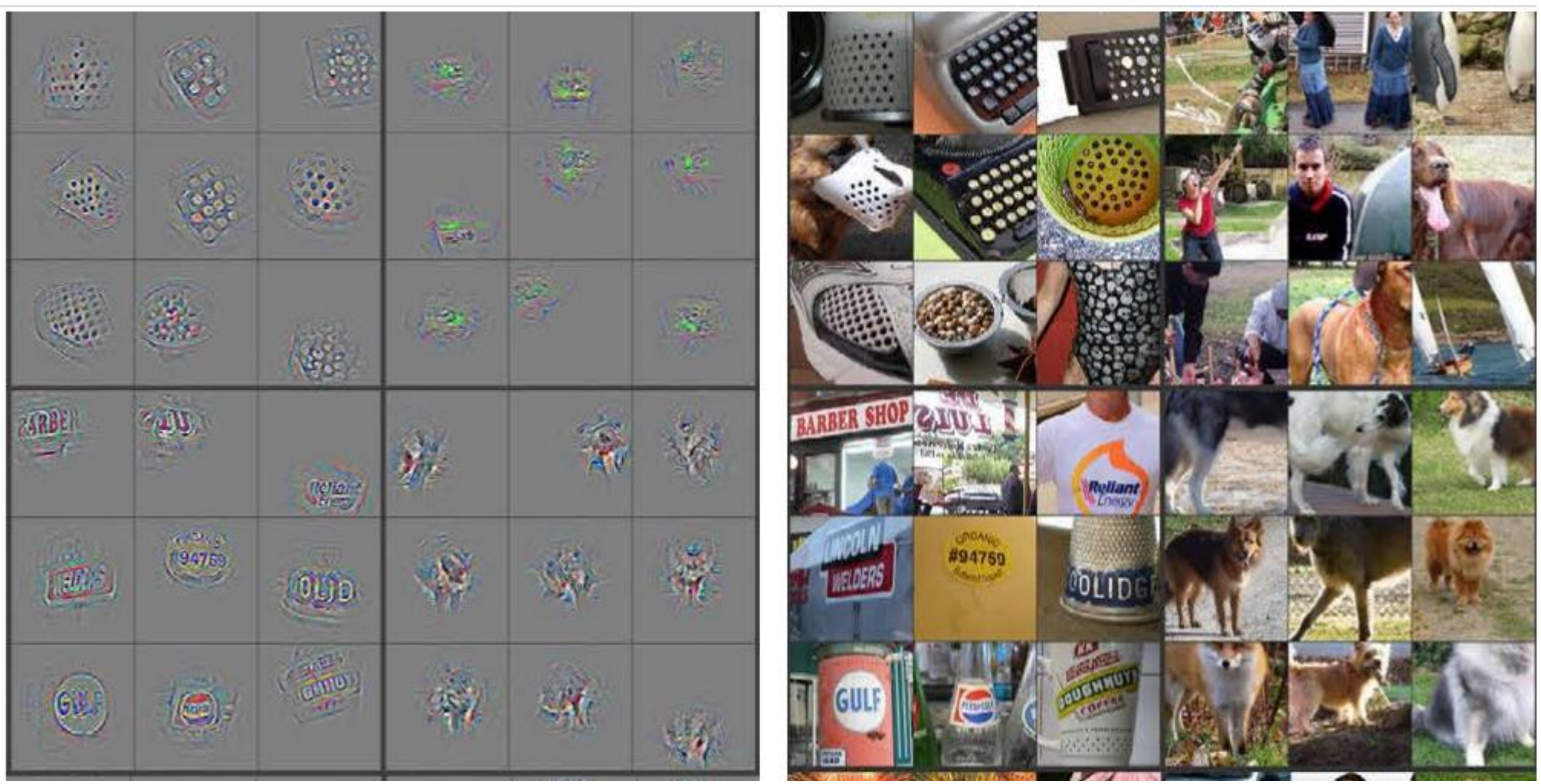
- Layer4



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Visualizing Convolutional Networks

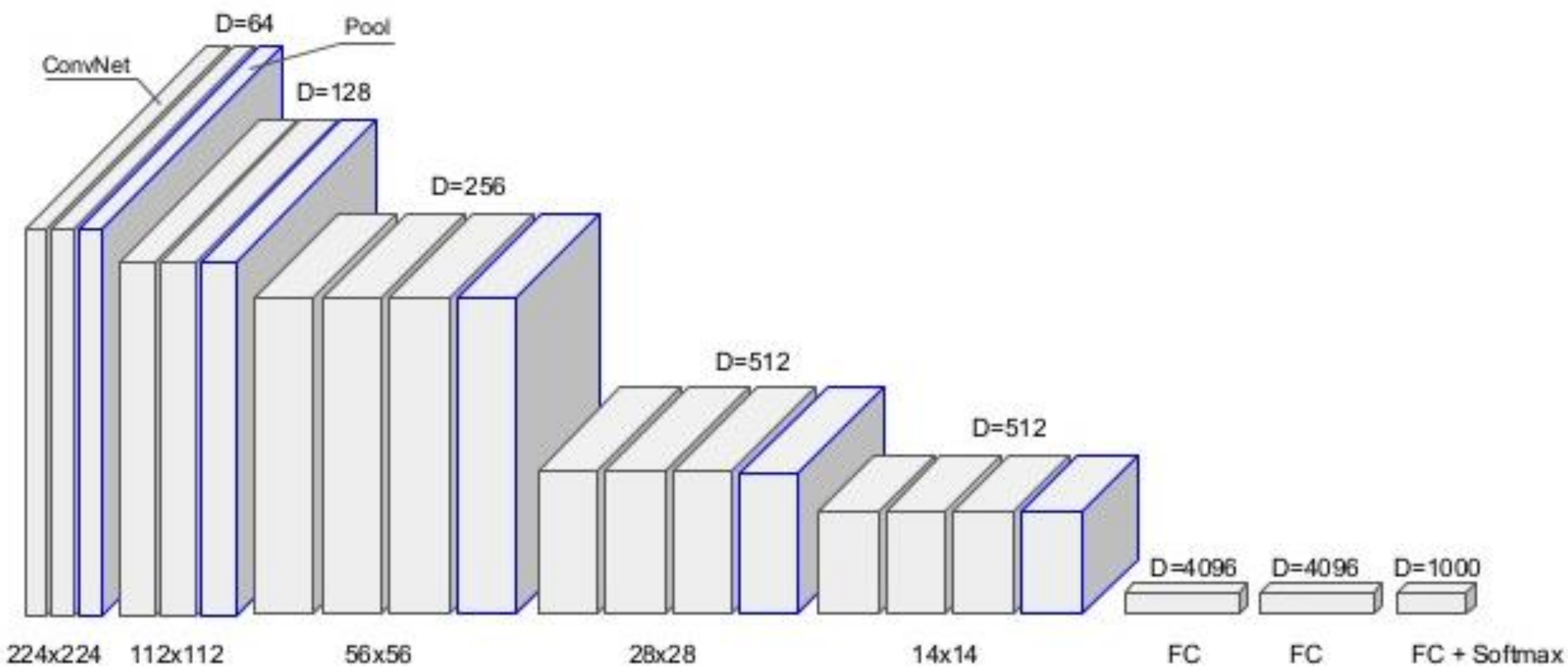
- Layer5



[1] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European Conference on Computer Vision*. Springer International Publishing, 2014.

VGGNet

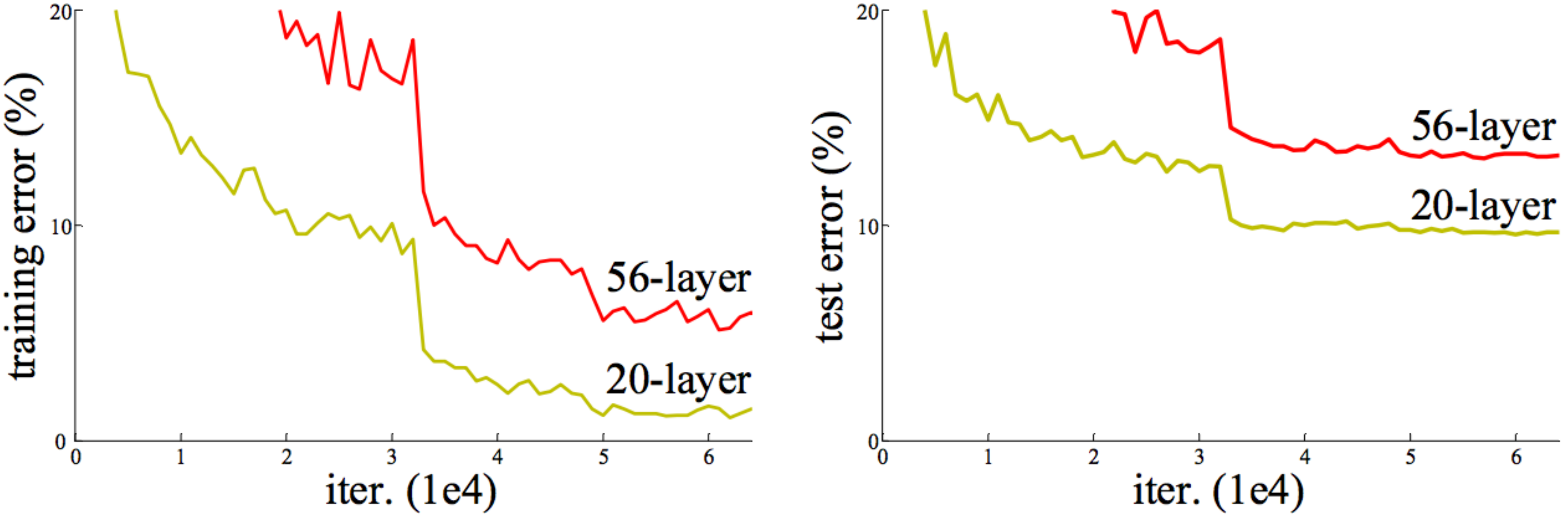
- ❑ Characteristics
  - ❑ Much deeper
  - ❑ Only 3x3 CONV and 2x2 MAX POOL are used



[1] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556*(2014).

