Mid-semester questionnaire

Written exam detailed info in case of online exam:

- Installation guideline and exam protocol (read carefully!)
- Link to exam.net (log in and enter the exam key): https://exam.net
- *The current exam Key (activated at the exam start): <tba>*
- Zoom link for the exam (open in your smart phone): <tba>
- Crucial: (1) *do not log out once entering the key in exam.net* ; (2) *Write down the exam key on a sheet of paper -- once the SEB starts, it will lock down your comp.*

Online exam protocol and setup instructions

Announcements & Discussion

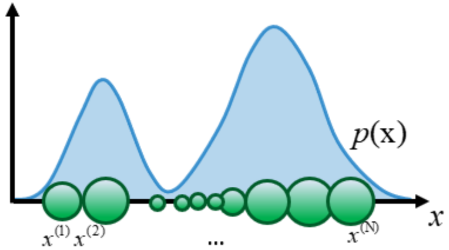
Questionaire about the course Advanced topics in computer vision (2022/23)

- Open until this Thursday (20.4.)
- Please give feedback on lectures/assignments
- Help us improve the course

Previously at ACVM…

- Posterior is non-Gaussian, solve by MC → Particle filter (PF)

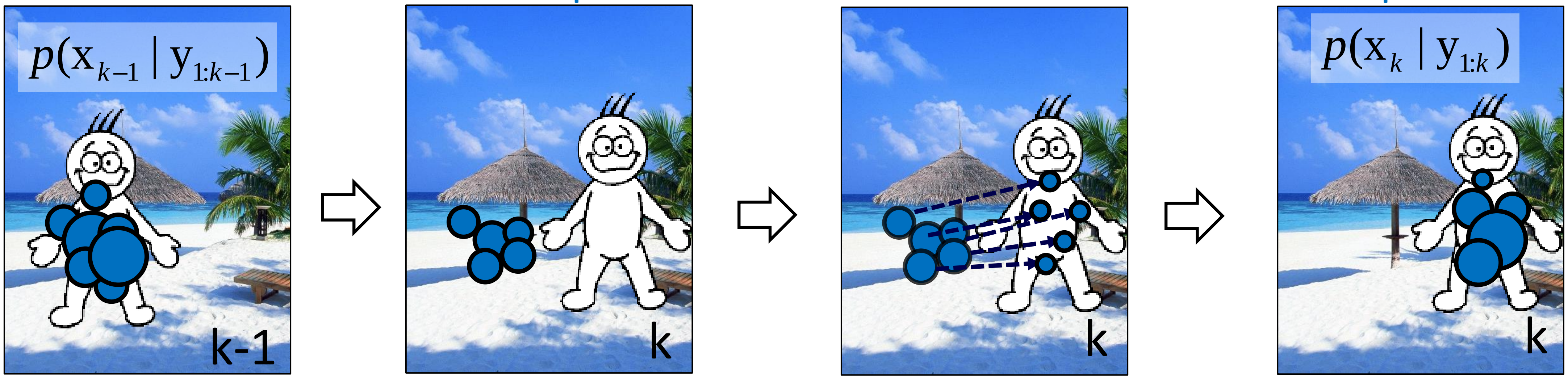
$$\underbrace{p(\mathbf{x}_k|\mathbf{y}_{1:k})}_{\text{posterior estimate}} \propto \underbrace{p(\mathbf{y}_k|\mathbf{x}_k)}_{\text{Observation model}} \int \underbrace{p(\mathbf{x}_k|\mathbf{x}_{k-1})}_{\text{motion model}} \underbrace{p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})}_{\text{posterior at k-1}} d\mathbf{x}_{k-1}$$



$$p(\mathrm{x}) \approx \sum_{i=1}^{N} \mathrm{w}^{(i)} \delta_{\mathrm{x}^{(i)}}(\mathrm{x})$$

- Recursion replaced by (re)sampling and re-weighting: Bootstrap PF

Resample | Predict | Update



Univerza v Ljubljani

Advanced CV methods
Fully-trainable trackers – deep learning for tracking

Matej Kristan

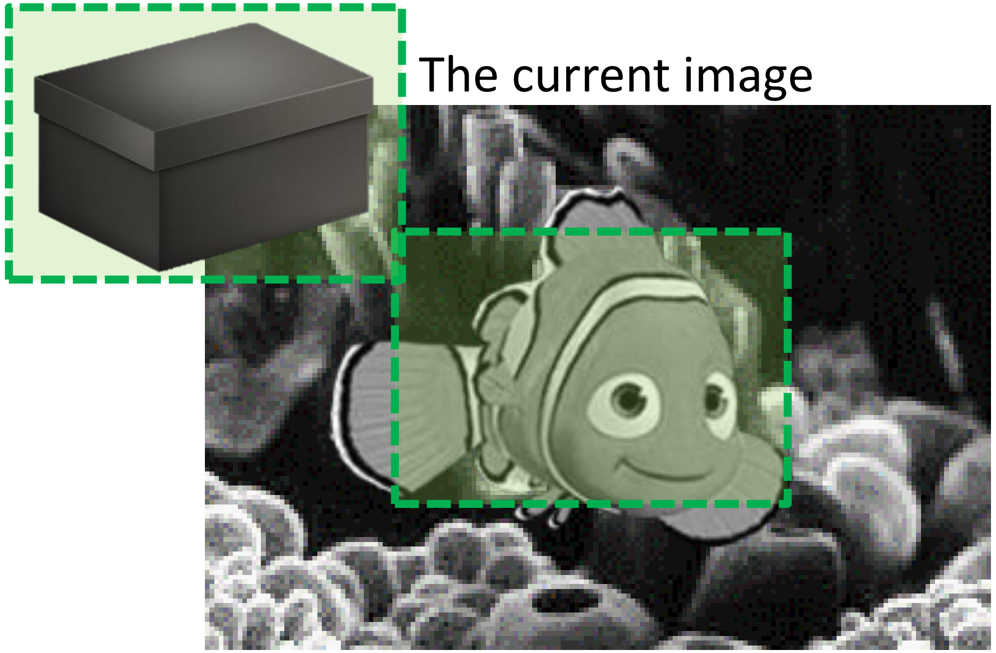
Laboratorij za Umetne Vizualne Spoznavne Sisteme,
Fakulteta za računalništvo in informatiko,
Univerza v Ljubljani

Recall tracking by online classifiers

Postive training examples

Negative training examples

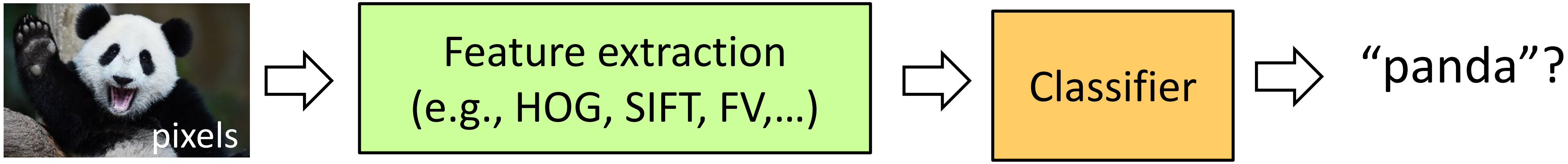
Feature extraction + classification



Traditional vs. modern approach

Traditional:

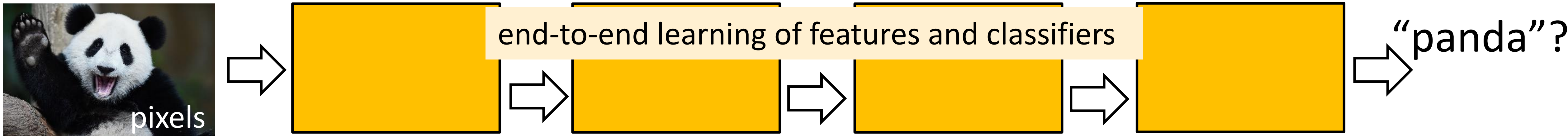
- (i) extract hand-crafted features, (ii) train a classifier



Modern ("started" in late 90s, entered mainstream in 2012):

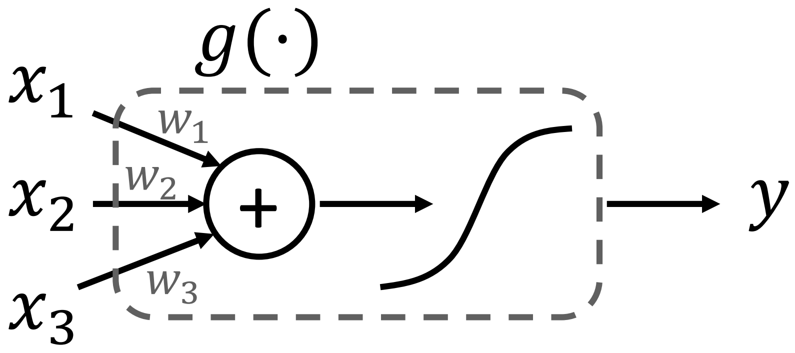
- Jointly learn features AND the classifier

  (without specifying where feature extraction ends and classifier begins)



Recall a simple neural network

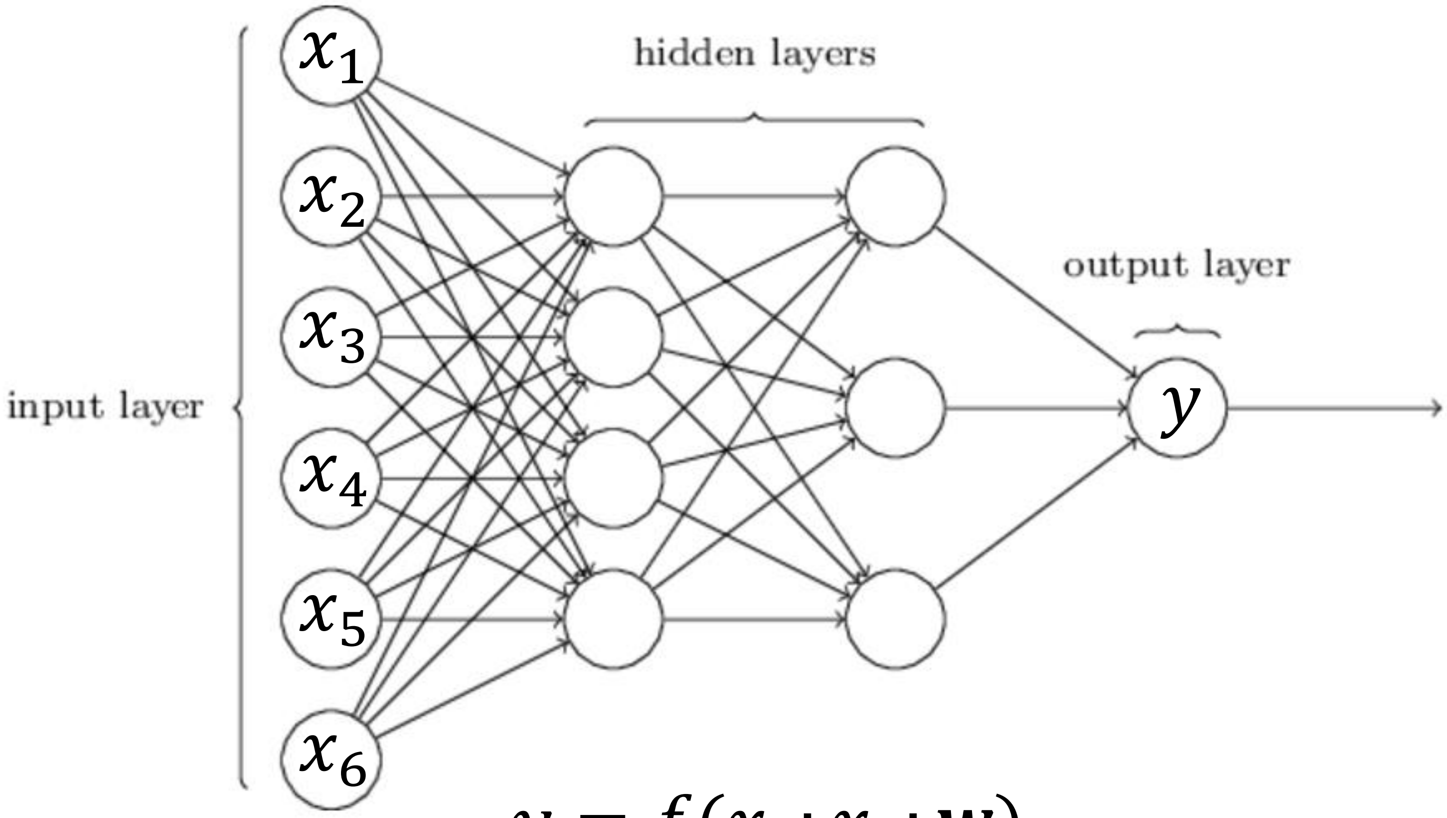
A sigmoid neuron:



A weighted sum of values x_i, transformed by a nonlinear function $g(\cdot)$:

$$y = g(\sum_i w_i x_i)$$

A network of neurons:

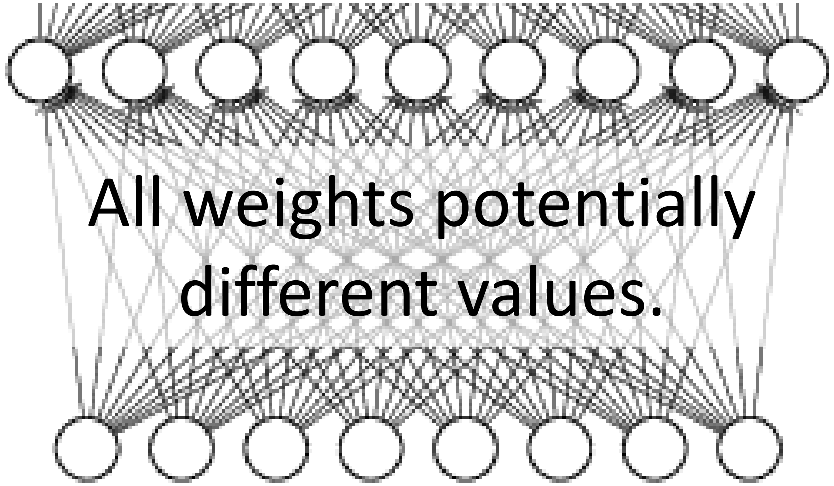


$$y = f(x_1 \colon x_6; \boldsymbol{w})$$

A traditional vs. convolutional neural network (CNN)

A fully connected neural network (fcn)

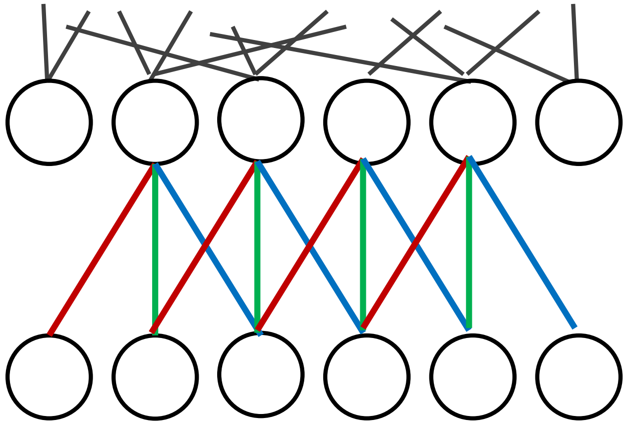
output



input

A convolutional neural network (CNN)