ResNet

- Problems with stacking more layers

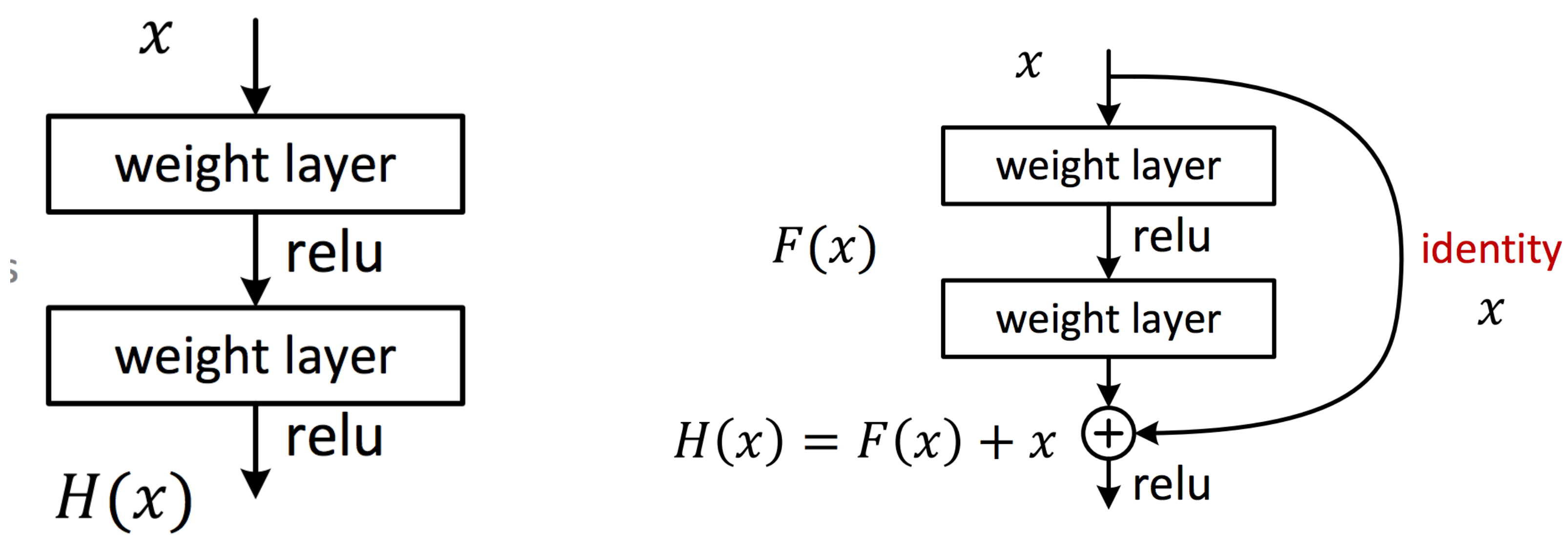


Optimization Problem

[1] He, Kaiming, et al. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385* (2015).

ResNet

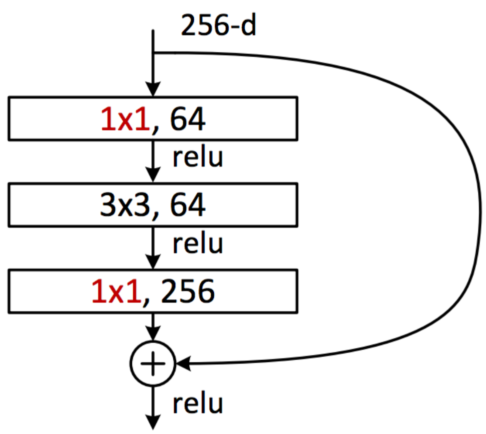
- Train weight layers to fit F(x) rather than H(x)

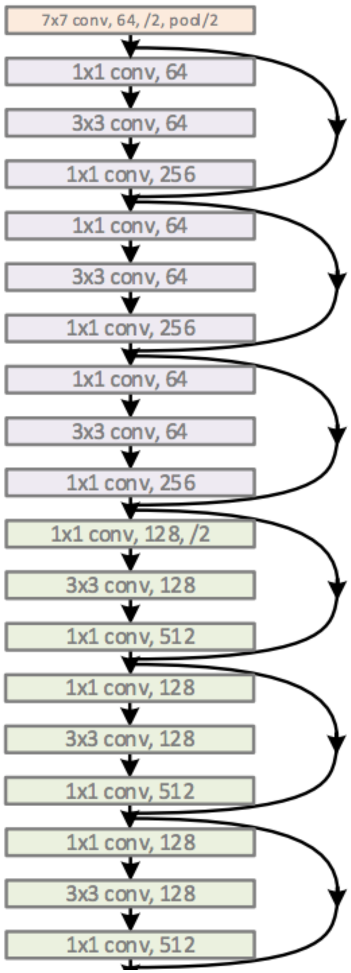


- Easy to find identity mapping and small fluctuations.

[1] He, Kaiming, et al. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385* (2015).

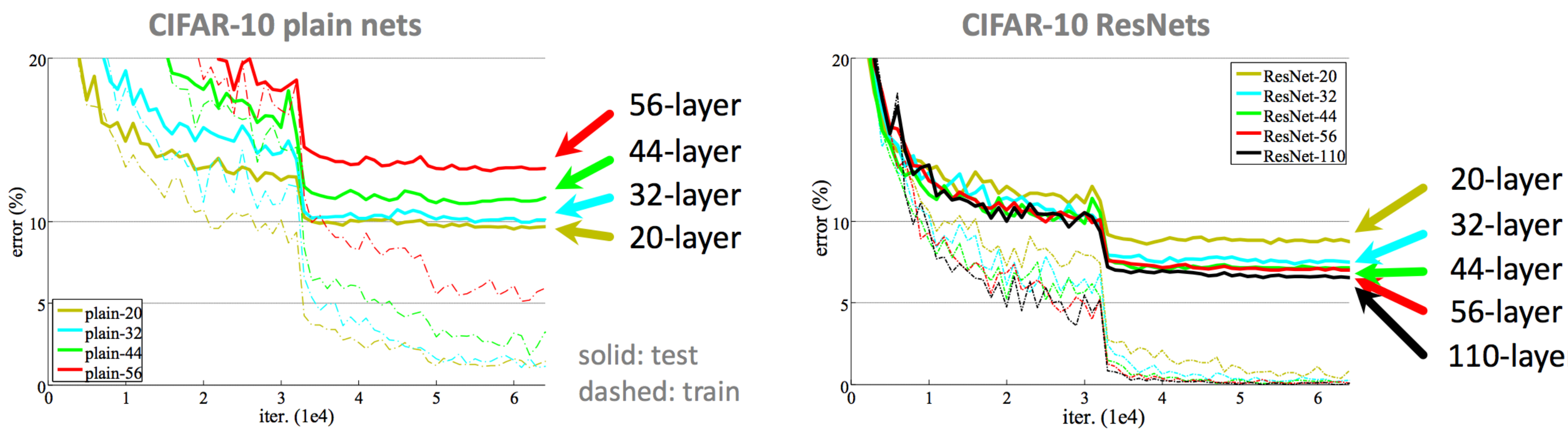
ResNet





[1] He, Kaiming, et al. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385* (2015).

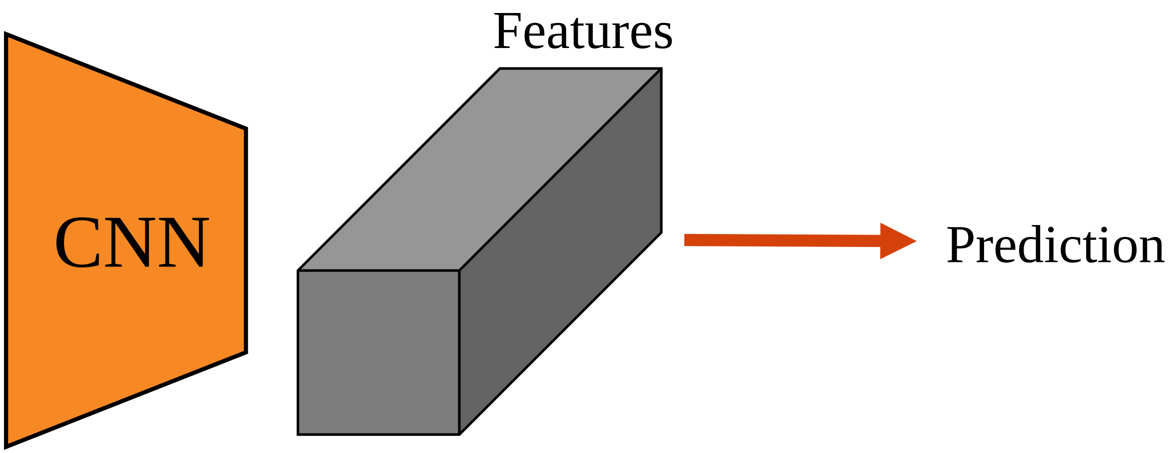
ResNet



[1] He, Kaiming, et al. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385* (2015).

Evolution of CNN

- Evolutional trend
  - Performance getting better
  - Architecture goes deeper
  - Design becomes simpler



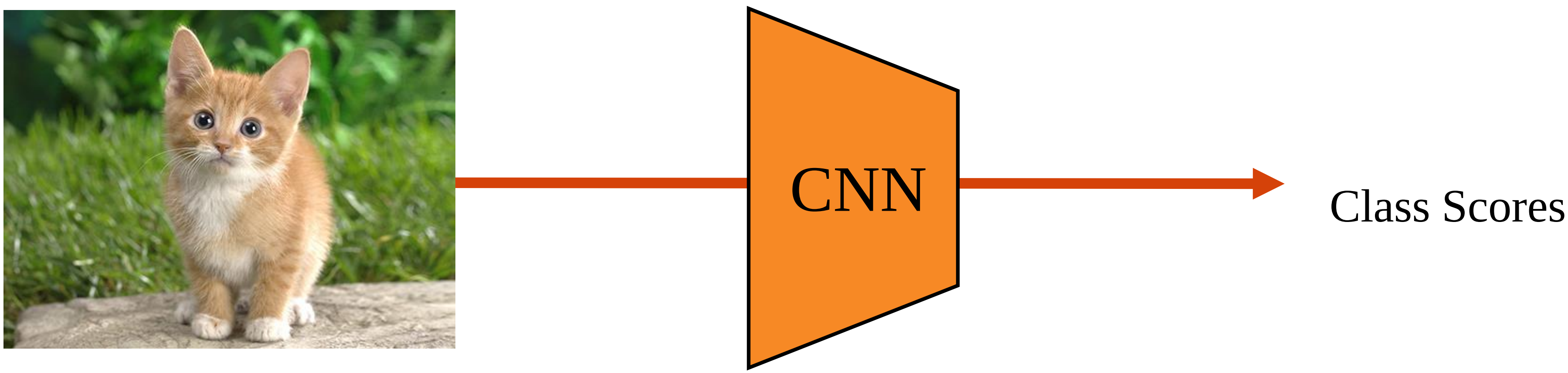
- CNN is the basic building block for recent visual system
  - Hierarchical feature descriptor.
  - Designed to solve different recognition problems.