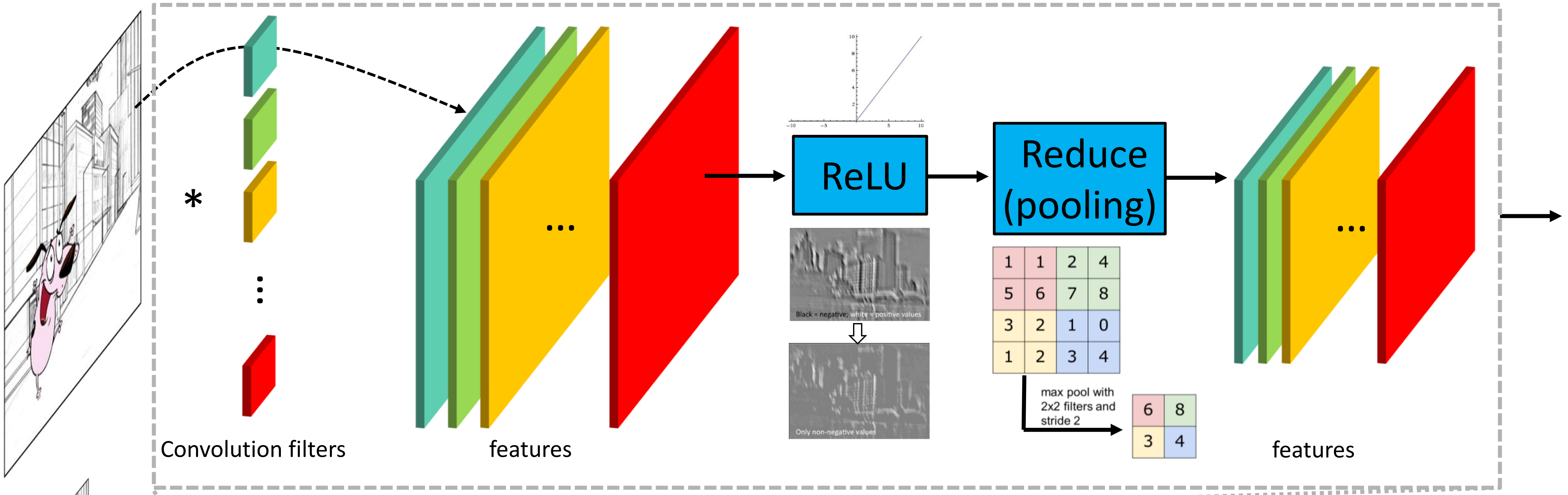
output

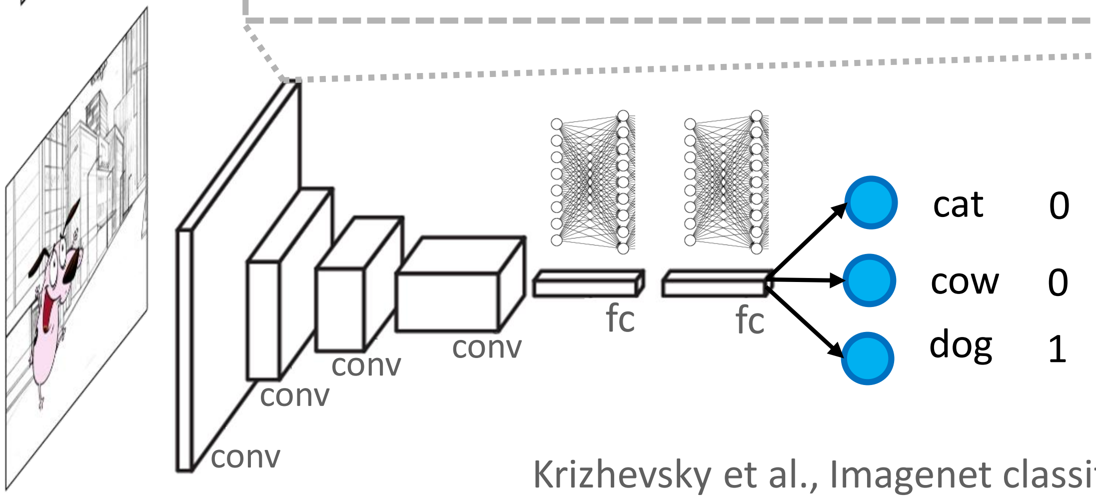


input

- See, e.g. http://cs231n.stanford.edu/, for a good intro to CNNs

The basic CNN architecture



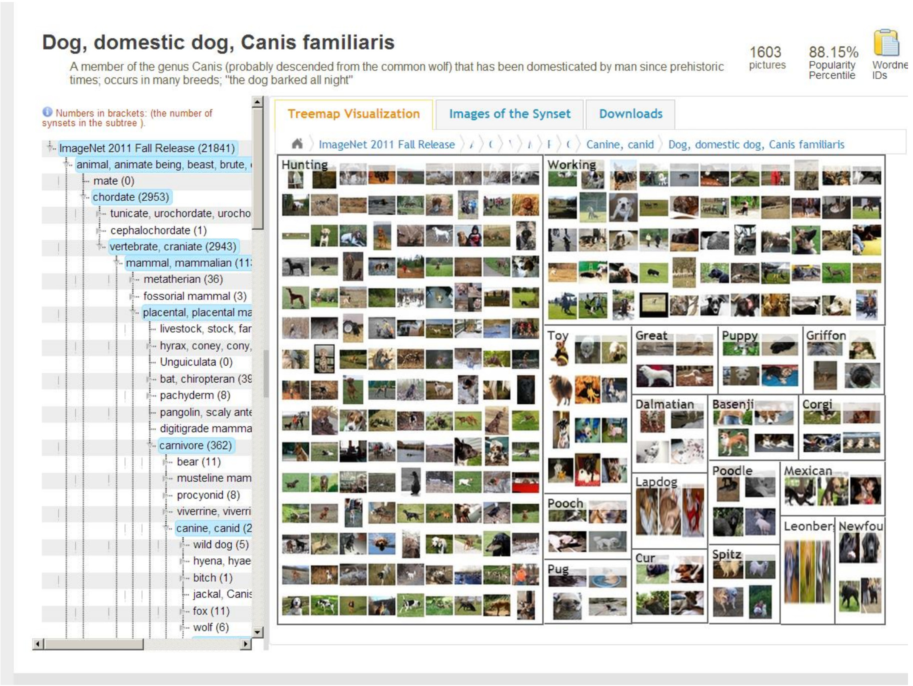


- The filters in all layers (and other free parameters) are trained to maximize the network accuracy.

Krizhevsky et al., Imagenet classification with deep convolutional neural networks, NIPS2012 (>100k citations!)

CNNs for object recognition (and more…)

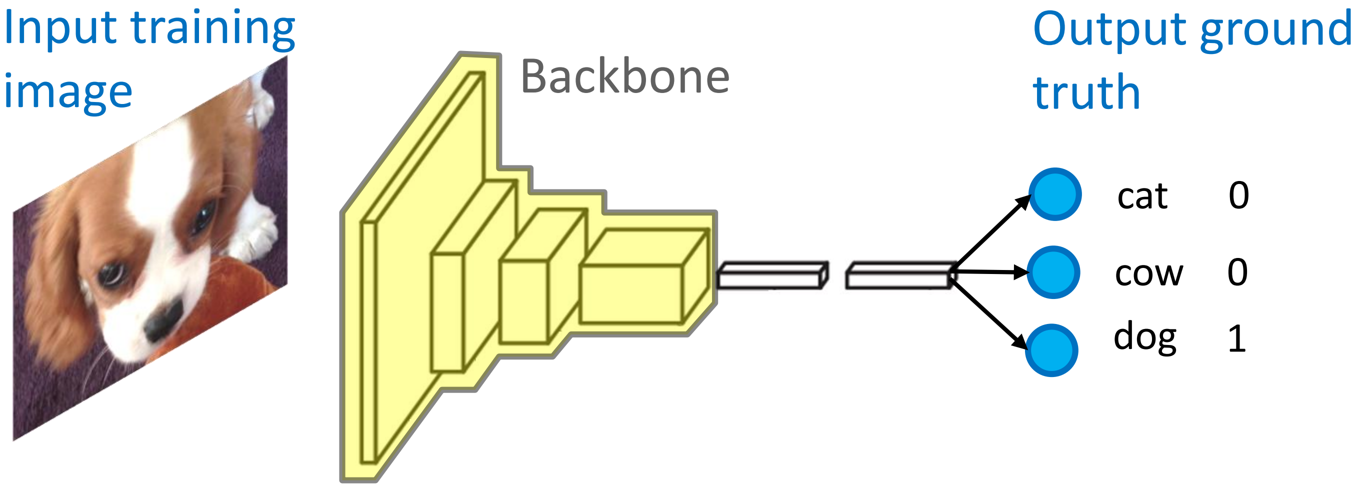
- Task to answer: *"Which object category is in the image?"*



(Introduced in 2006 by Fei Fei Li et al.)

14 million labeled images,
20K categories

http://www.image-net.org/



Feature generalization:
Early layers of the pre-trained network (backbone) can be repurposed for other tasks, e.g., for segmentation…

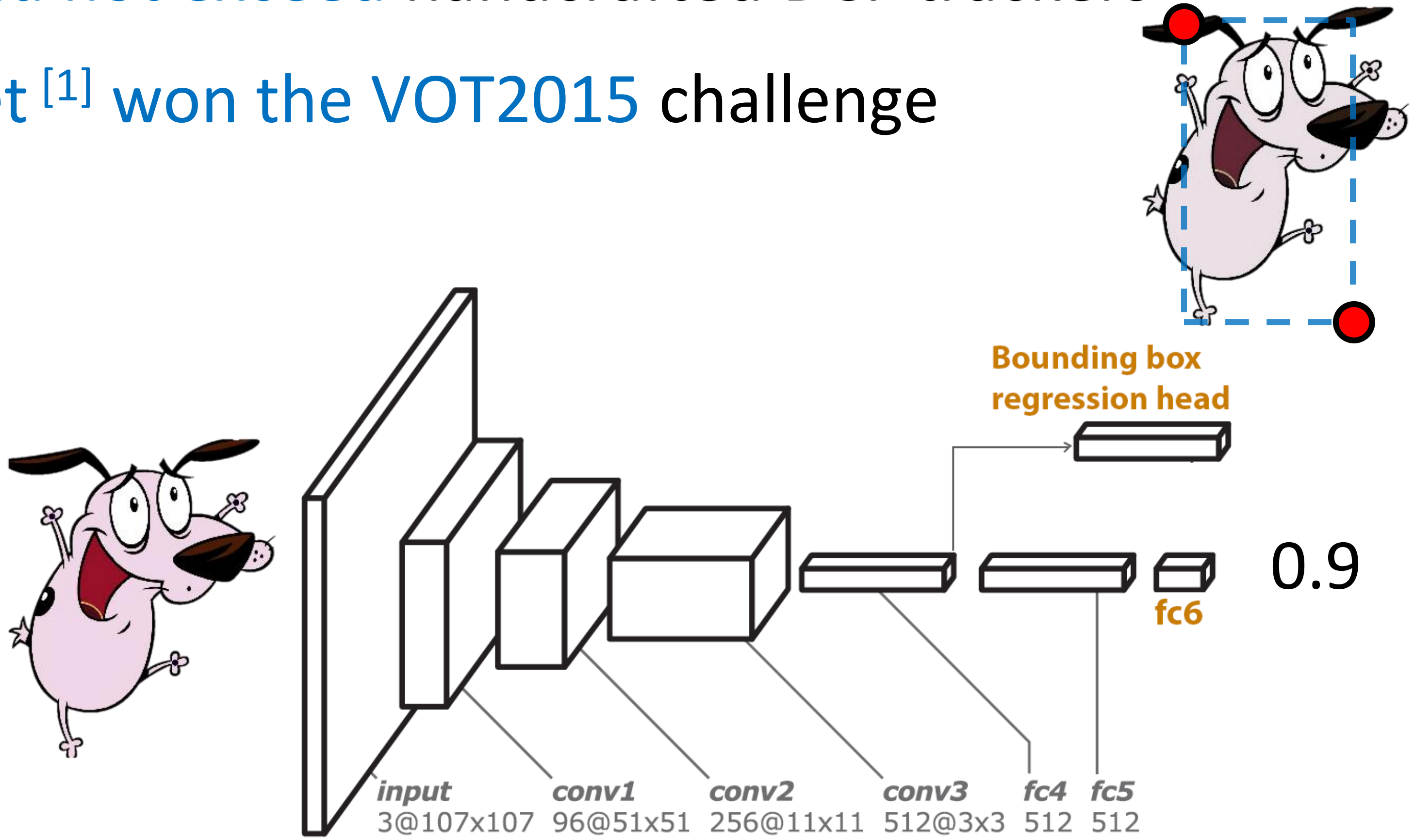
Krizhevsky et al., Imagenet classification with deep convolutional neural networks, NIPS2012 (>100k citations!)

CNN architectures for tracking (2015 onward)

- CNNs were first successfully applied to recognition, detection, semantic segmentation, optical flow, …
- But it took a while to come up with architectures and learning strategies appropriate for online tracking
- Overall: tracking has drawn significantly on object detection research
- In the following we will overview what I consider milestones in CNN trackers that made significant leaps in performance
(this is by no means an exhaustive overview)

MDNet: Multi-Domain Convolutional Neural Network Tracker

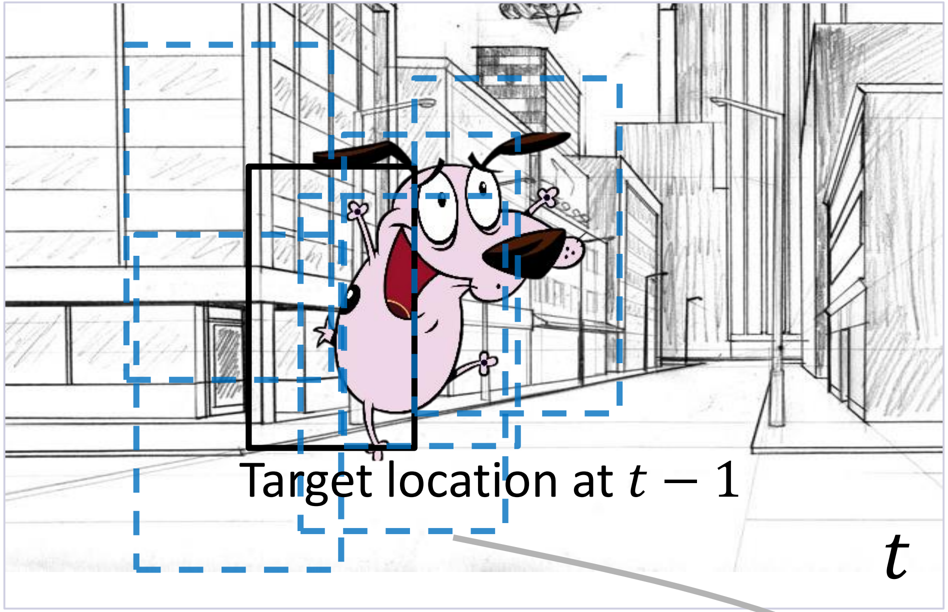
- Several attempts made to harvest the CNN potential in tracking
- Until 2015 the CNN trackers did not exceed handcrafted DCF trackers
- In 2015 a tracker called MDNet [1] won the VOT2015 challenge
- Core ideas:
  - Draw on recent developments in object detection/recognition
  - Light-weight backbone
  - Efficient backbone pre-training
  - Efficient online training

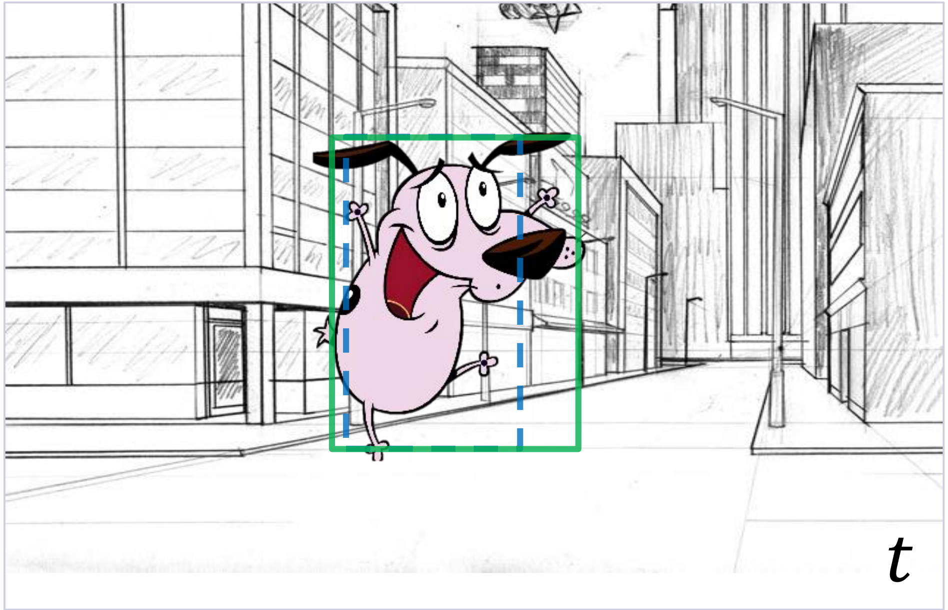


https://github.com/hyeonseobnam/MDNet

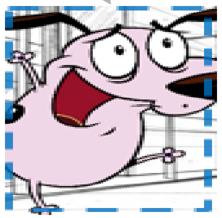
[1] Nam and Han, Learning Multi-Domain Convolutional Neural Networks for Visual Tracking, CVPR2016

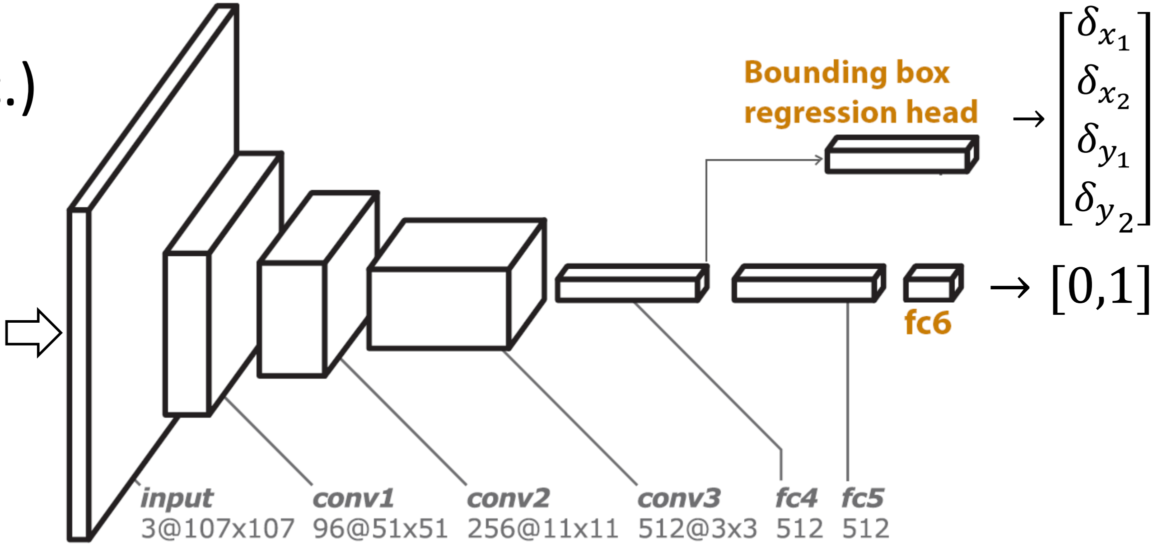
MDNet: Target localization principle





- Sample bounding boxes (around t-1 loc.)
- Compute classification score
- Take the BB with max. score
- Regress the BB parameters

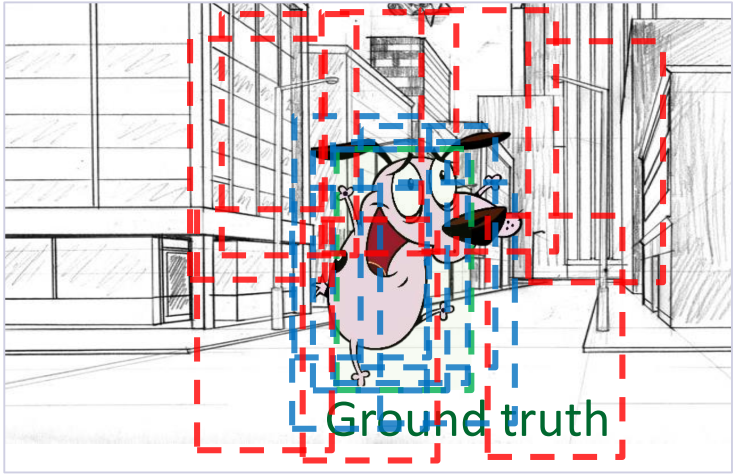


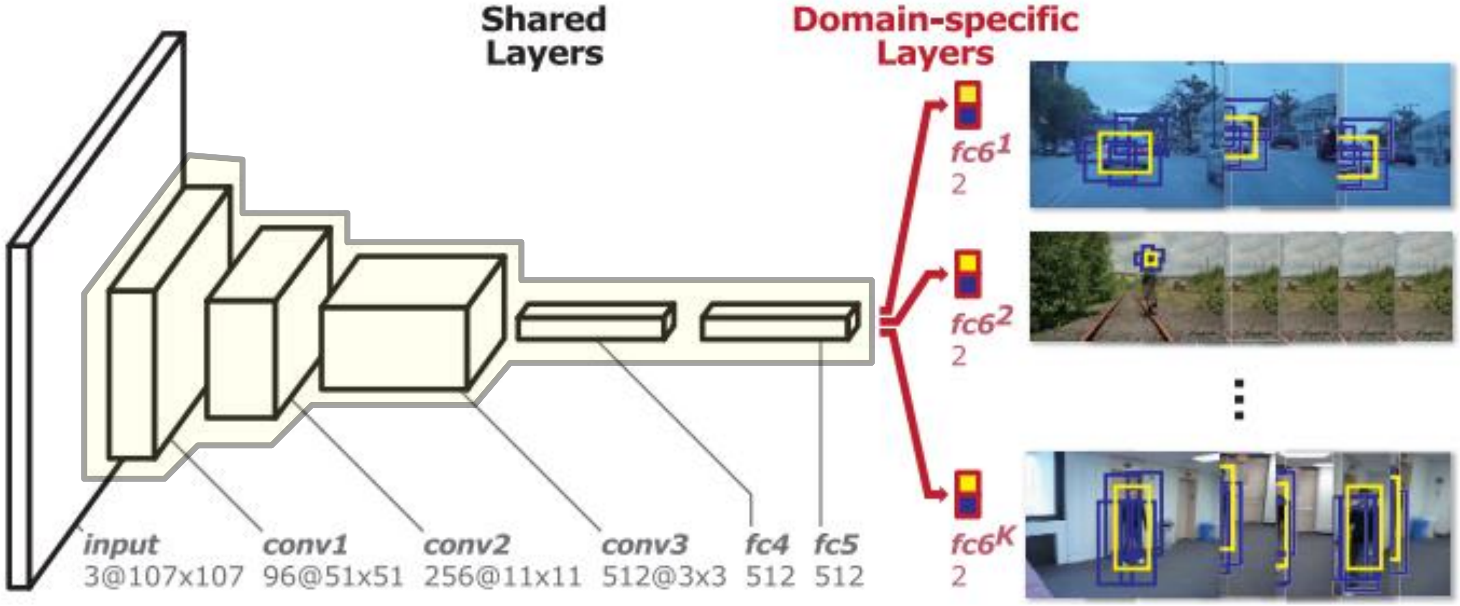


MDNet: backbone pre-training

- Pre-trained on sequences, with each sequence having its own fc6
- Assumption:
  - Each sequence is its own tracking domain and requires a specialized fc6
  - But the backbone should be shared among all “domains” (sequences)

In each selected frame, sample
50 pos & 200 neg samples

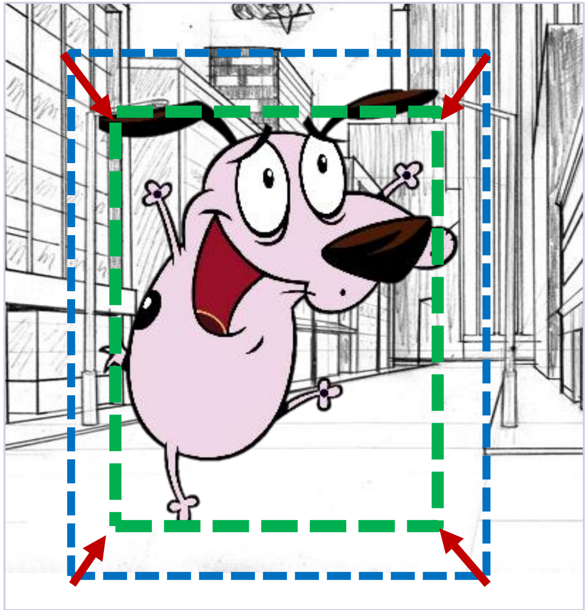




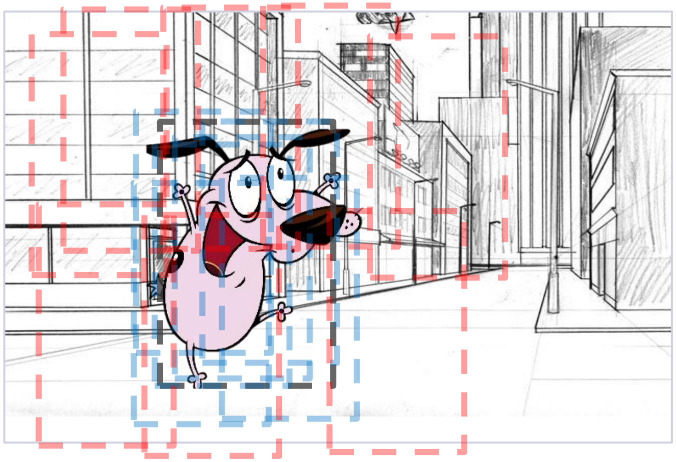
MDNet: Initialization on a new sequence

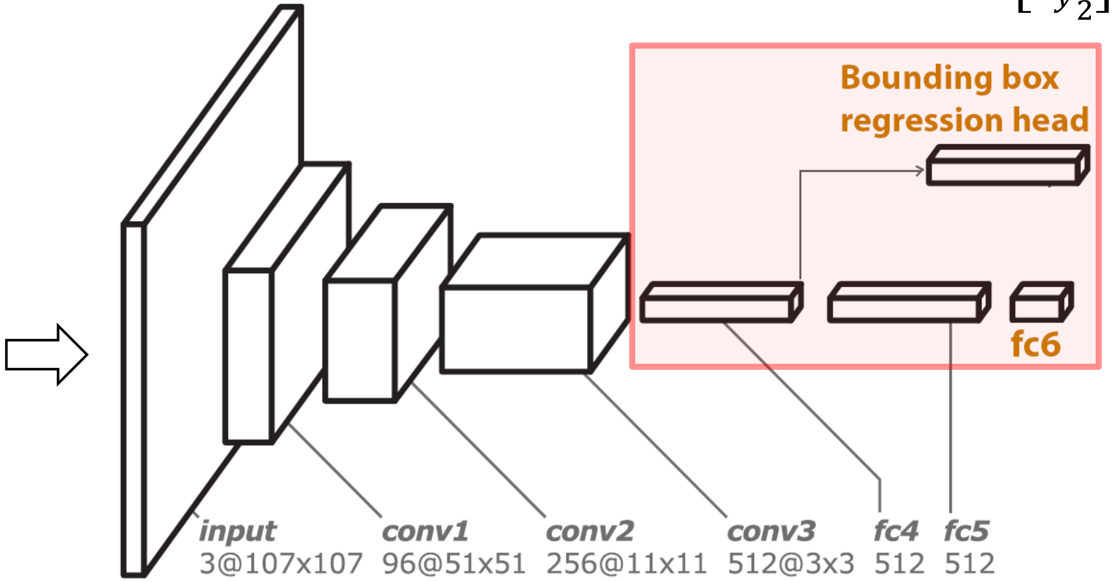
- After pre-training, the fc6 layers are removed and a new fc6 is created
- Initialization frame:
  - Fine-tune fc4/5 layers train fc6 from scratch
  - Train bounding box regression head
- During tracking: fine-tune all fc layers

$$\begin{bmatrix} \delta_{x_1} \\ \delta_{x_2} \\ \delta_{y_1} \\ \delta_{y_2} \end{bmatrix}$$



Initialization frame





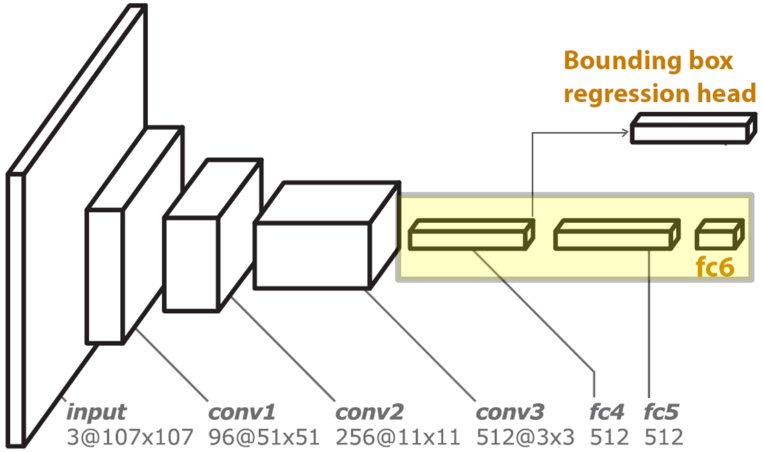
MDNet: Online tracking

- Sample target positions, classify, output the one with max score

Target localization

Target hard negative mining

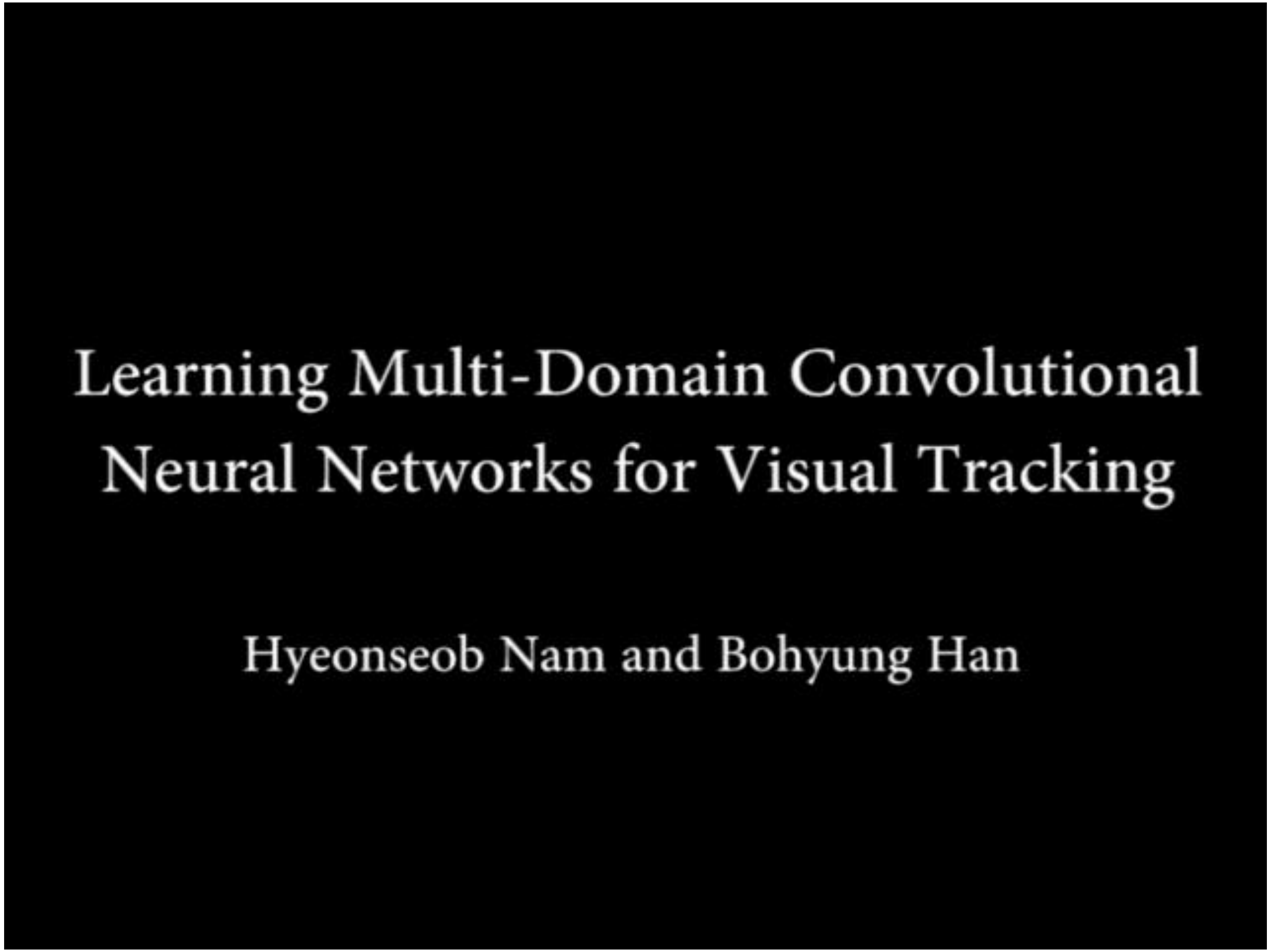
- Fine-tune all fc layers (fc4, fc5,fc6)
  - Hard negative mining
    (negative samples with a high “positive” score)
  - Short-term and long-term memory samples
  - Do not update during target loss



MDNet in action

- Remarkably robust

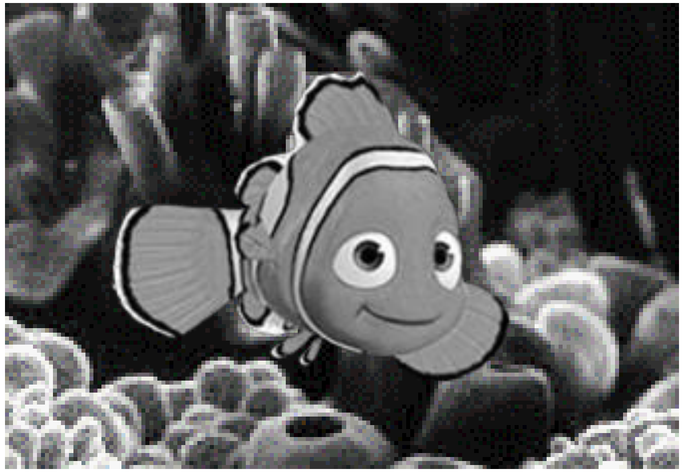
- I suspect, that smart training samples mining and careful updating *significantly contributes* to performance…



Nam and Han, Learning Multi-Domain Convolutional Neural Networks for Visual Tracking, CVPR2016
Jung, Son, Baek, Han, Real-Time MDNet, ECCV2018

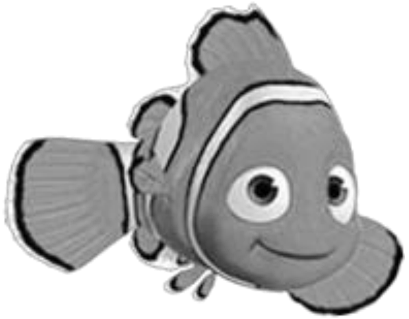
Recall the idea behind tracking by correlation