Roadmap

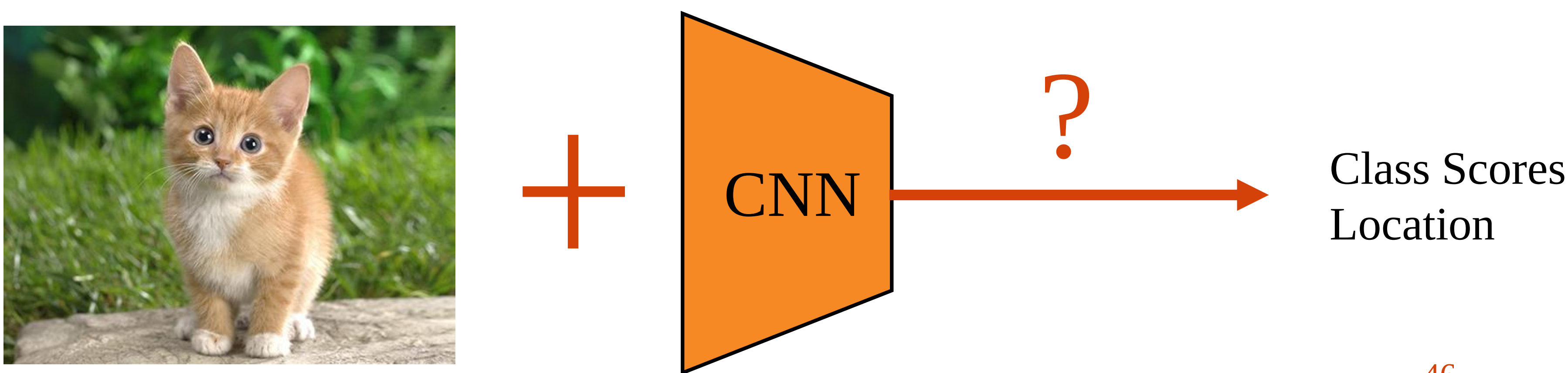
- Introduction
- Convolutional Neural Network
- Application
  - Image Classification
  - Object Detection
  - Object Tracking
- Our Work
- Conclusion and Discussion

Object Detection

- ❑ Image Classification



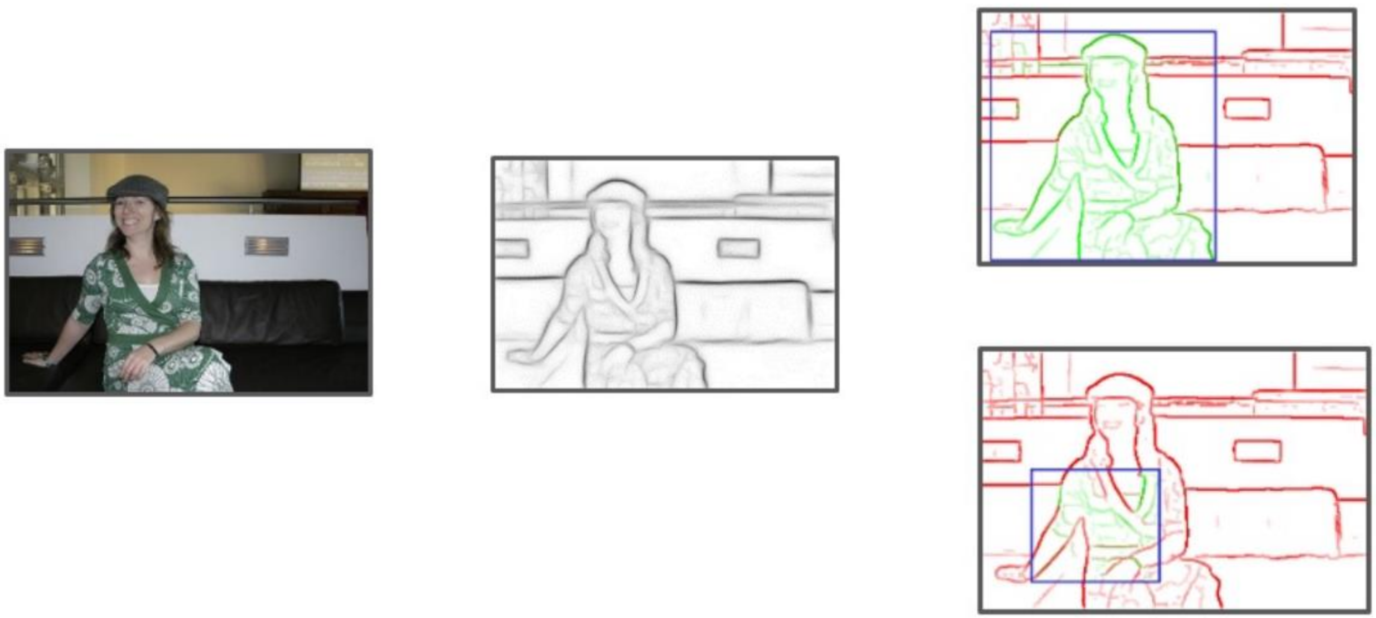
- ❑ Object Detection



Region Proposal

- Find potential regions that might have objects.
- Example: Edge Boxes

  The number of contours that are wholly contained in a bounding box is indicative of the likelihood of the box containing an object

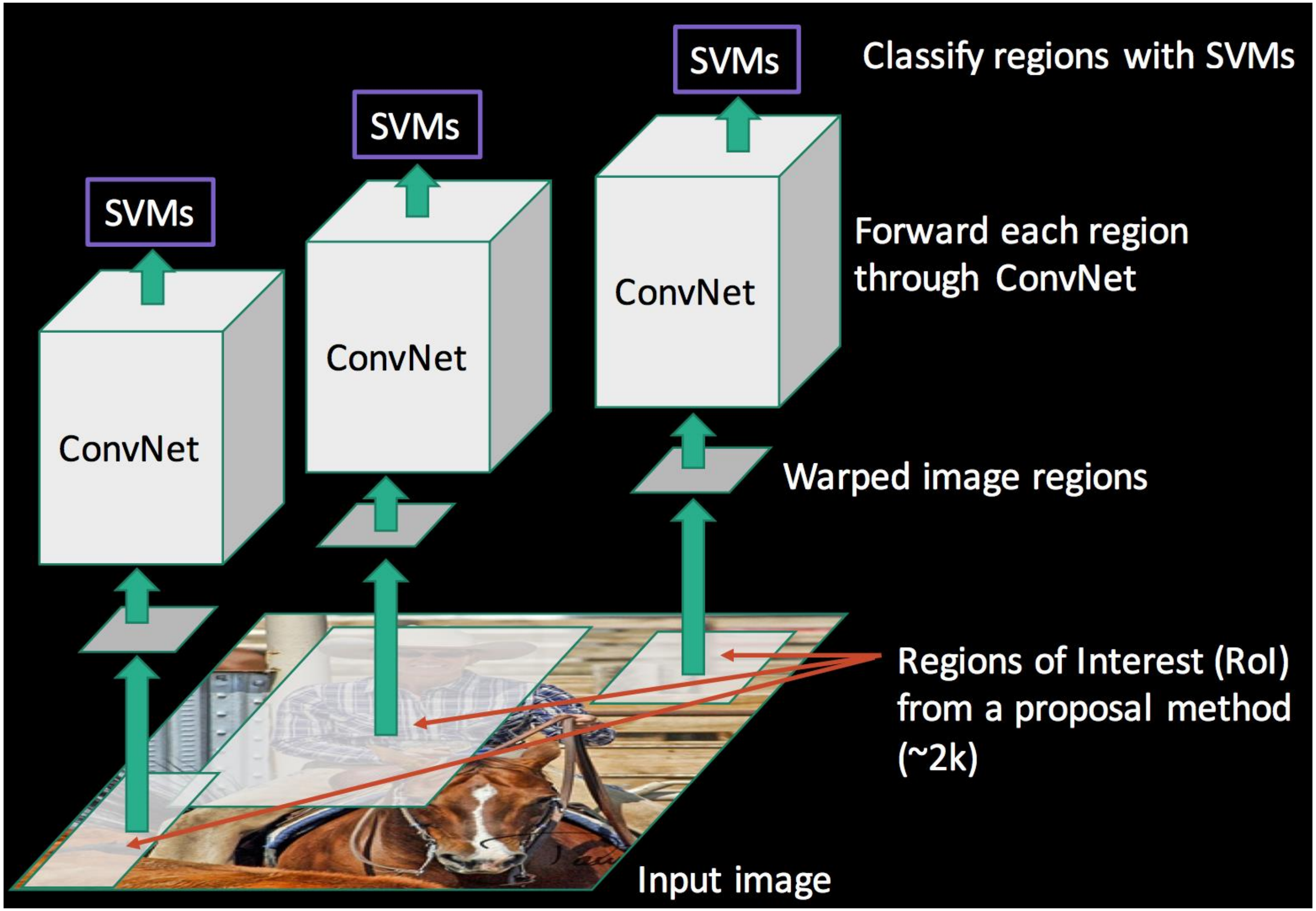


Good

Bad

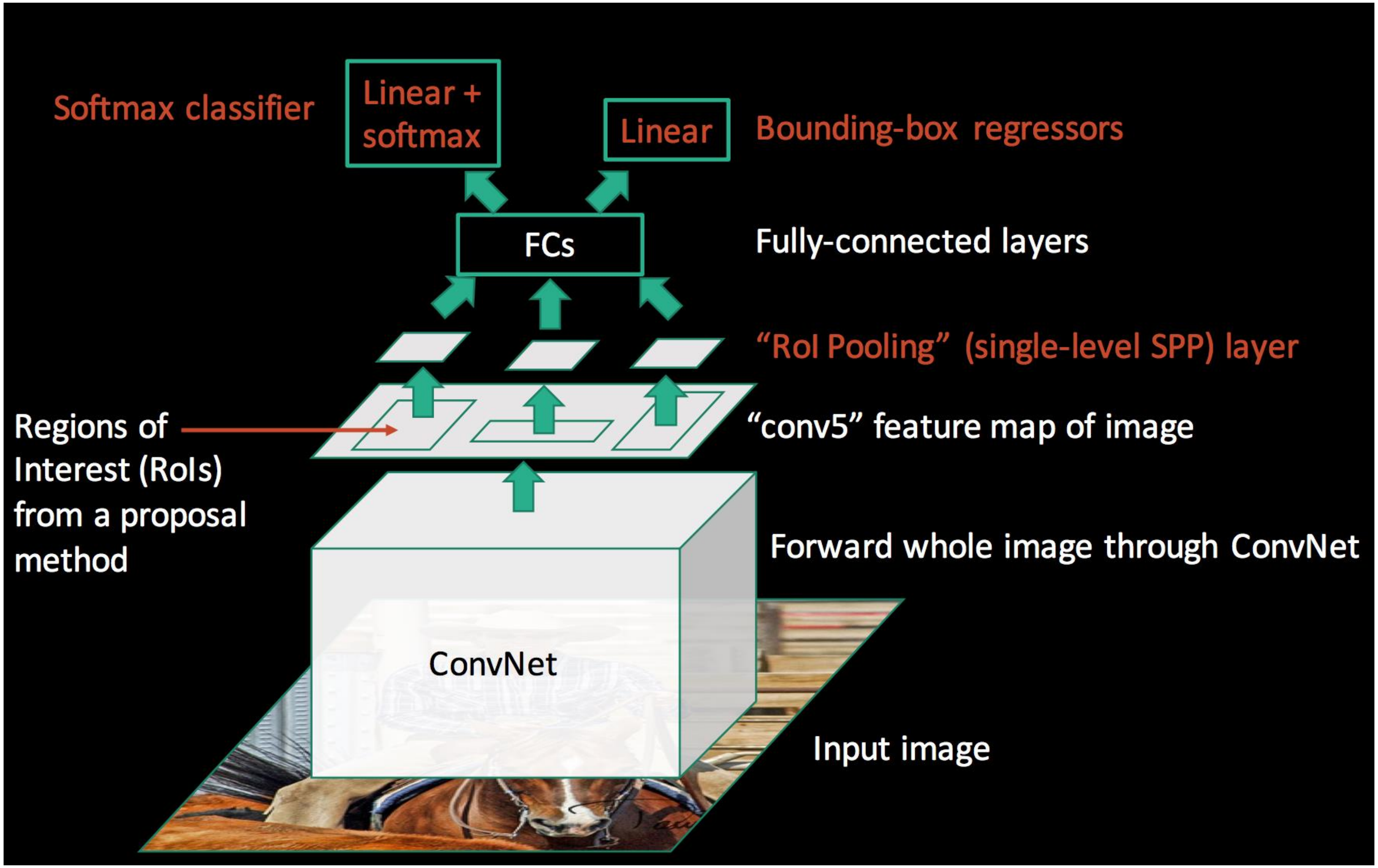
[1] Zitnick, C. Lawrence, and Piotr Dollár. "Edge boxes: Locating object proposals from edges." *European Conference on Computer Vision*. Springer International Publishing, 2014.