Image: $\boldsymbol{f}$



$\star$

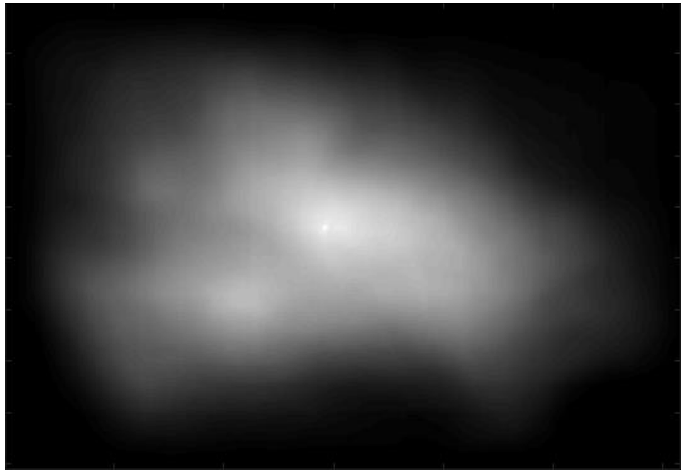
Template: $\mathbf{h}$



Using all grayscale image pixels (standard correlation)

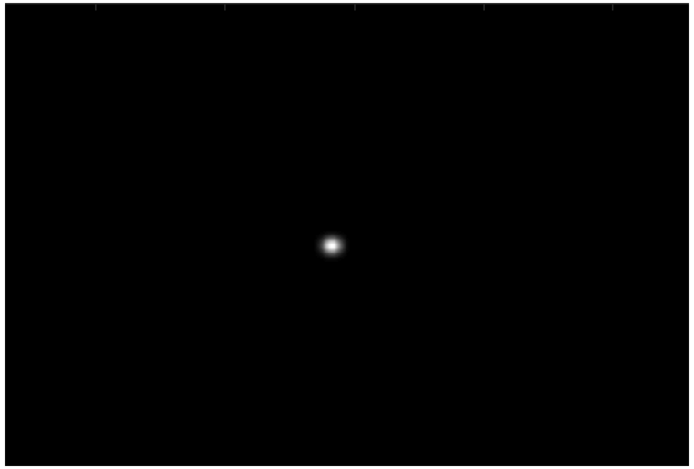
$=$

Correlation output: $\mathbf{g}' = \boldsymbol{f} \star \boldsymbol{h}$



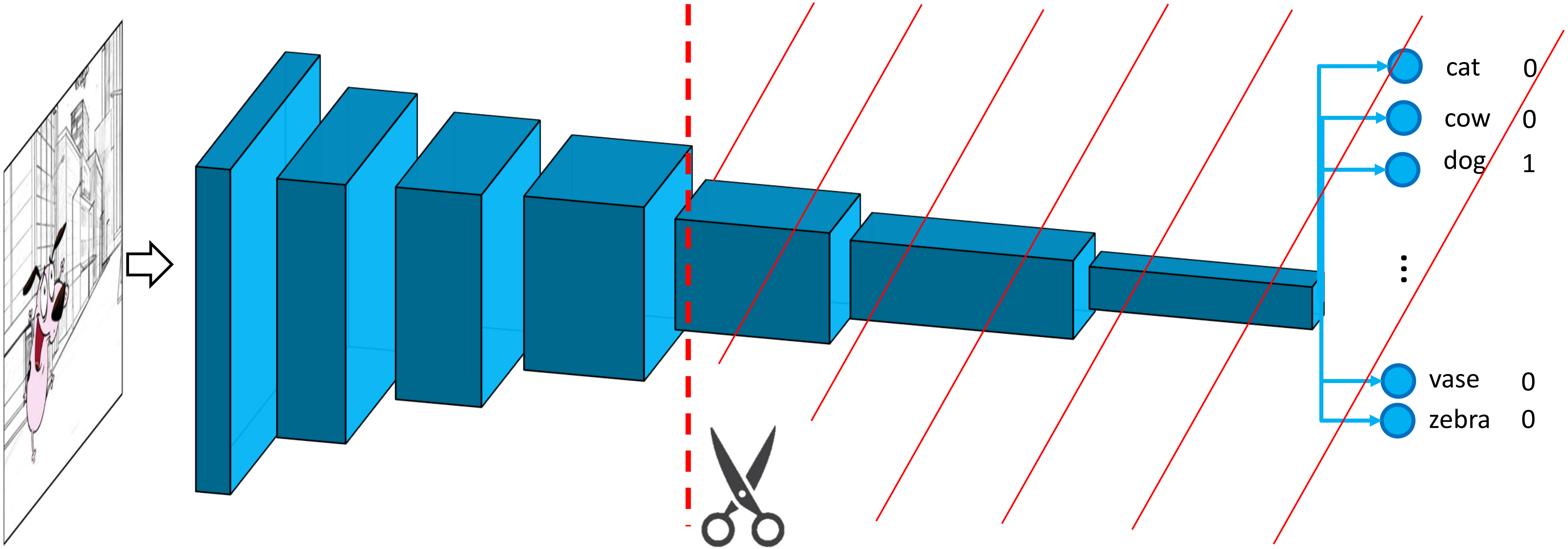
- Problem: *Intensity values are very weak features*
  - Correlation response not well expressed at target location
  - Tracking may quickly drift when the target appearance changes

Desired correlation output: $\mathbf{g}$



CNN as a feature extractor

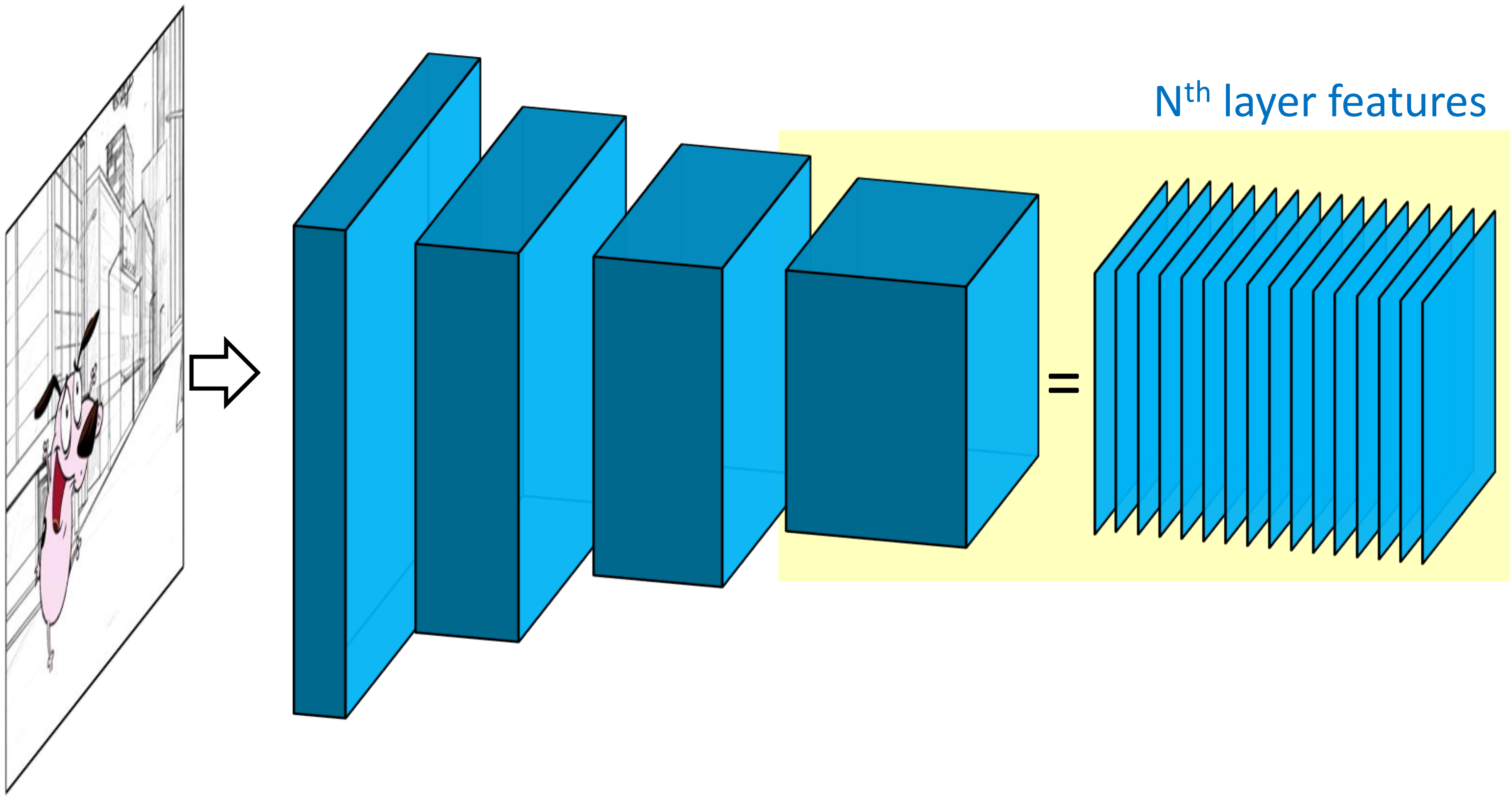
- Apply a CNN for object detection pretrained on Imagenet for many categories (~1000) and cut away the higher layers



[1] K He, X Zhang, S Ren, J Sun, Deep Residual Learning for Image Recognition, CVPR2016

CNN as a feature extractor

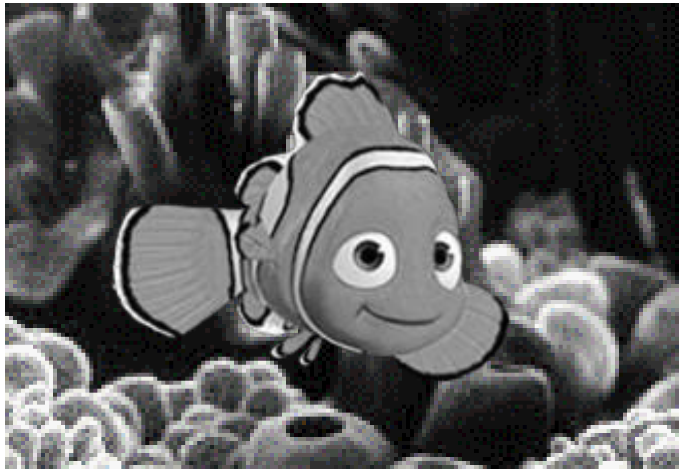
- Apply a CNN for object detection pretrained on Imagenet for many categories (~1000) and cut away the higher layers



[1] K He, X Zhang, S Ren, J Sun, Deep Residual Learning for Image Recognition, CVPR2016

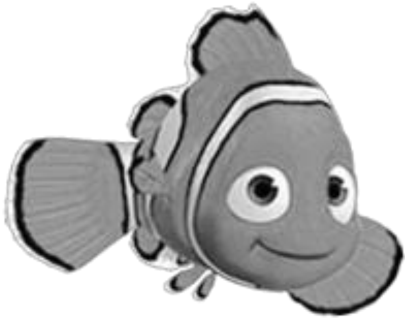
Robustifying template correlation by CNN features

Image: $\boldsymbol{f}$



$\star$

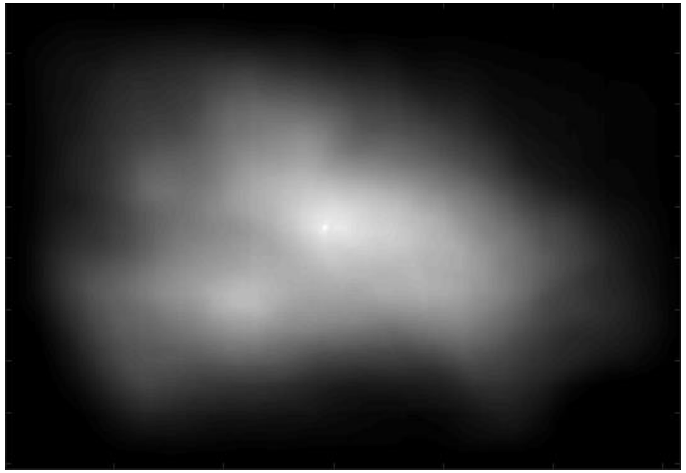
Template: $\mathbf{h}$

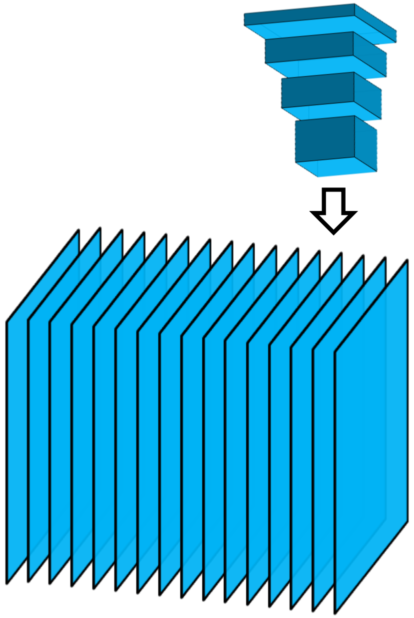


Using all grayscale image pixels
(standard correlation)