Region CNN



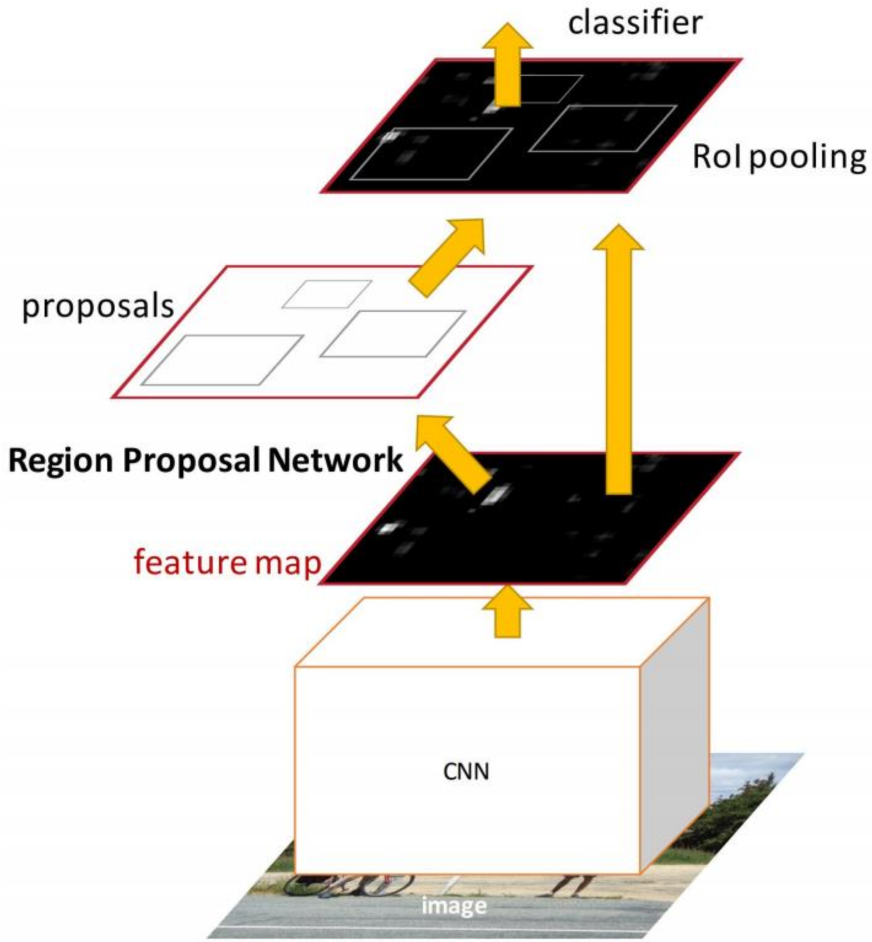
[1] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.

Fast R-CNN



[1] Girshick, Ross. "Fast r-cnn." *Proceedings of the IEEE International Conference on Computer Vision*. 2015.

Faster R-CNN



An end-to-end trainable neural network architecture using the same CNN feature map for both region proposal and proposal classification.

[1] Ren, Shaoqing, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems*. 2015.

Model Comparison

- R-CNN
  - Traditional region proposal + CNN classifier for each proposal
- Fast R-CNN
  - Traditional region proposal + CNN classifier for entire image
- Faster R-CNN
  - An unified CNN architecture for region proposal & proposal classification

	R-CNN	**Fast R-CNN**	**Faster R-CNN**
Test time	50 s	2 s	0.2 s
mAP (%)	66.0	66.9	66.9

	AlexNet	**VGG-16**	**ResNet-101**
mAP (%)	62.1	73.2	76.4

Using different CNNs in faster R-CNN

Roadmap

- Introduction
- Convolutional Neural Network
- Application
  - Image Classification
  - Object Detection
  - Object Tracking
- Our Work
- Conclusion and Discussion

Object Tracking

- ❑ Image Classification & Object Detection (discrete, static)

- ❑ Object Tracking (continuous, temporal)

Tracking by Classification

- Tracking by classification
  - Train a CNN classifier.
  - Randomly sample potential bounding boxes in new frame.
  - Classify each bounding box and pick the optimal one.
  - Update CNN weights during tracking.

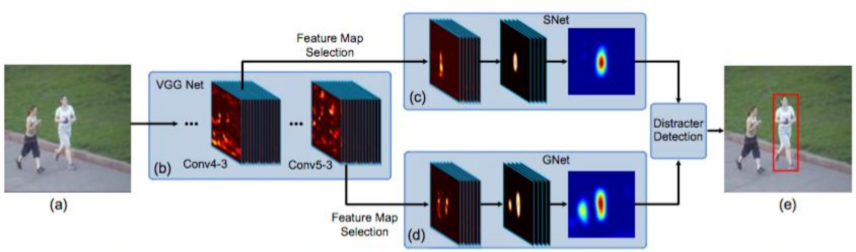
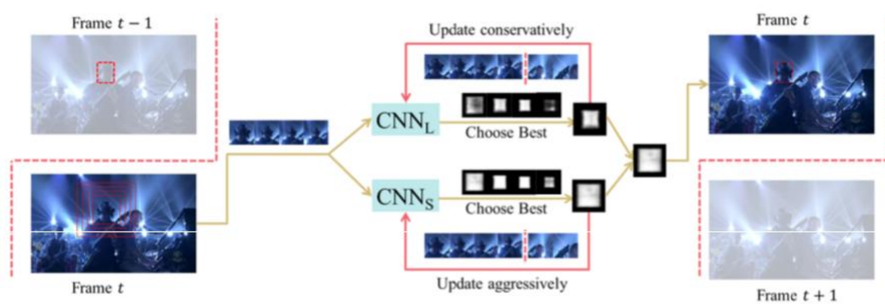


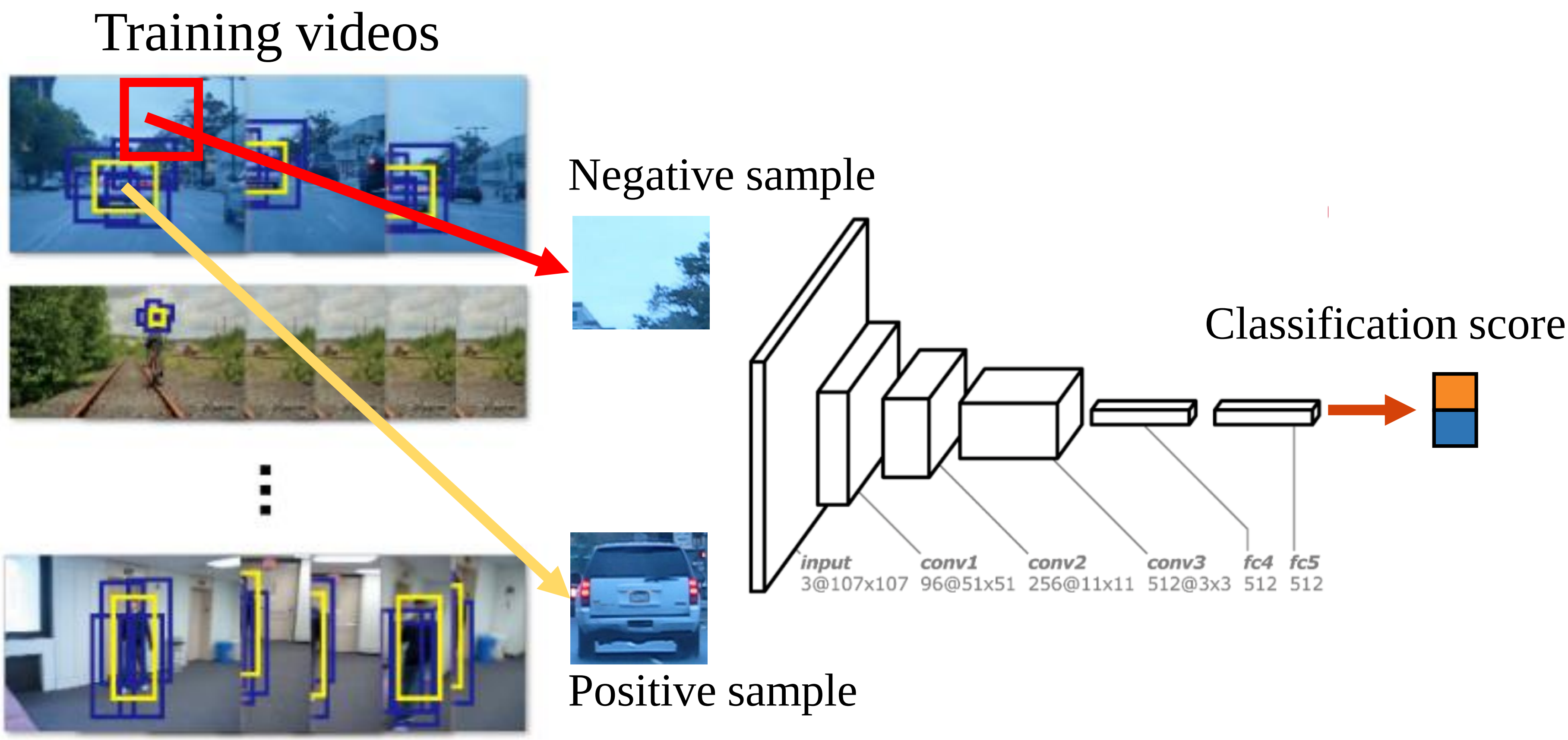
Figure 5. Pipeline of our algorithm. (a) Input ROI region. (b) VGG network. (c) SNet. (d) GNet. (e) Tracking results.



[1] Wang, Naiyan, et al. "Transferring rich feature hierarchies for robust visual tracking." arXiv preprint arXiv:1501.04587 (2015).
[2] Wang, Lijun, et al. "Visual tracking with fully convolutional networks."Proceedings of the IEEE International Conference on Computer Vision. 2015.
[3] Nam, Hyeonseob, and Bohyung Han. "Learning multi-domain convolutional neural networks for visual tracking." arXiv preprint arXiv:1510.07945 (2015).