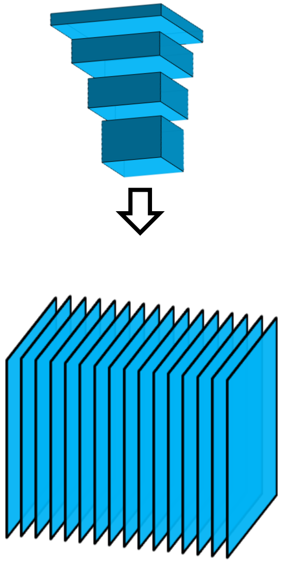
$=$

Correlation output: $\mathbf{g}' = \boldsymbol{f} \star \boldsymbol{h}$





$\star$

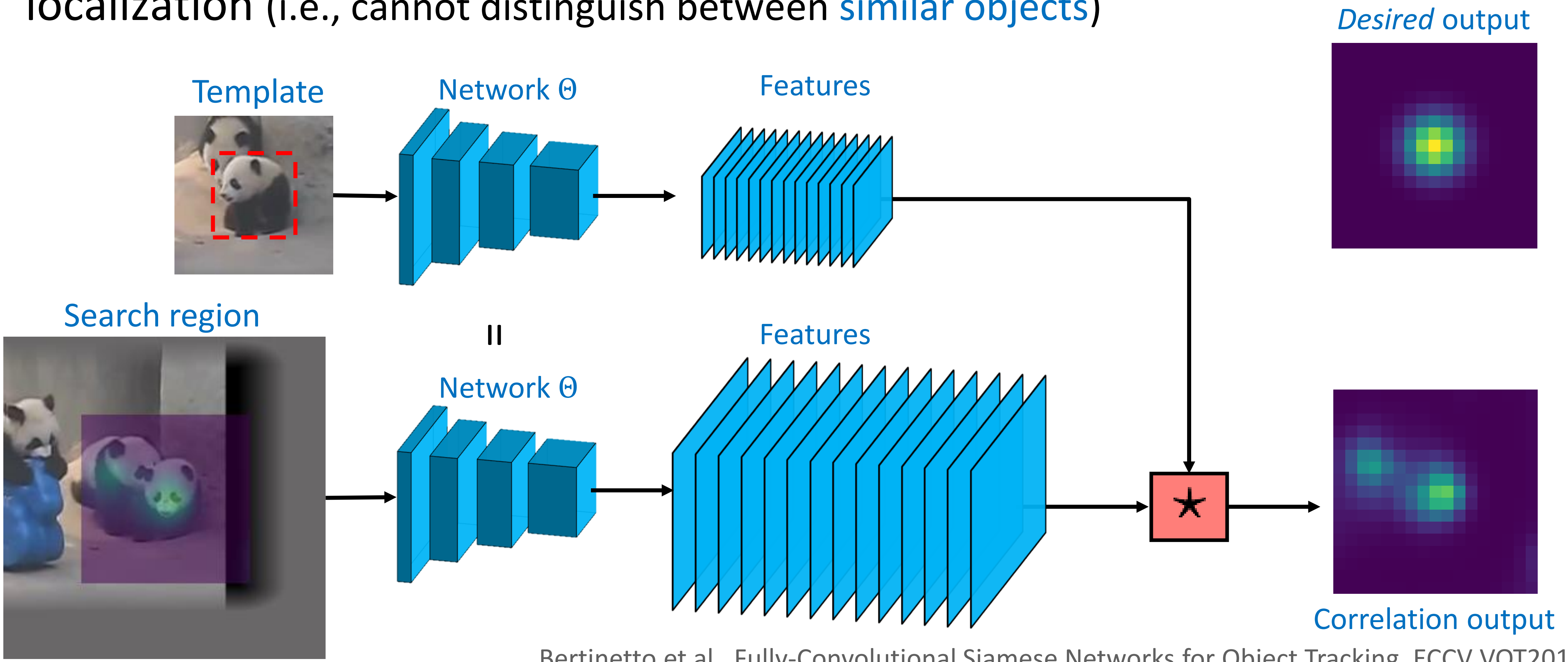


$=$

Correlation output

Robustifying template correlation by CNN features

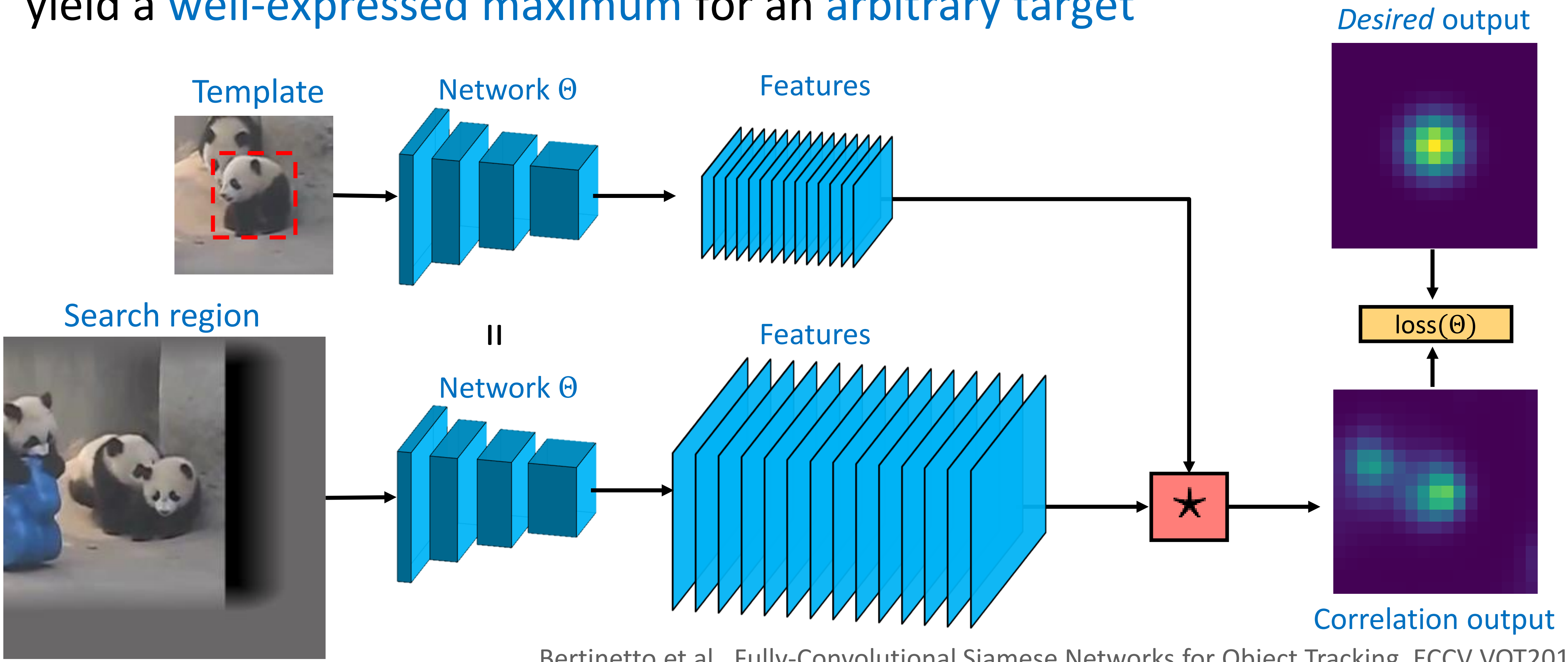
- Issue: CNN features were pre-trained for classification, not for discriminative localization (i.e., cannot distinguish between similar objects)



Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016

Robustifying template correlation by CNN features

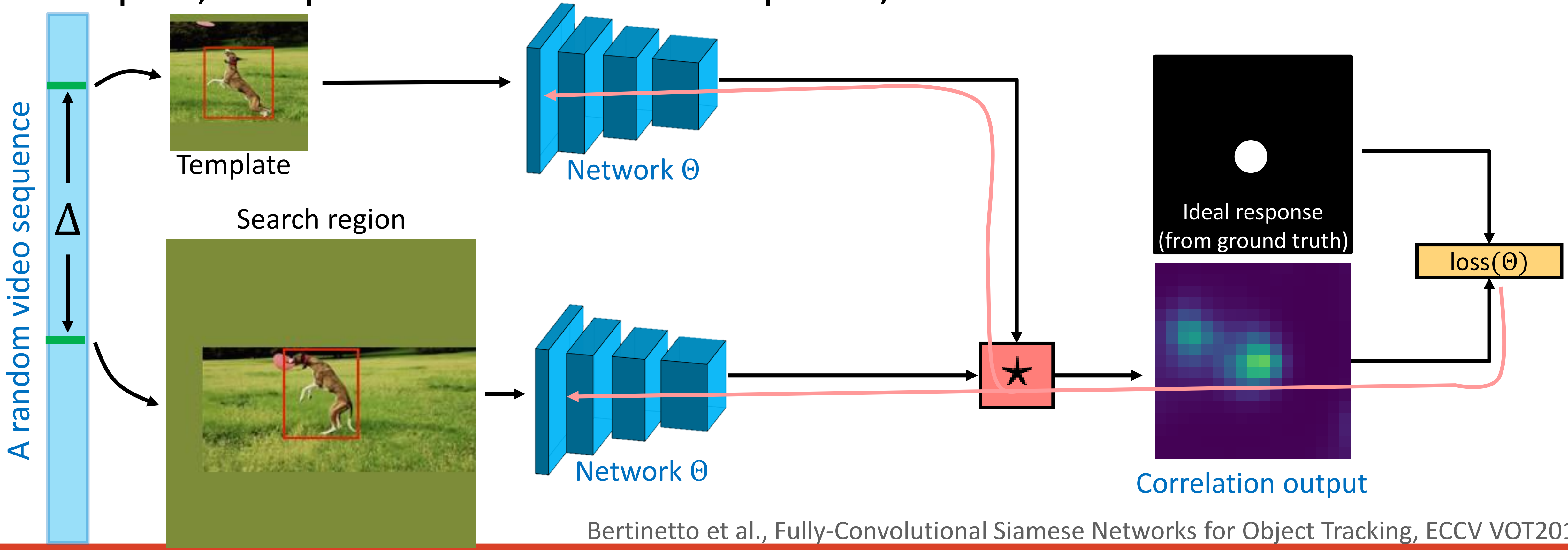
- Solution: pre-train the backbone parameters, such that the correlation will yield a well-expressed maximum for an arbitrary target



Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016

SiamFc (Siamese fully conv. net): Pre-training

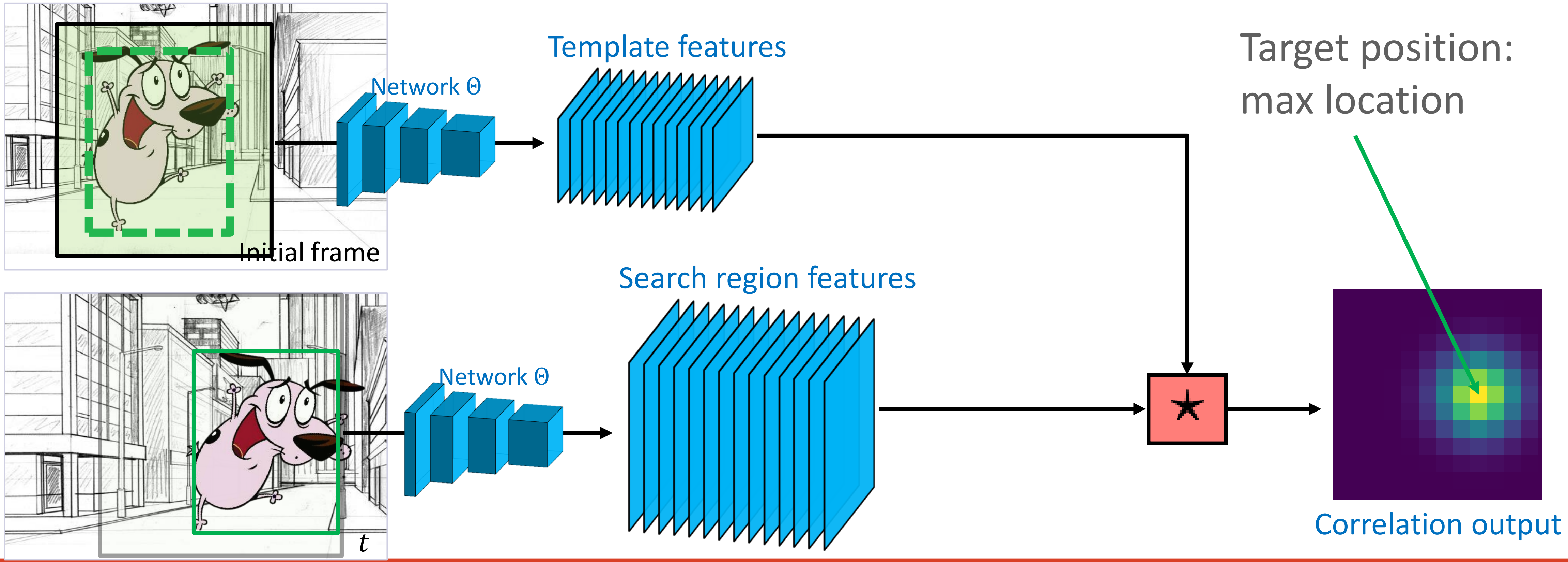
- ImageNet VID challenge – a video dataset with targets annotated
- Take many pairs of random images from the same sequence Δ frames apart, compute the correlation response, and minimize the loss w.r.t. Θ



Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016

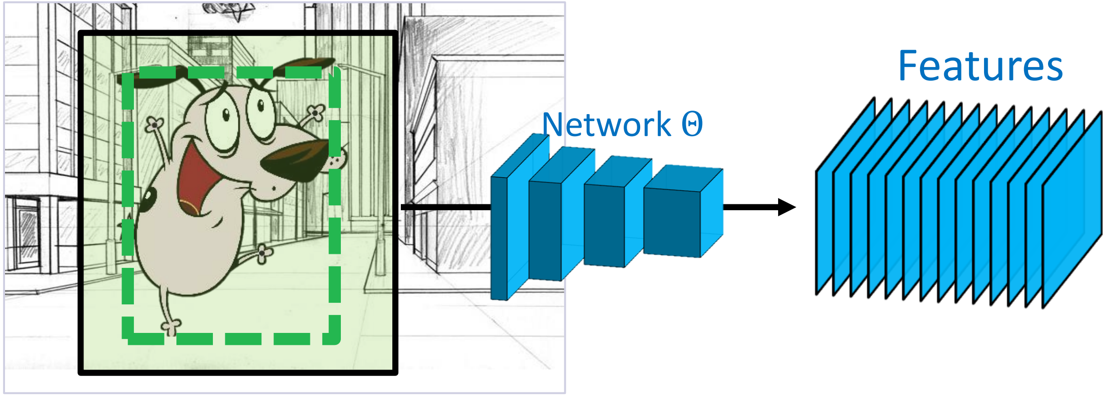
SiamFc: Tracking

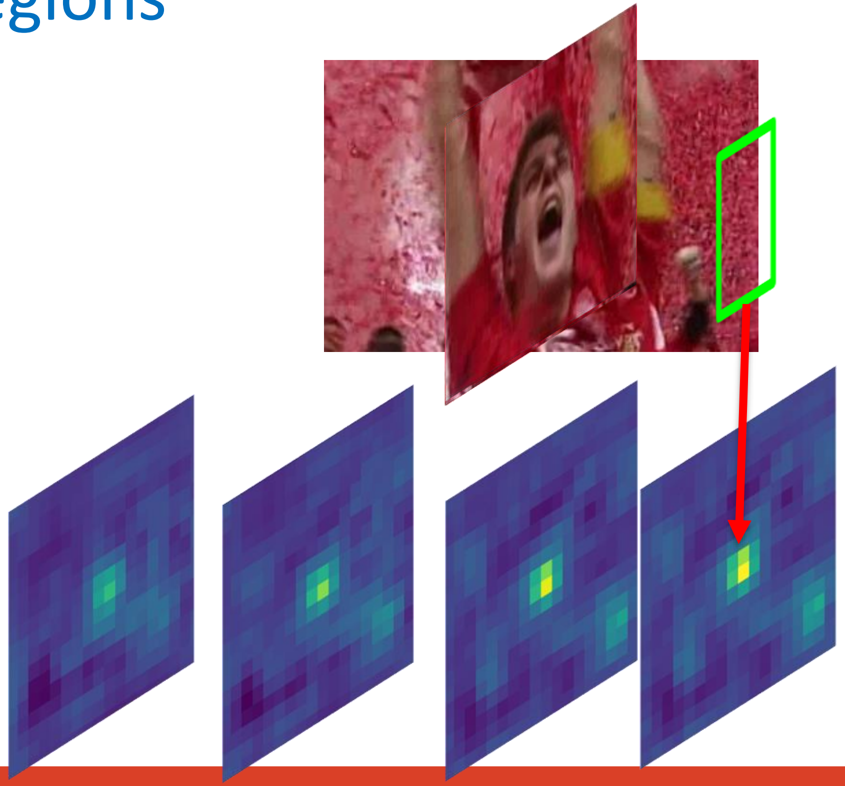
- Template extracted in the first frame
- Target localization in t-th frame: maximum of the correlation between search region and the template (both encoded by CNN)



SiamFc: Scale estimation

- Template extracted in the first frame
- Target localization in t-th frame: correlate with the template on several resized search regions





Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016

SiamFc: Tracking examples

- A fully-convolutional Siamese network
  Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016
- Template is *not updated* during tracking
- Super fast: ~60fps

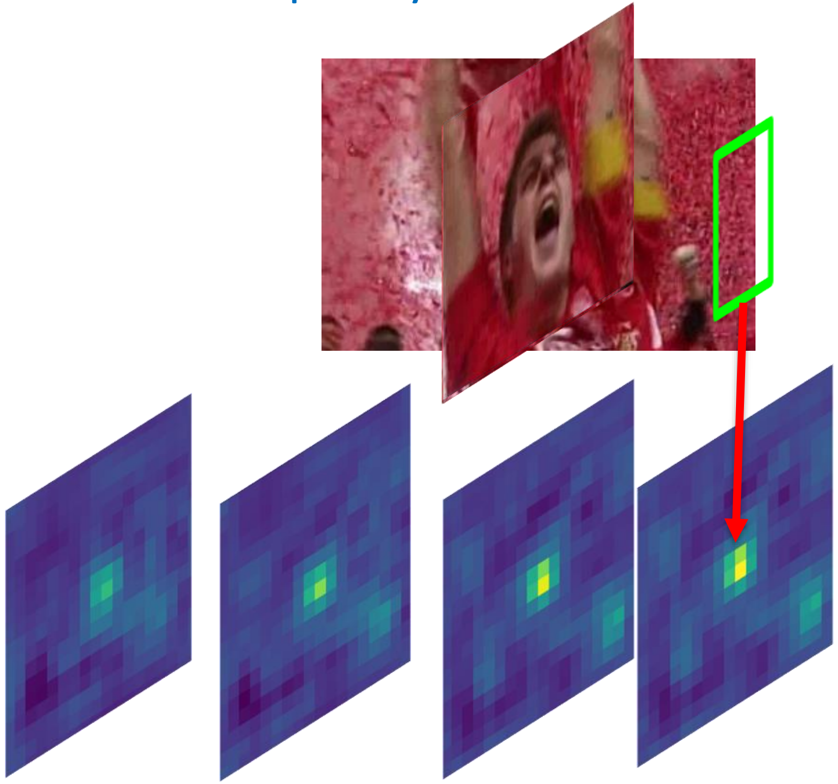
- Recent work on template updating
  Zhang et al., Learning the Model Update for Siamese Trackers, ICCV2019
- Extension with segmentation
  Wang et al. Fast Online Object Tracking and Segmentation: A Unifying Approach. CVPR 2019

Video credit: Jack Valmadre

Issues with bounding box estimation

- Standard approach: resize the input image to several scales and correlate on each

Explicitly test several scales



- Poor approximation of the aspect change…

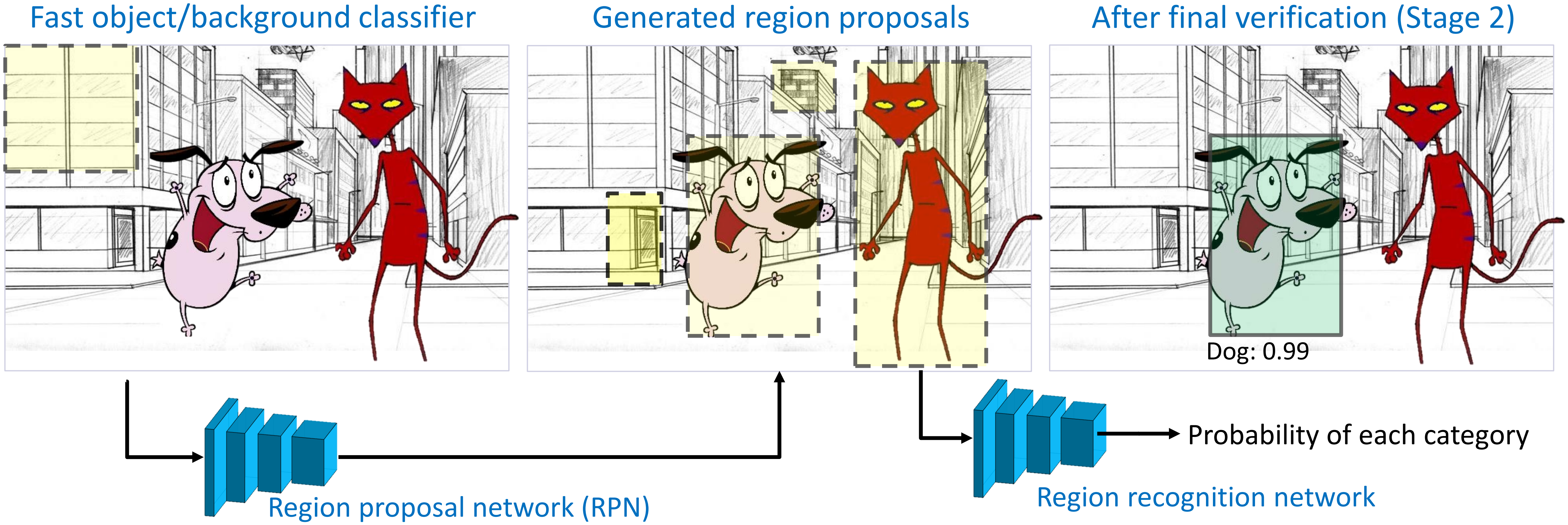
A standard approach for object detection

Ren et al., Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, NIPS215

- Stage 1: identify potential regions with objects – region proposals (RP)

  (requirement: the RP classifier has to be fast)

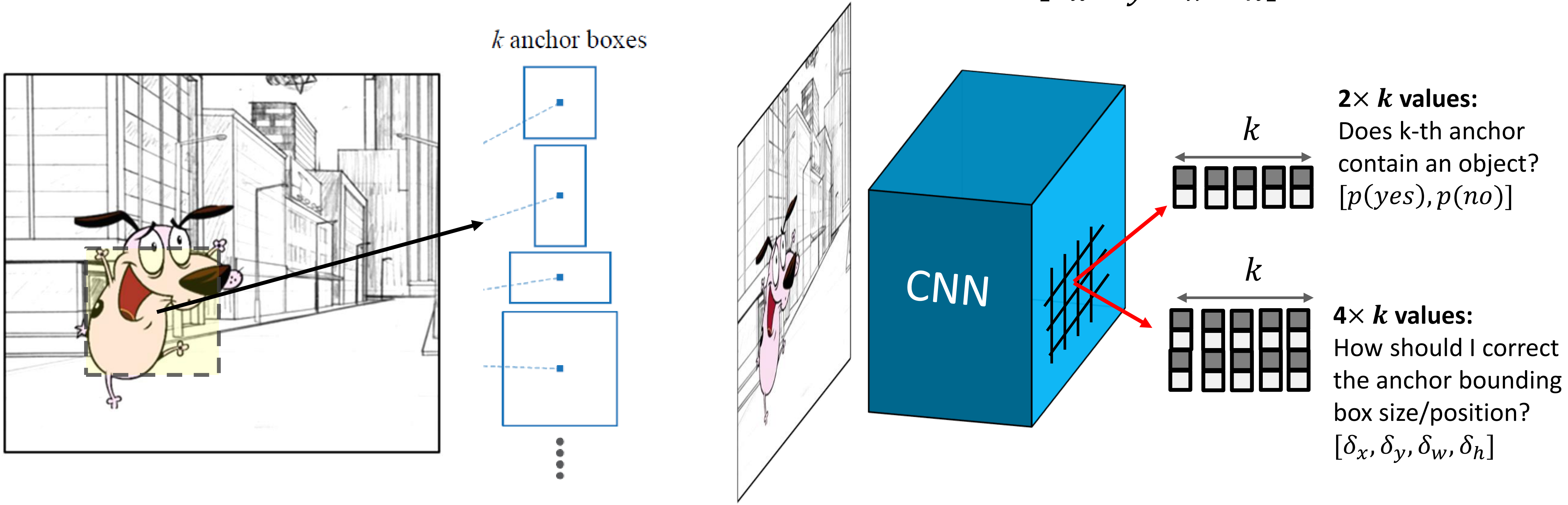
- Stage 2: classify each selected region by a strong classifier into categories



The region proposal network (RPN)

Ren et al., Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, NIPS215

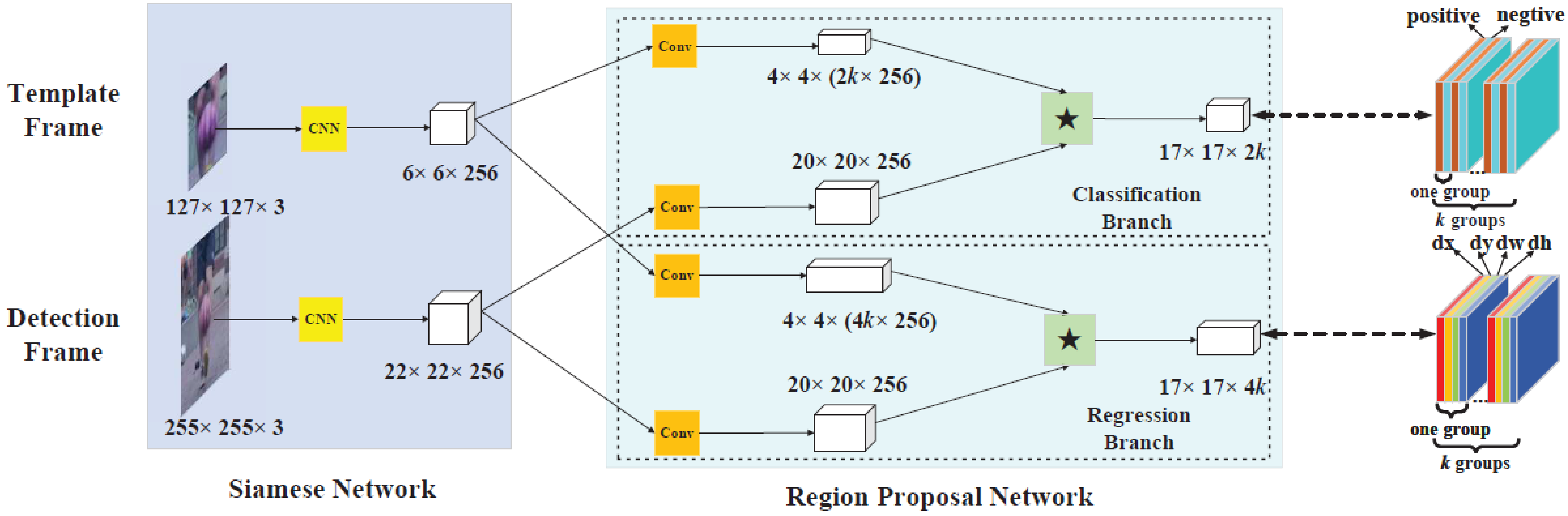
- At each location: *test for k bounding box shapes*
  - Tests a hypothesis that a certain shape bounding box is positioned there
  - Predicts “delta” coordinates to that hypothesized box $[\delta_x, \delta_y, \delta_w, \delta_h]$



SiamRPN – an RPN added to a Siamese tracker

Li et al., High Performance Visual Tracking with Siamese Region Proposal Network, CVPR2018

- Issue: the standard RPN is trained for general object detection
- Solution: a region proposal network is modulated by the template so that region proposals get specialized for the template



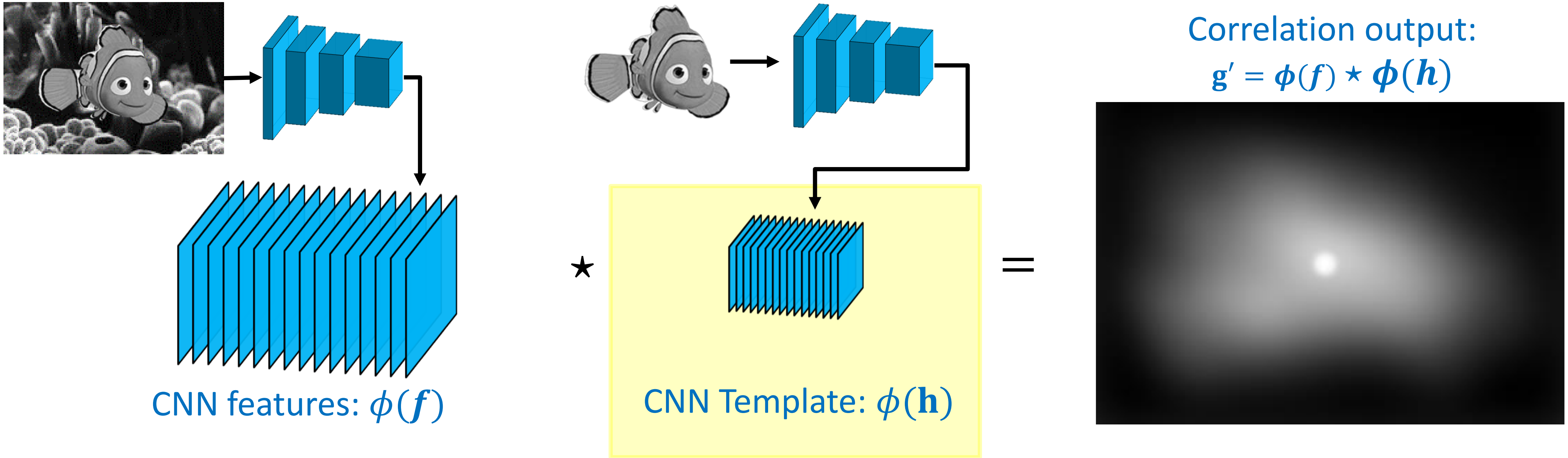
SiamRPN: Tracking example

- Aspect change is well addressed
- 160fps (PyTorch, PC with an Intel i7, 12G RAM, Nvidia GTX 1060)
- Improved version proposed recently [1]

[1] Li et al., SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks, CVPR2019

Siamese networks issues

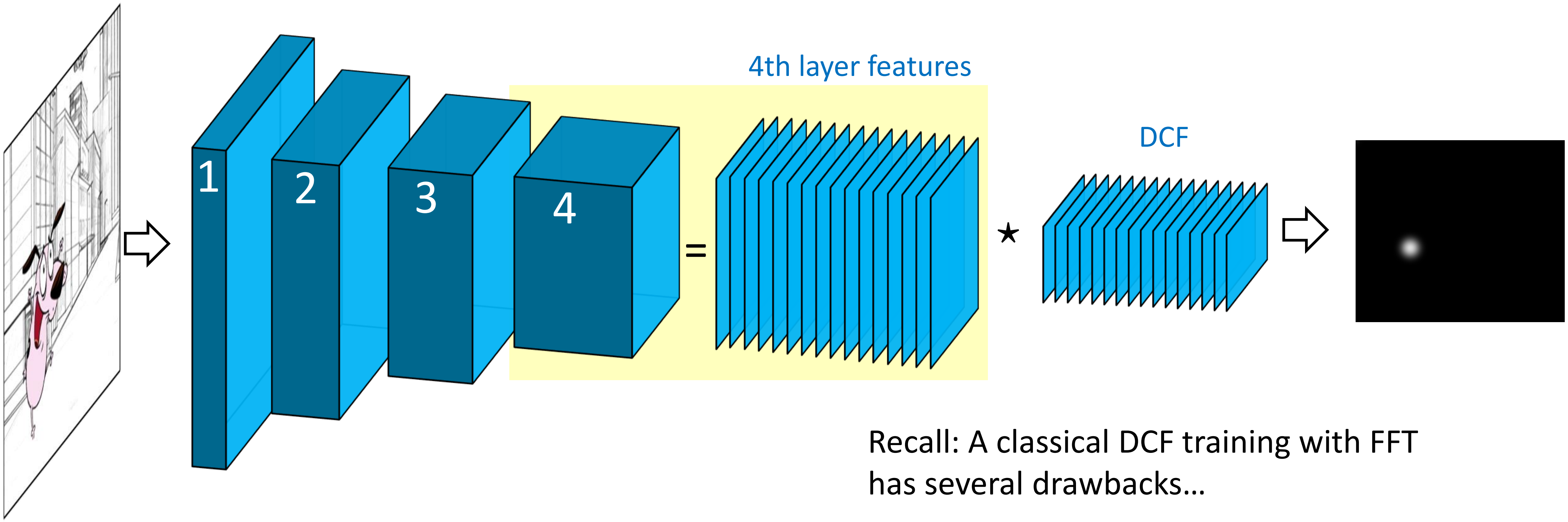
- Localization by a *generative* template – not by a *discriminative* template
- Cannot focus on features that separate the *selected target* from the background



Why not learn a discriminative template?

Training a DCF on CNN features

- Apply a Resnet18 [1] pretrained on ImageNet and consider the output features of the 4^{th} layer

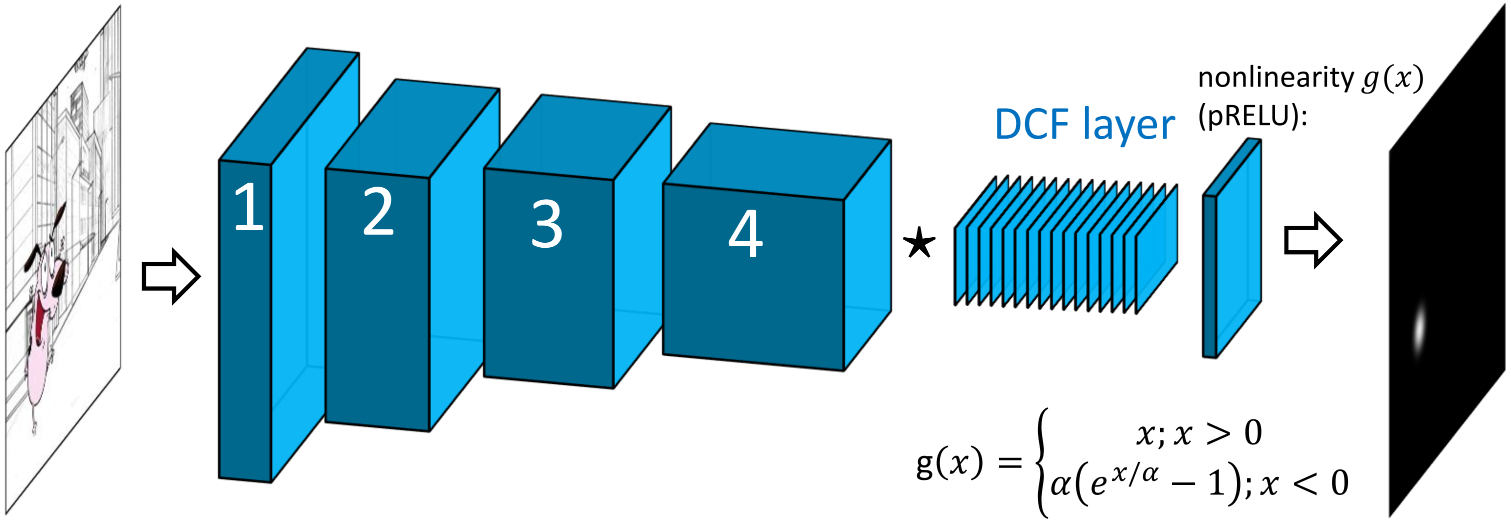


Recall: A classical DCF training with FFT has several drawbacks…

[1] K He, X Zhang, S Ren, J Sun, Deep Residual Learning for Image Recognition, CVPR2016

A DCF as a CNN layer

- A DCF of any size can be formulated as a single output correlation layer.
- Any nonlinear transformation to the output can be enforced.



Issue: how to train the DCF?