MDNet

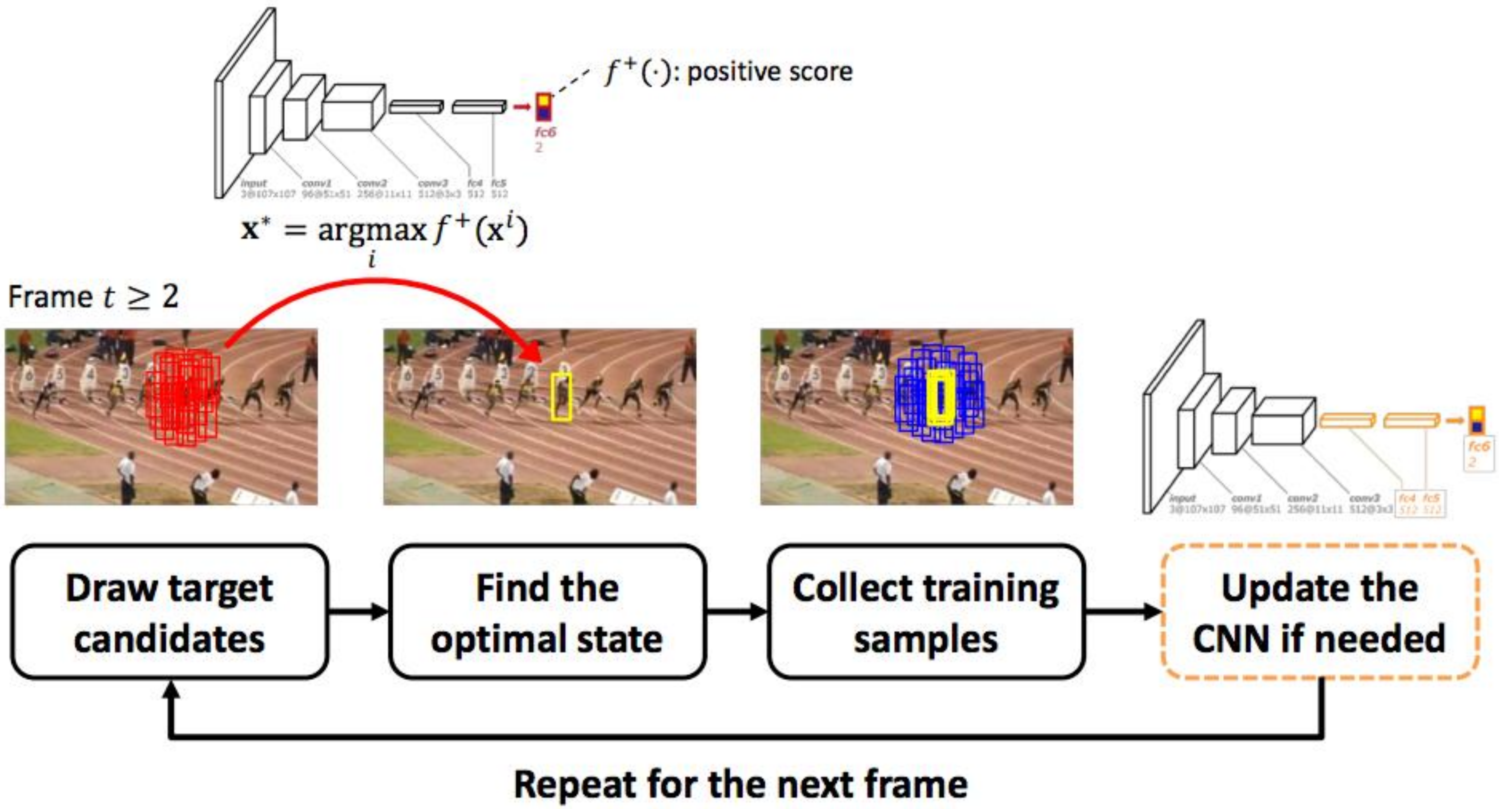
❑ Train a CNN classifier offline



[1] Nam, Hyeonseob, and Bohyung Han. "Learning multi-domain convolutional neural networks for visual tracking." arXiv preprint arXiv:1510.07945 (2015).

MDNet

- Apply the classifier for each consecutive potential regions



[1] Nam, Hyeonseob, and Bohyung Han. "Learning multi-domain convolutional neural networks for visual tracking." arXiv preprint arXiv:1510.07945 (2015).

MDNet

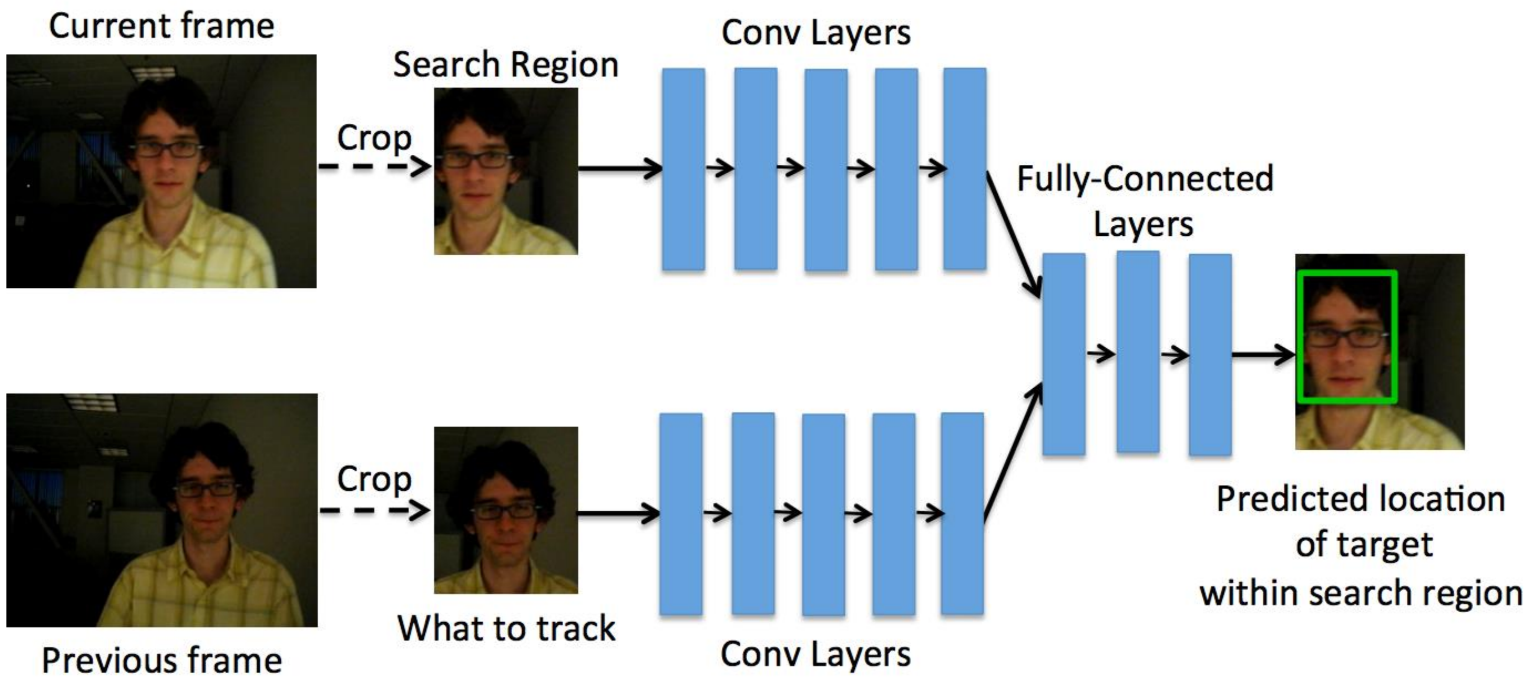
- Fine-tune the classifier online during tracking

- **Long-Term Updates**
  - performed at regular intervals
  - using long-term training samples
  - **For Robustness**
- **Short-Term Updates**
  - performed at abrupt appearance changes ($f^+(\mathbf{x}^*) < 0.5$)
  - using short-term training samples
  - **For Adaptiveness**

[1] Nam, Hyeonseob, and Bohyung Han. "Learning multi-domain convolutional neural networks for visual tracking." arXiv preprint arXiv:1510.07945 (2015).

Tracking by Regression

- ❑ Two Convolutional neural networks
  - ❑ A search region from the current frame.
  - ❑ A target from the previous frame.
  - ❑ Compare crops and find target object.



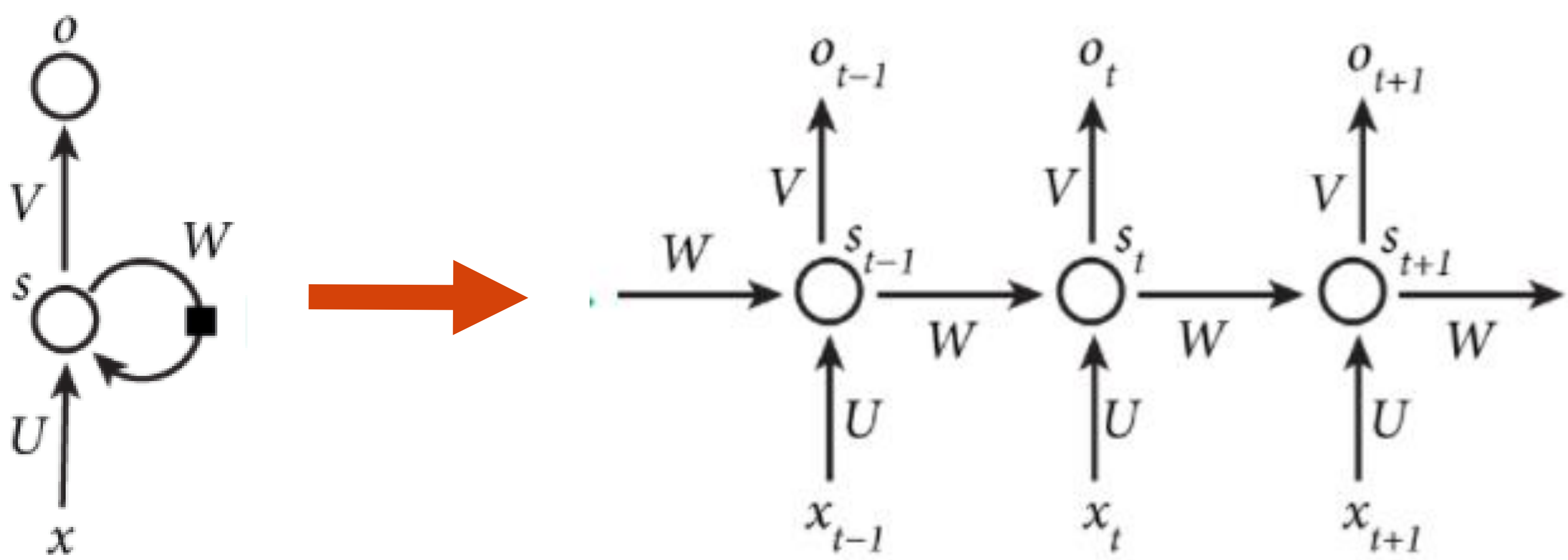
[1] Held, David, Sebastian Thrun, and Silvio Savarese. "Learning to Track at 100 FPS with Deep Regression Networks." *arXiv preprint arXiv:1604.01802*(2016).