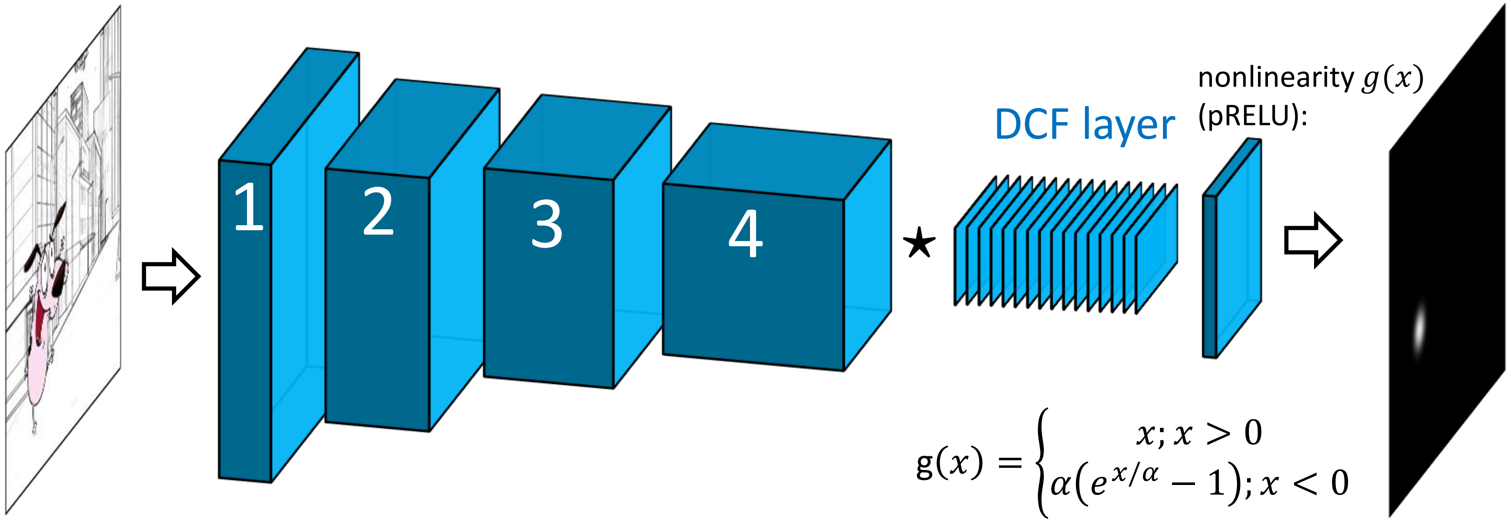
- The DCF cost function:

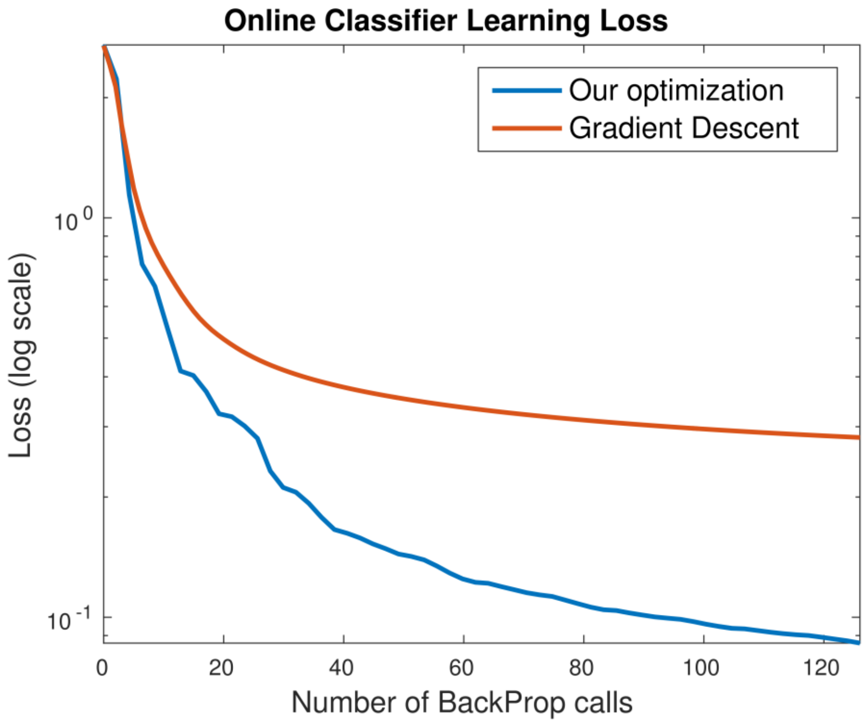
$$L(w) = \sum_{j=1}^{m} \gamma_j \| f(x_j; w) - y_j \|^2 + \sum_k \lambda_k \| w_k \|^2 .$$

[1] Danelljan et al., ATOM: Accurate Tracking by Overlap Maximization, CVPR2019

A DCF as a CNN layer

- Efficient training by a conjugated gradient descent, implemented via backprop methods already in CNN – fully trainable within the CNN





- Introduced as part of ATOM [1].

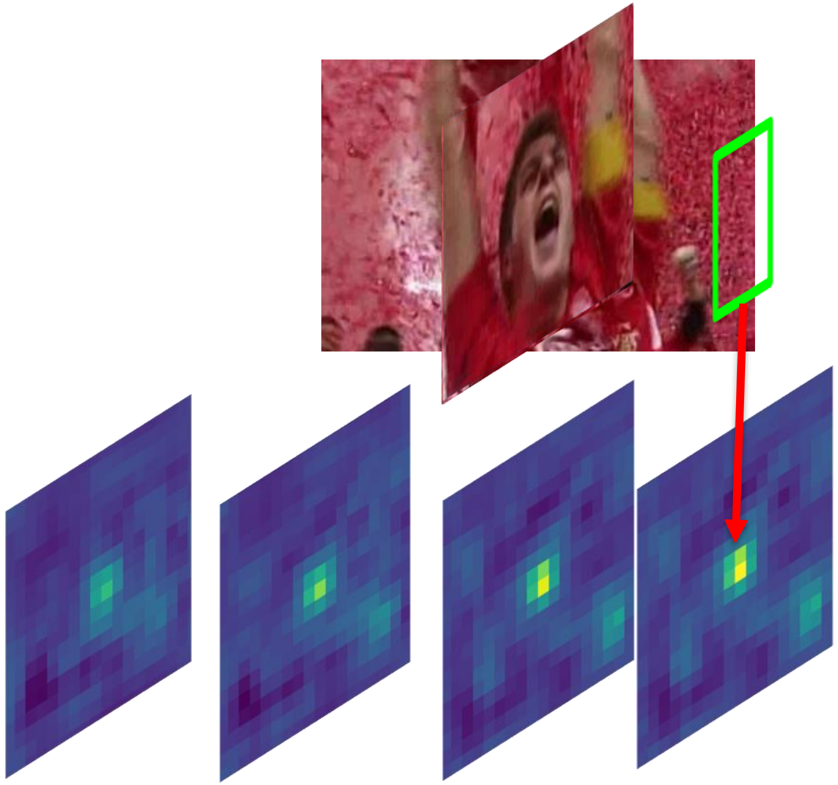
[1] Danelljan et al., ATOM: Accurate Tracking by Overlap Maximization, CVPR2019

Graph from: Martin Danneljan

Recall the issues with bounding box estimation

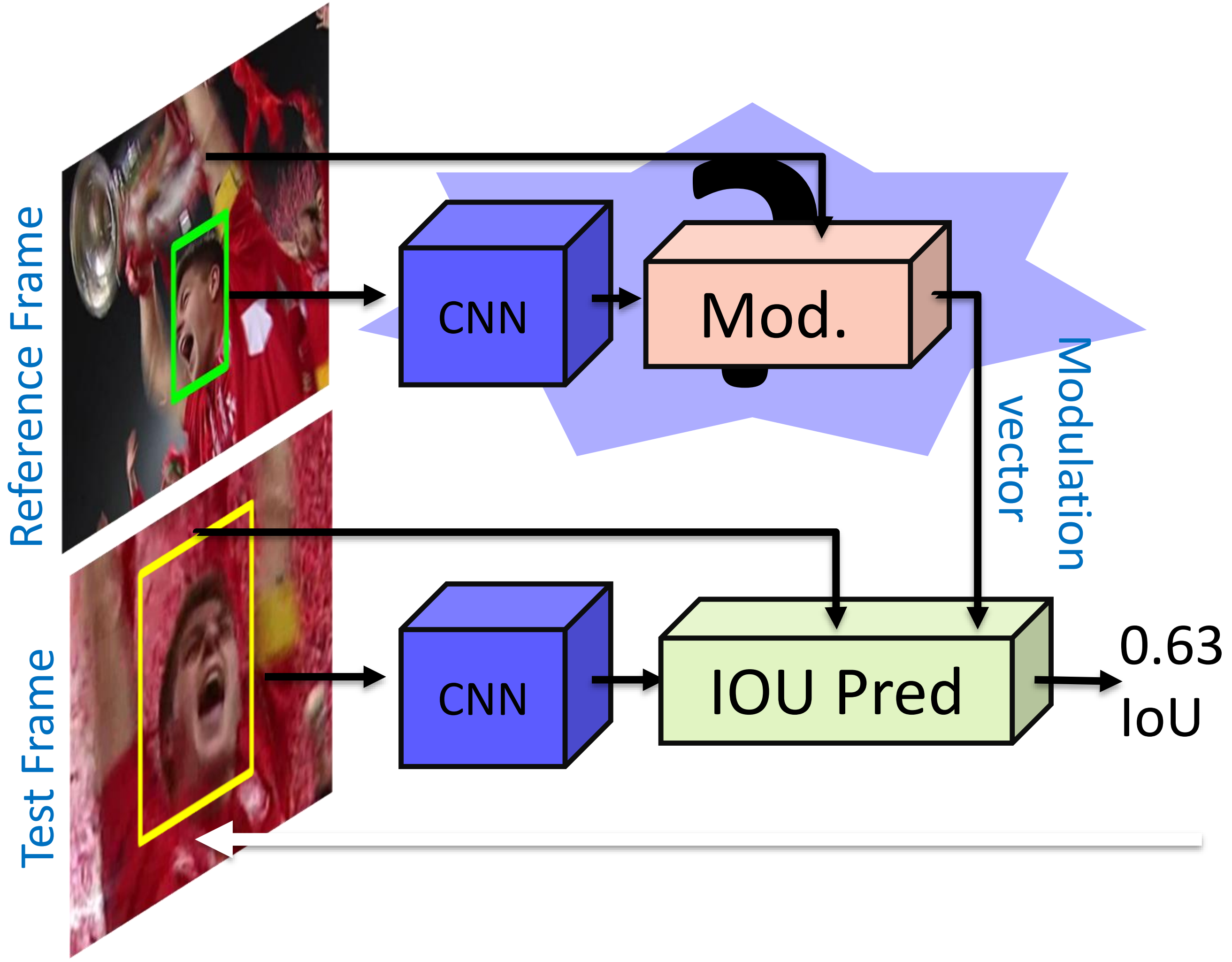
- Standard approach: Apply a DCF to differently resized images
- Poor approximation of the aspect change…

Explicitly test several scales



Apply another CNN for bounding box fitting

- Could apply an IoU-net [1] to predict the box fit (without knowing the GT)
- But IoU-net is trained for object detection and is not aware of the selected target!
- A modification was proposed by [2].



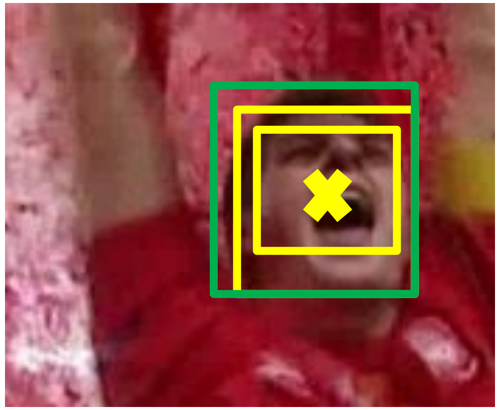
[1] IOUNet: B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang. Acquisition of localization confidence for accurate object detection. In ECCV, 2018
[2] Danelljan et al., ATOM: Accurate Tracking by Overlap Maximization, CVPR2019

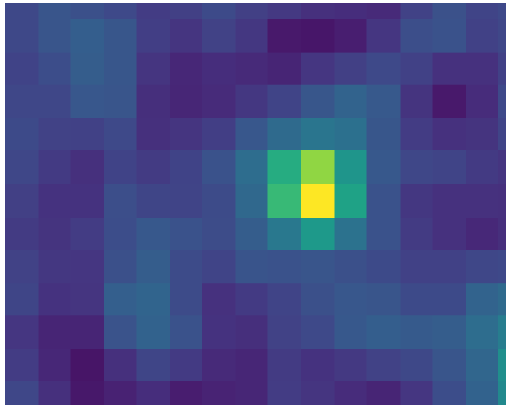
ATOM: Accurate Tracking by Overlap Maximization

Initial frame

Frame t

Frame t+1





1. Approximately localize by the deep DCF
2. Generate the proposal at DCF output
3. Refine the proposal by the modified IoU-net
4. Update the deep DCF

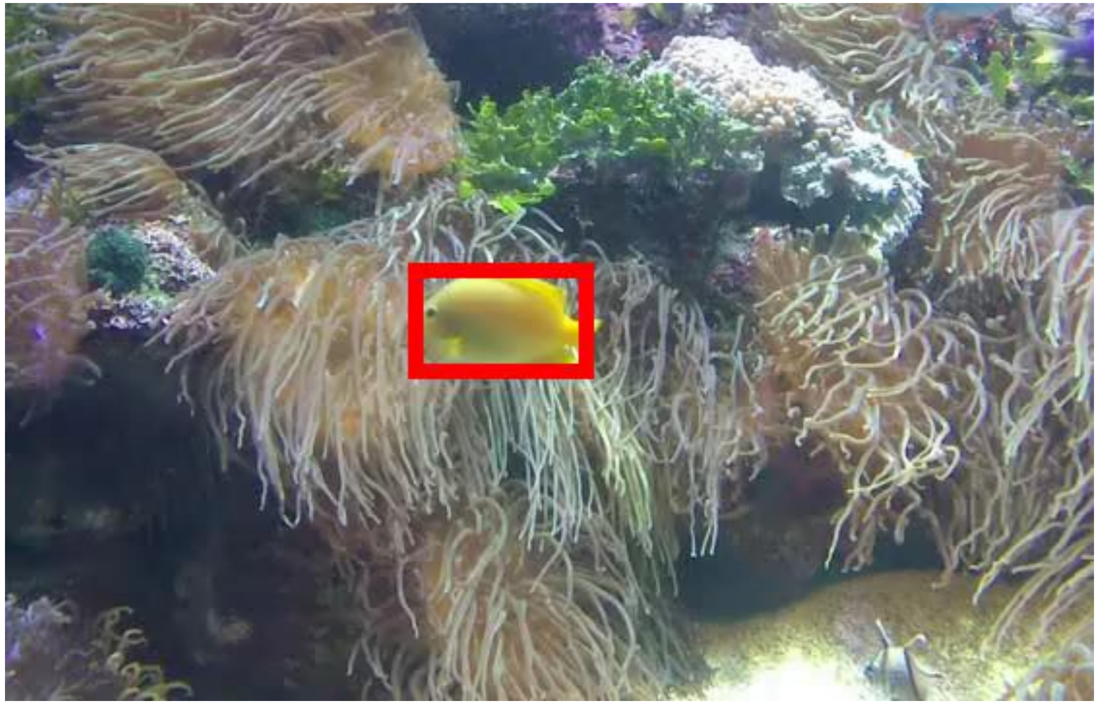
Slide adapted from: Martin Danneljan

ATOM and beyond

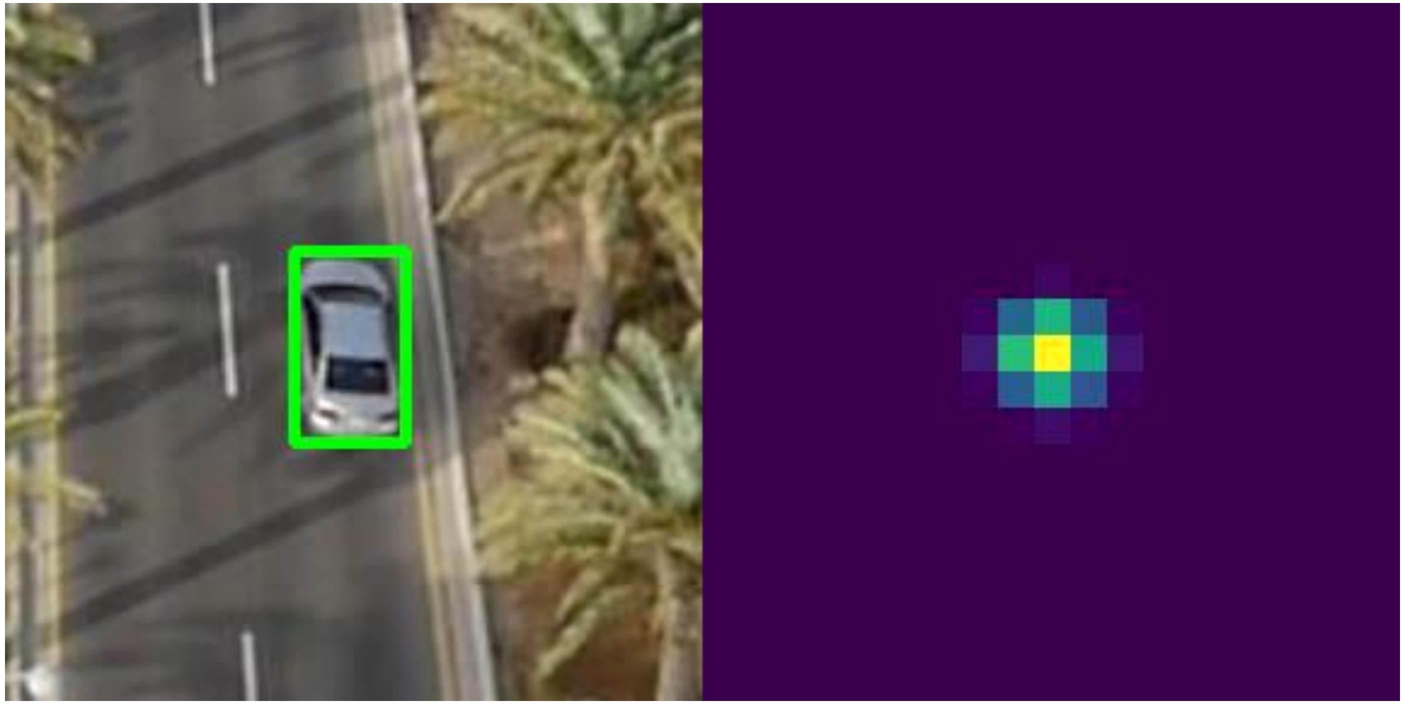
- The bounding box prediction network is trained on three huge datasets [1]
- Recent extension DiMp [1] (DCF training improved & hard negative mining added)