Roadmap

- ❑ Introduction
- ❑ Convolutional Neural Network
- ❑ Application
  - ❑ Image Classification
  - ❑ Object Detection
  - ❑ Object Tracking
- ❑ Our Work
- ❑ Conclusion and Discussion

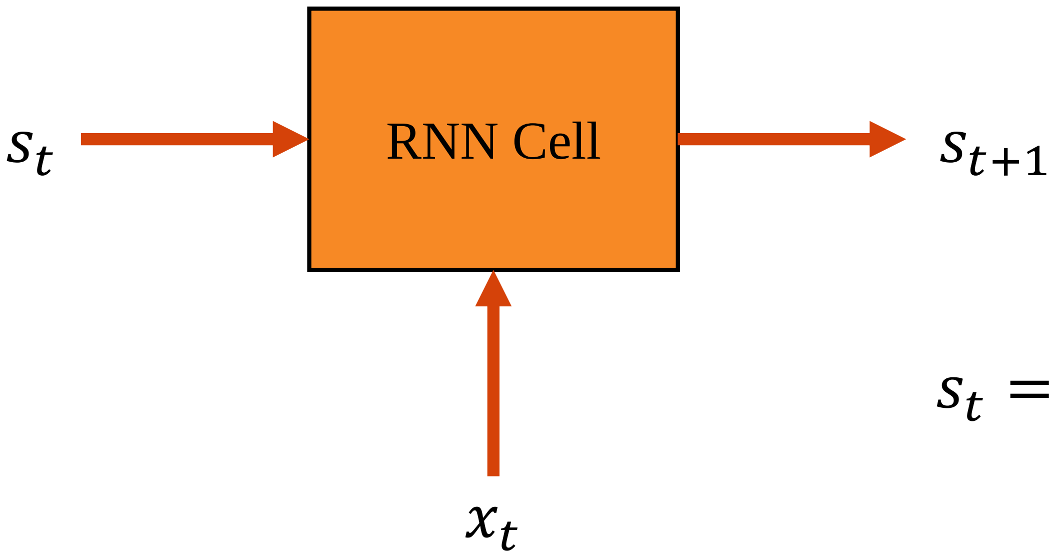
Recurrent Neural Network

- Recurrent Neural Network
  - Better for processing sequential information.
  - Creates and updates a hidden state.
  - The hidden state captures a history of inputs.



$$s_t = f(Ux_t + Ws_{t-1})$$

Different RNN Cells

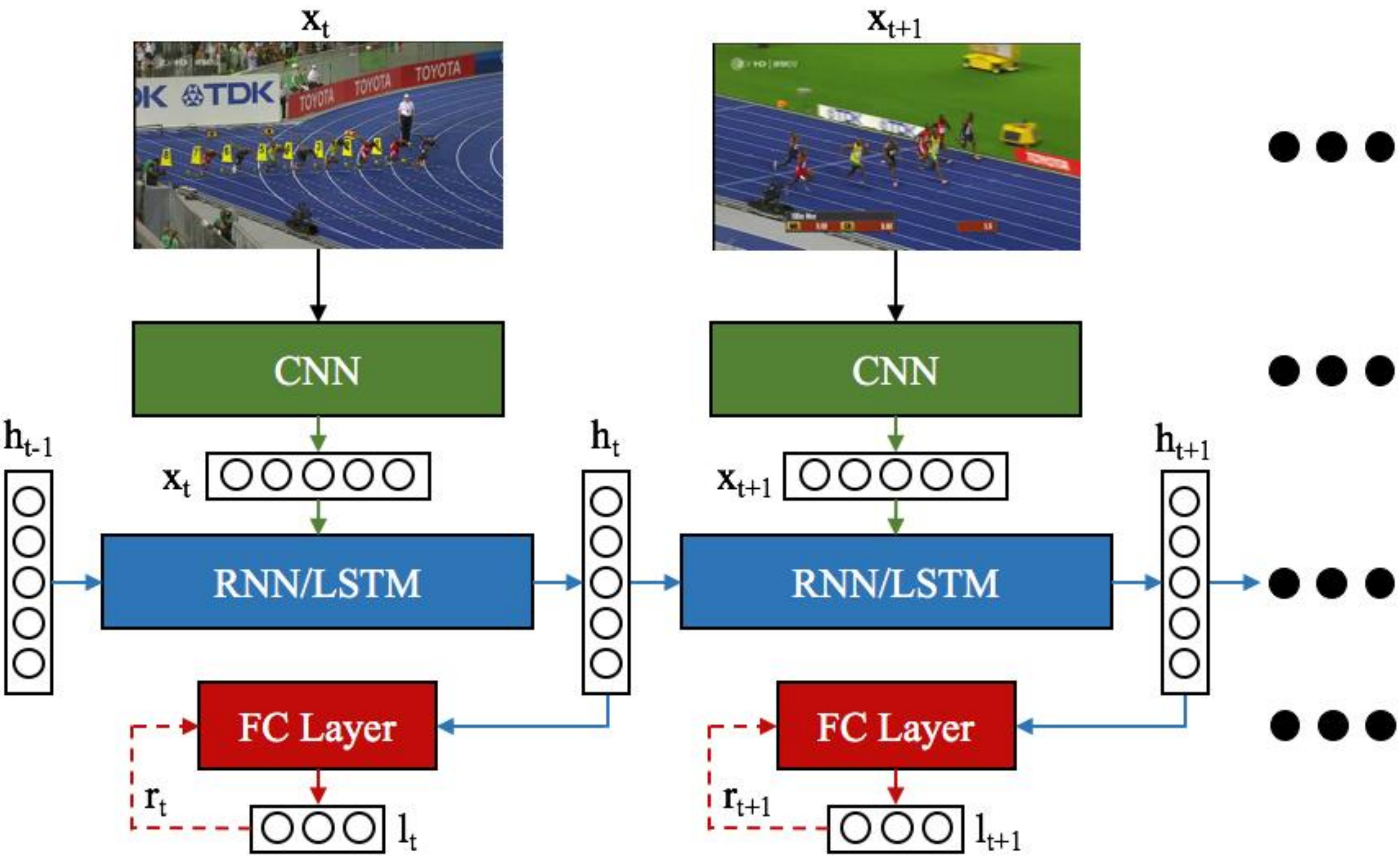


$$s_t = f_W(x_t, s_{t-1})$$

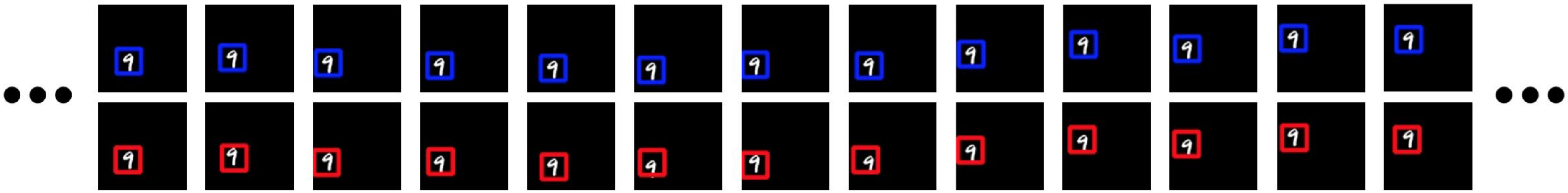
- RNN Cells

  Basic RNN Cell, LSTM Cell, GRU Cell and etc.

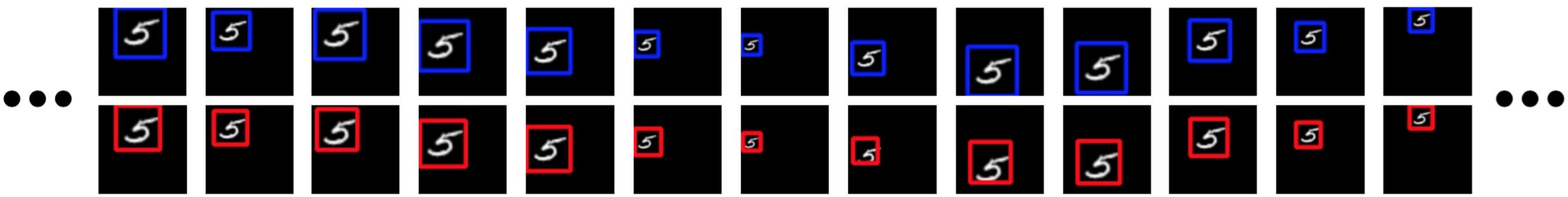
Proposed Framework



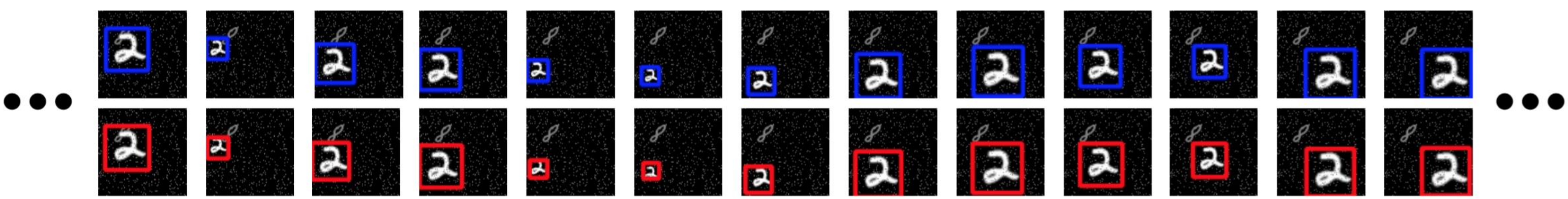
Preliminary Results



(a) Fixed-size MNIST moving against black background



(b) Scalable MNIST moving against black background



(c) Scalable MNIST moving against noisy background

Discussion

Computer Visual Recognition

Discrete, static images

Image Classification:

- image => class scores
- Convolution Neural Network
- AlexNet, VGGNet, ResNet

Object Detection:

- image => location+scores
- Region proposal+CNN
- End-to-end CNN design

Continuous, temporal videos

Object Tracking:

- video => location sequence
- Tracking by classification
- Tracking by regression

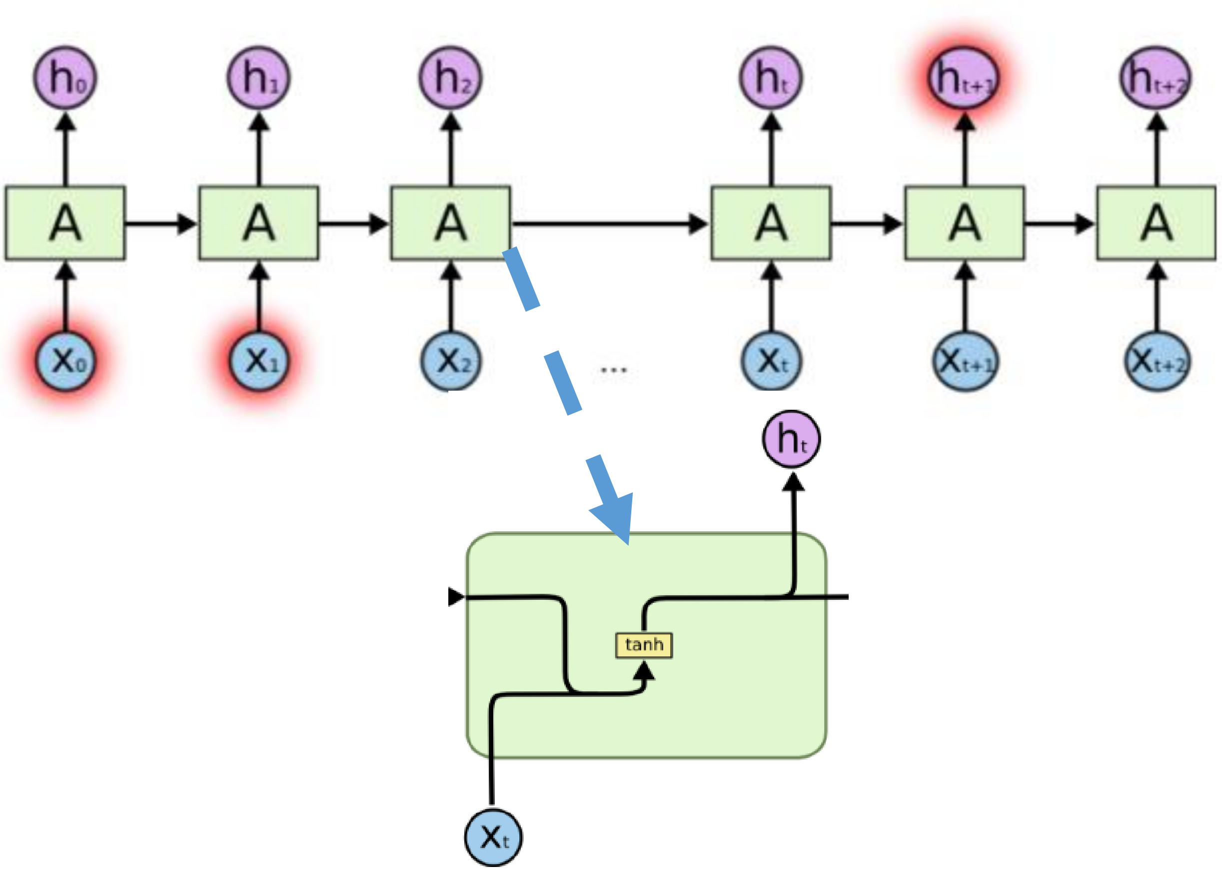
Our work:

- Recurrent Neural Network
- Reinforcement Learning
- End-to-end tracker design

[1] https://docs.google.com/spreadsheets/d/1Q-61cQxzt3ehiSUMSNZeQMmMVIOrftijTJJoeWrh3-0/edit#gid=0

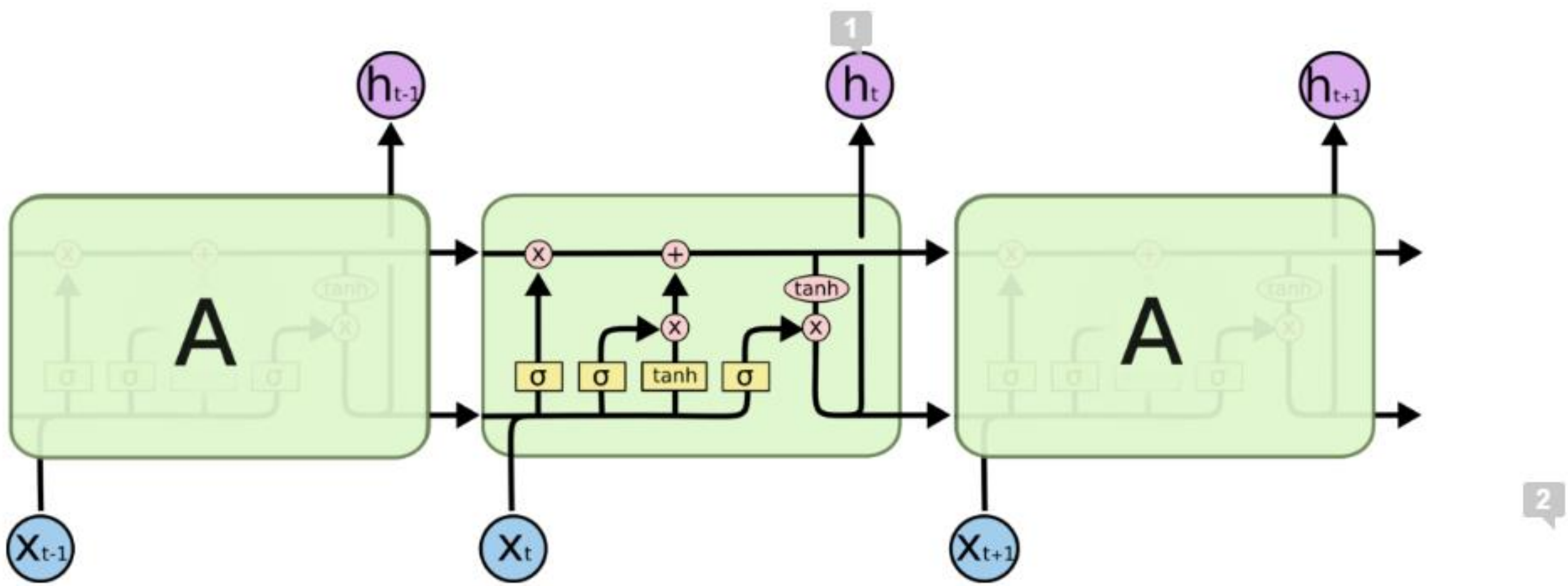
Others: LSTM

- ❑Traditional RNN have very simple internal structures
- ❑Long-term propagation can be tricky



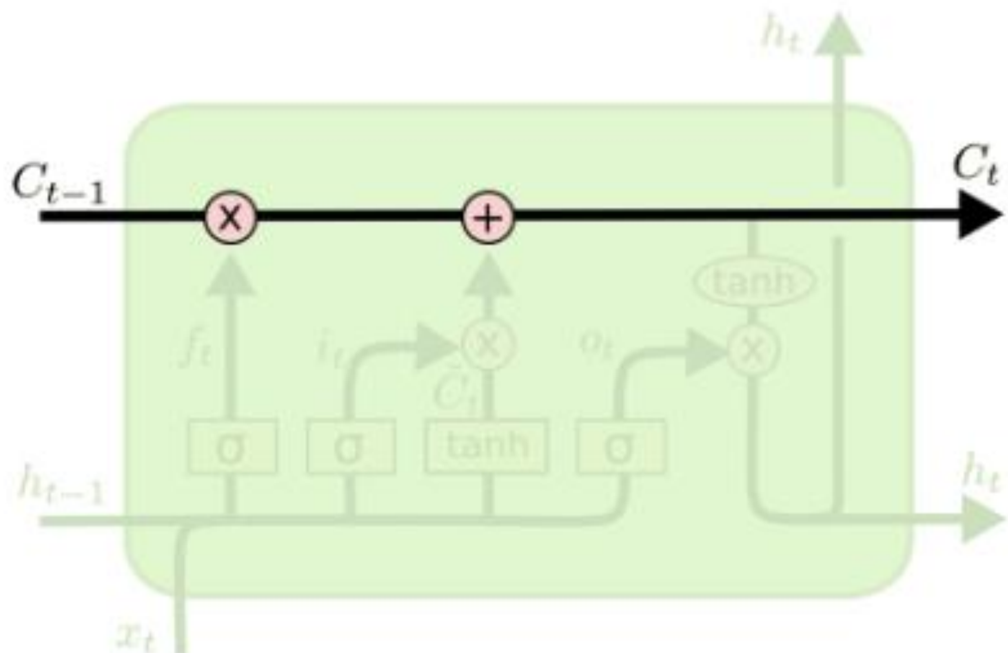
Others: LSTM

- ❑LSTM has more elaborate internal structure
- ❑Think about fancy Kalman Filter or HMM



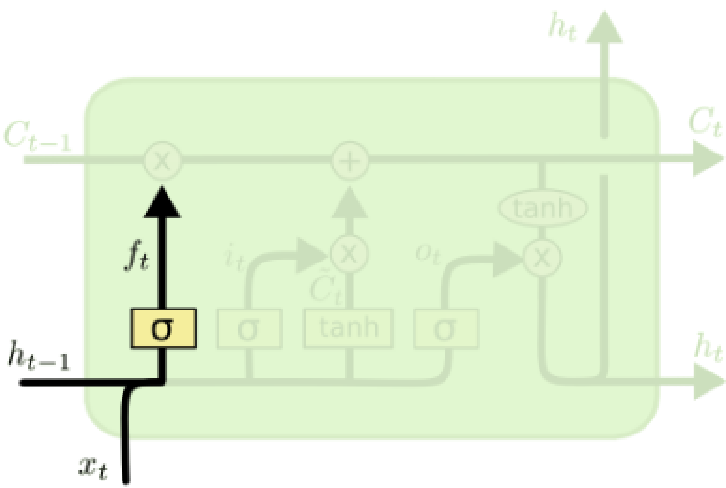
Others: LSTM

- It maintains an internal state (C_t a bit string)



Others: LSTM

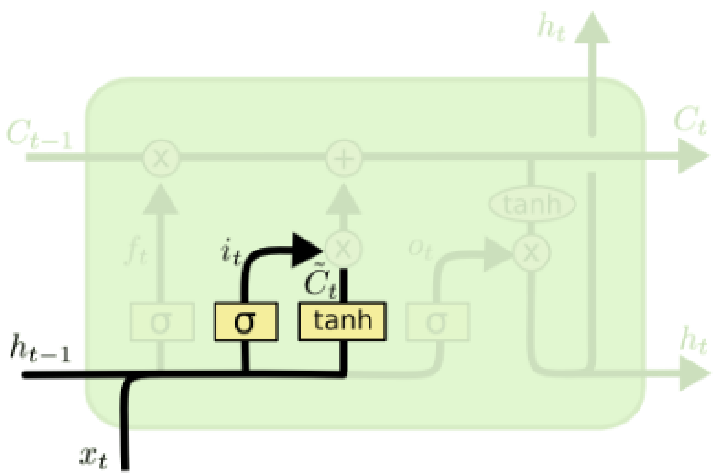
- State vector evolves with time
  - Influenced by inputs (x_t), prior state (C_{t-1}) and prior output (h_{t-1})
  - Can be forgotten (f_t: remembering, or forgotten, factor)



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

Others: LSTM

- State vector evolves with time
  - Influenced by inputs (x_t), prior state (C_{t-1}) and prior output (h_{t-1})
  - Can be augmented and changed by current input
    - i_t: changed (new) factor
    - $\widehat{C}_t$: changed (new)state

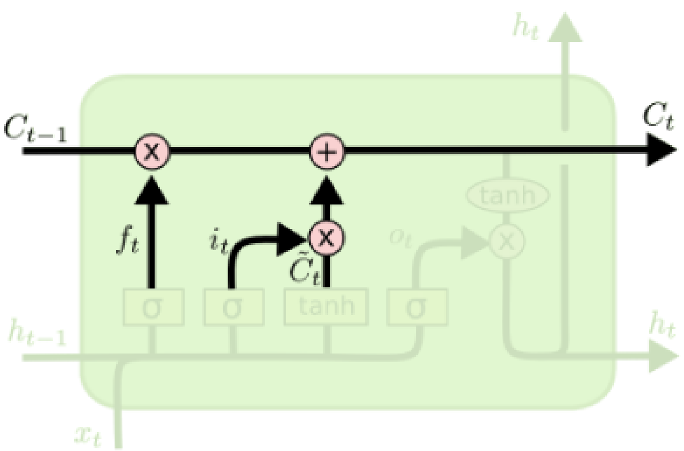


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

Others: LSTM

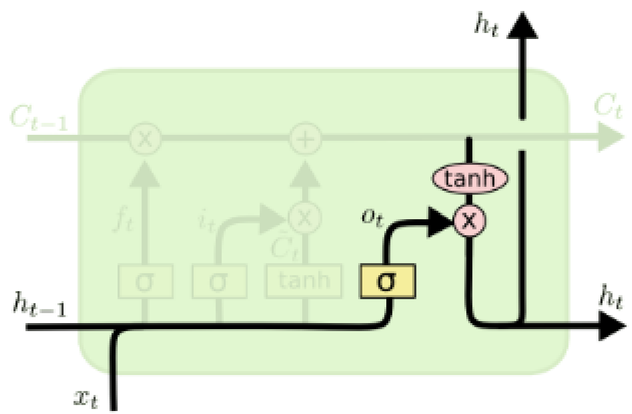
- State vector evolves with time
  - Influenced by inputs (x_t), prior state (x_{t-1}) and prior output (h_{t-1})
  - These factors are then combined to generate next state



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Others: LSTM

- Output Vectors
  - Determined by state and input

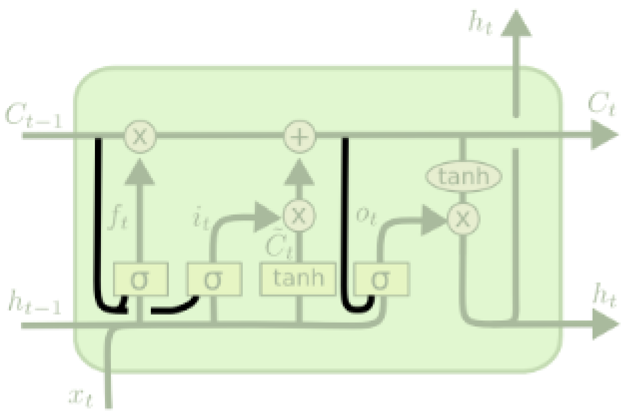


$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

Others: LSTM

- Previously, state (C) is assumed to be hidden (not directly observable), similar to
  - Kalman Filter
  - HMM

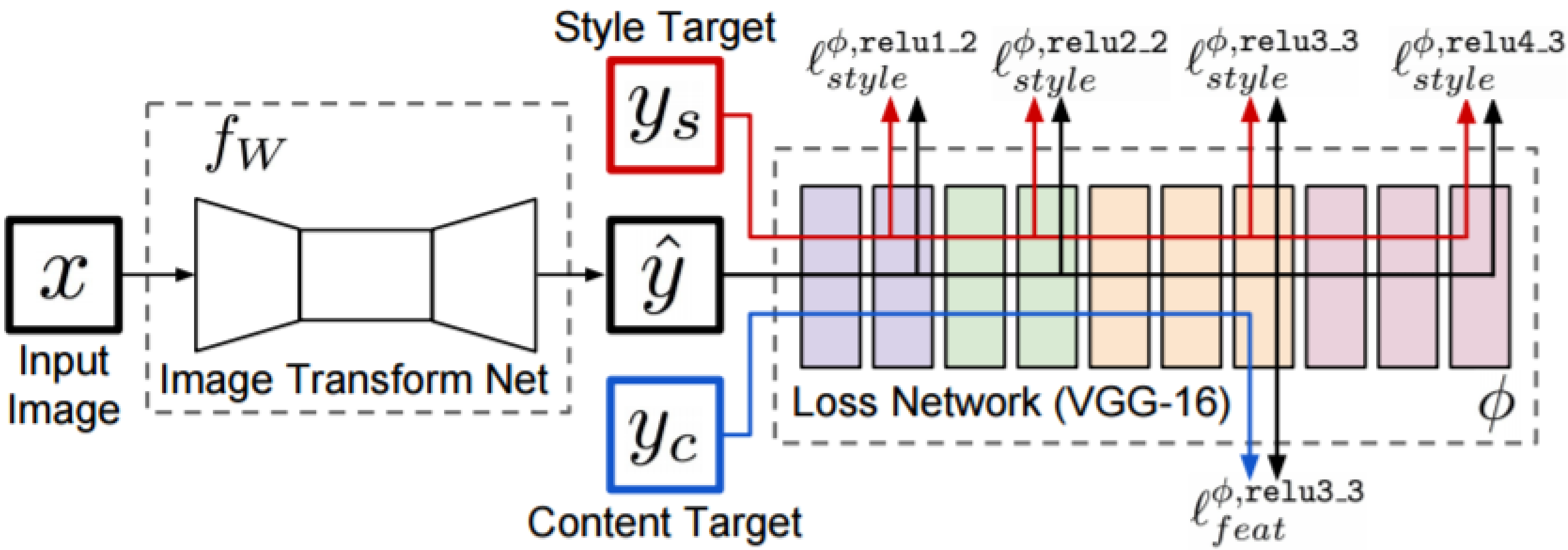


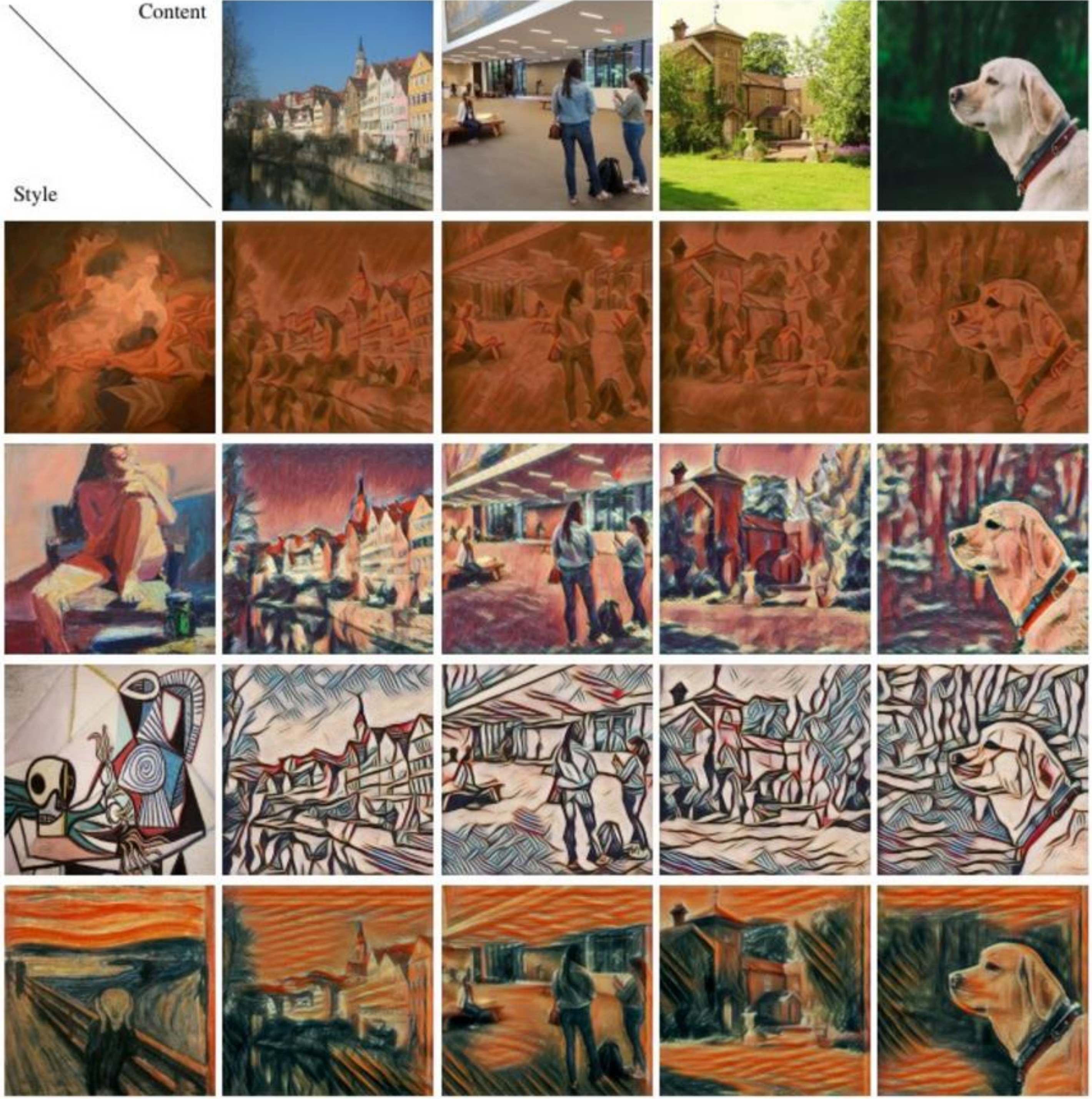
$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \;+\; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \;+\; b_o\right)$$

Others: Style Transfer Network

❑Networks with two kinds of losses

- Content losses
- Style losses




Content
Style

Others: GAN

- Dual competing networks: Generative + discriminative
- Generator
  - Seeded with randomized inputs with predefined distribution
  - Back propagate with negation of classification error of the discriminator
- Discriminator
  - CNN: trained with certain data sets
- Goal: Generator to learn the Discriminator distribution as to generate results that are indistinguishable from training data to the Discriminator

Conclusion and Discussion

- Advances make CNN very efficient today
  - Big data and great computational power.
  - Research advancements: ReLU, dropout and etc.

- CNN is the basic building block for computer vision system
  - Hierarchical and general features.
  - End-to-end training.

- Still lots of unsolved problems in visual recognition
  - Special image analysis (satellite and etc.)
  - Video analysis
  - Multi-view analysis (activity recognition and etc.)
  - New architecture + new application
  - …