30-40fps

ATOM

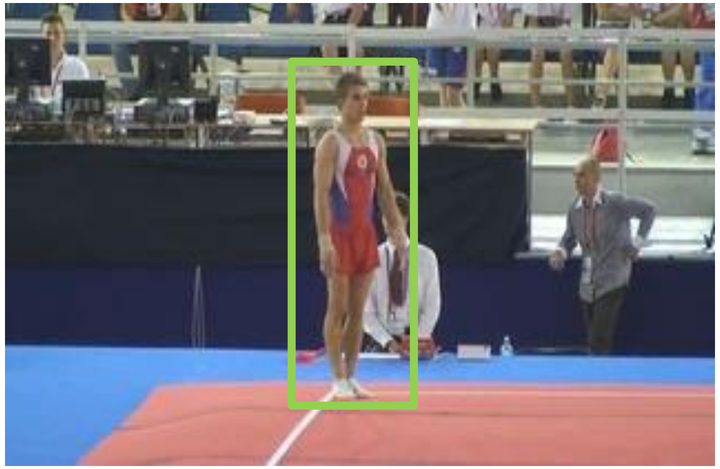


DiMp

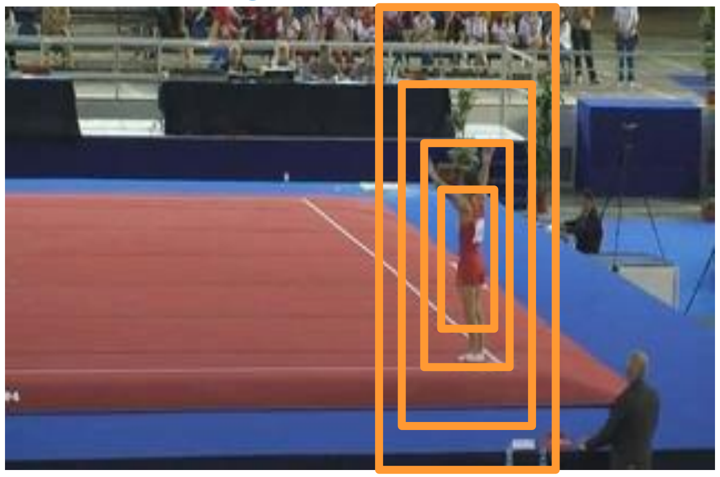


[1] Danelljan et al., ATOM: Accurate Tracking by Overlap Maximization, CVPR2019
[2] Bhat et al., Learning Discriminative Model Prediction for Tracking, ICCV2019

Challenges for Template-Based Trackers

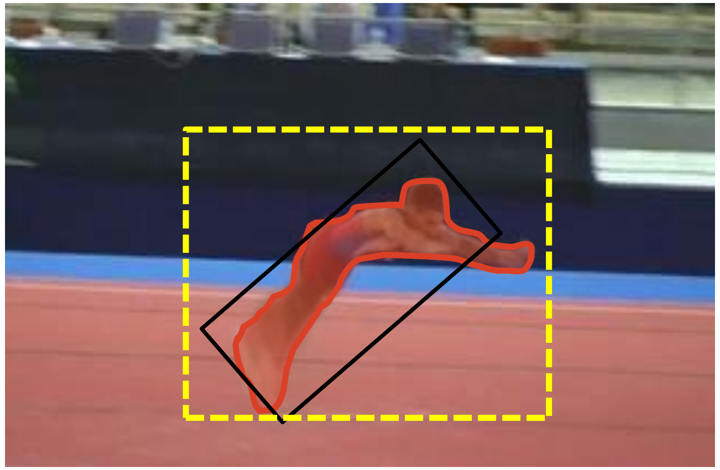


Scale change

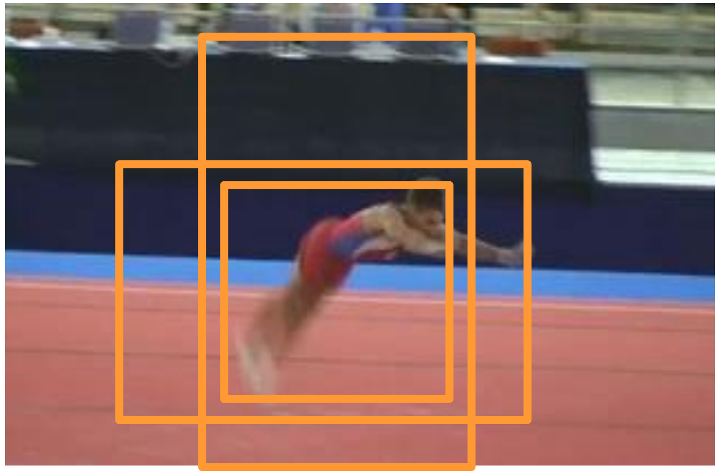


Exhaustive scale-space search [1,2]

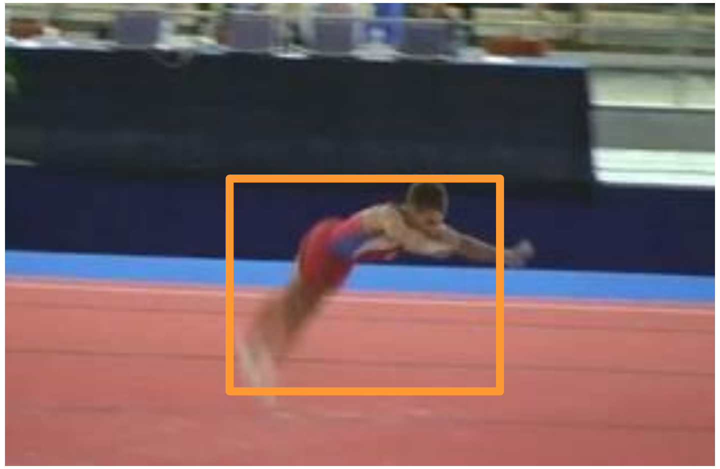
Rotated bbox (segmentation) [5]



Scale + aspect change



Region proposals [3]



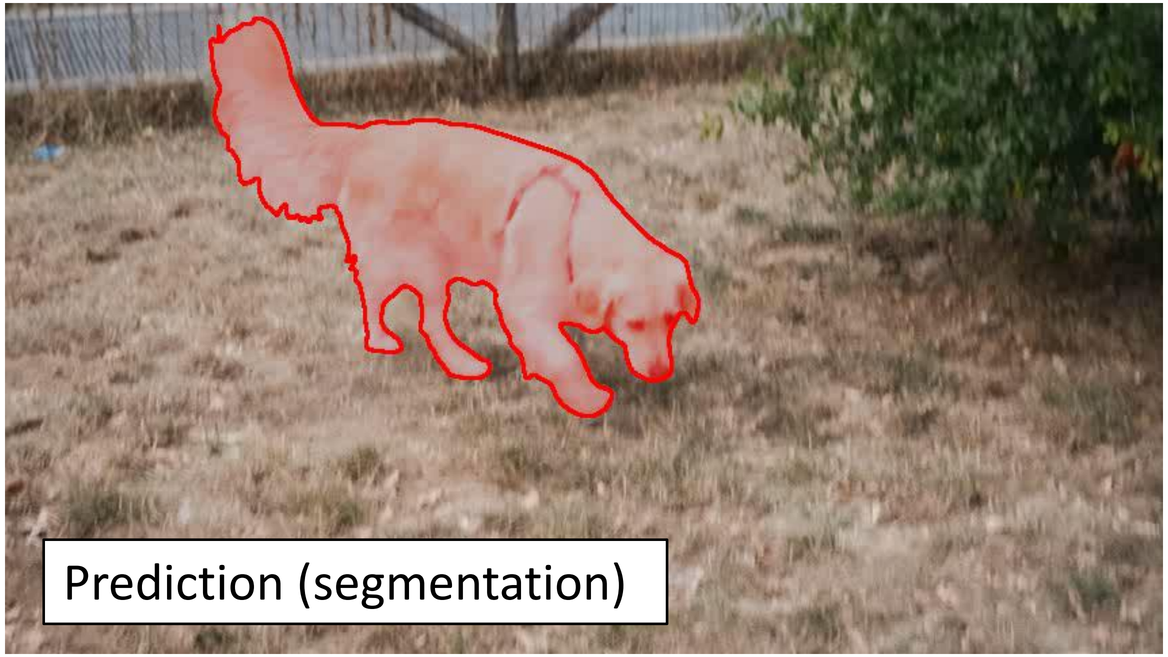
Bbox refinement, regression [4]

Drawbacks:

- Two stage approach
  prevents end-to-end learning
- Template is not discriminatively updated
  similar objects, significant appearance change

[1] Danelljan et al. ECO: Efficient Convolution Operators for Tracking. CVPR 2017
[2] Bertinetto et al. Fully-Convolutional Siamese Networks for Object Tracking. ECCVW 2016
[3] Li et al. SiamRPN++: Evolution of Siamese Visual Tracking With Very Deep Networks. CVPR 2019
[4] Danelljan et al. ATOM: Accurate Tracking by Overlap Maximization. CVPR 2019
[5] Wang et al. Fast Online Object Tracking and Segmentation: A Unifying Approach. CVPR 2019

Video Object Segmentation

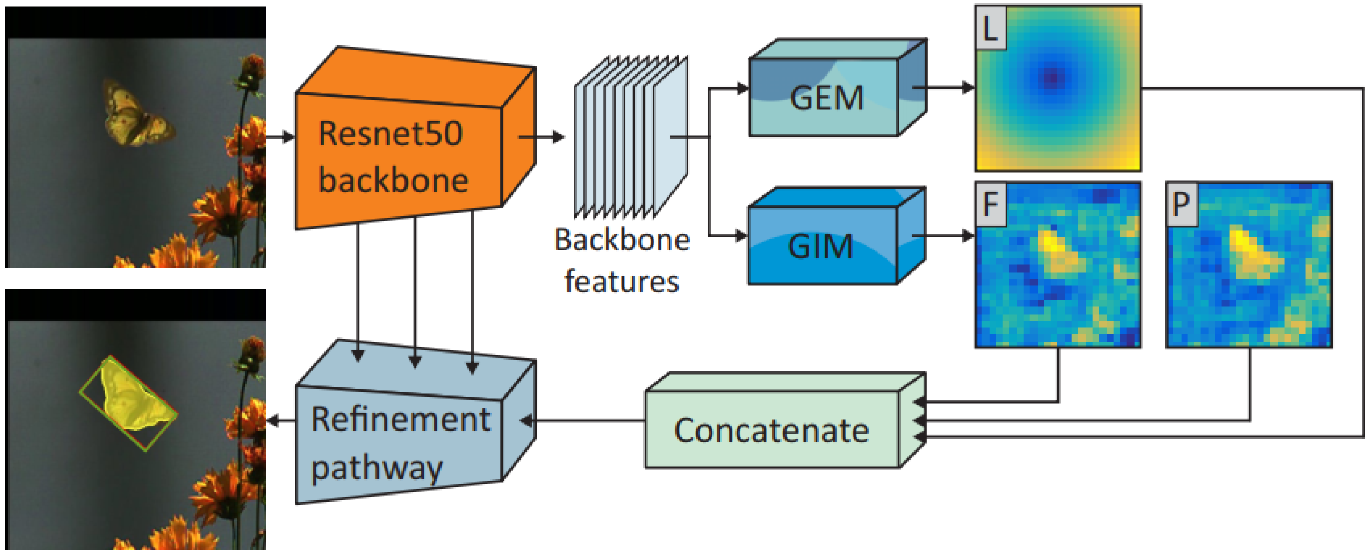


- Drawbacks:
  - Optimized for large objects
  - Cannot address significant appearance changes
  - Cannot address fast moving targets
  - Often computationally intensive

[1] Caelles et al. One-shot video object segmentation, CVPR 2017
[2] Chen et al. Blazingly fast video object segmentation, CVPR 2018
[3] Cheng et al. Fast and accurate online video object segmentation via tracking parts, CVPR 2018
[4] Hu et al. Video matxh: Matching based video object segmentation, ECCV 2018
[5] Voigtlaender et al. Online adaptation of convolutional neural networks for
video object segmentation, BMVC 2017
[6] Yang et al. Efficient video object segmentation via network modulation, CVPR 2018

Discriminative Tracking by Segmentation (D3S)

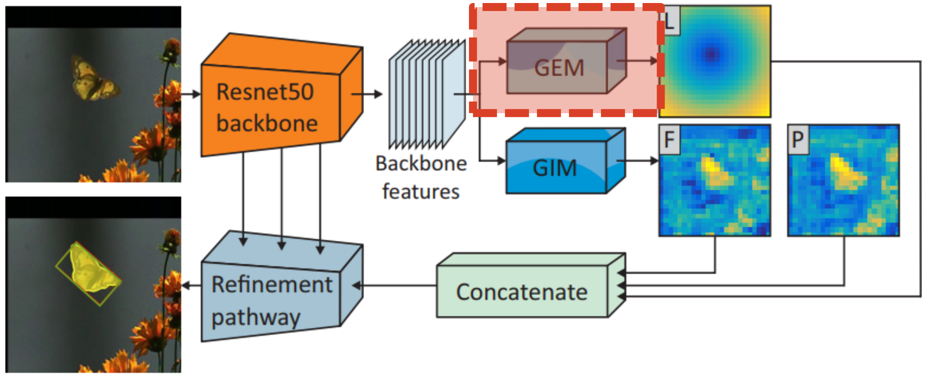
- Single-shot segmentation network
- Two target appearance models
  - Geometrically constrained Euclidean Model (GEM)
    Robust localization
  - Geometrically Invariant Model (GIM)
    Address significant deformations
- Fusion for accurate segmentation (Refinement pathway)
- Bounding box fitted to the mask (if required)

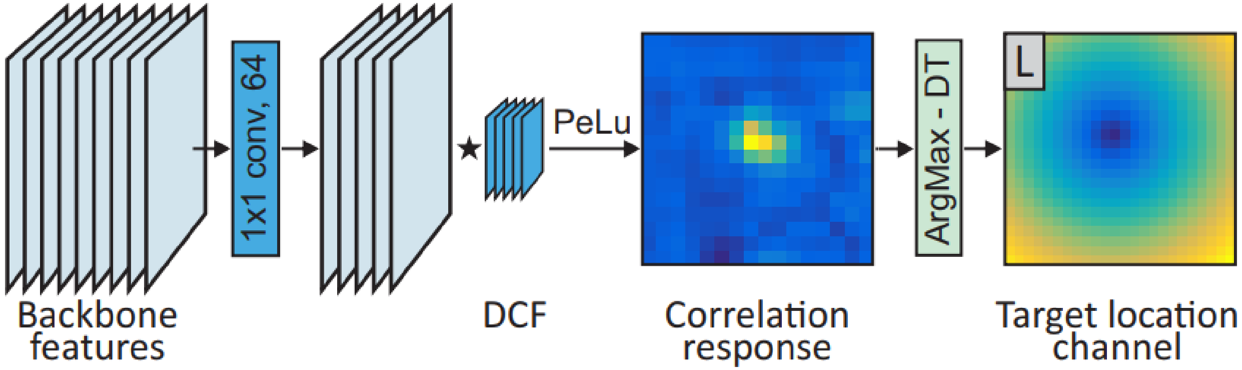


Lukežič, Matas, Kristan, D3S -- A Discriminative Single Shot Segmentation Tracker, CVPR2020

D3S: Geometrically Constrained Euclidean Model (GEM)

- Deep discriminative correlation filter (DCF) formulation [1]
- Localization:
  - Correlation response: target center likelihood
  - Required for segmentation: per pixel target region likelihood ⟶ Distance transform

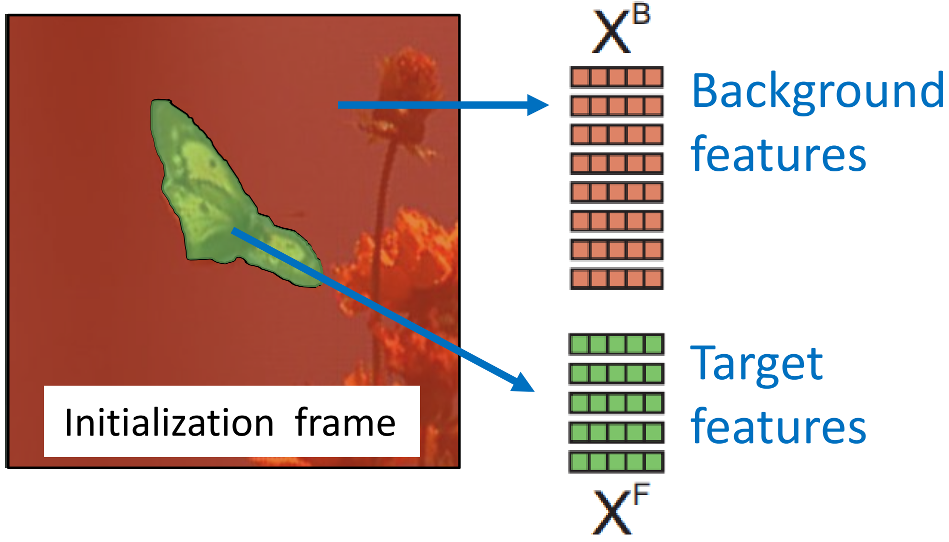


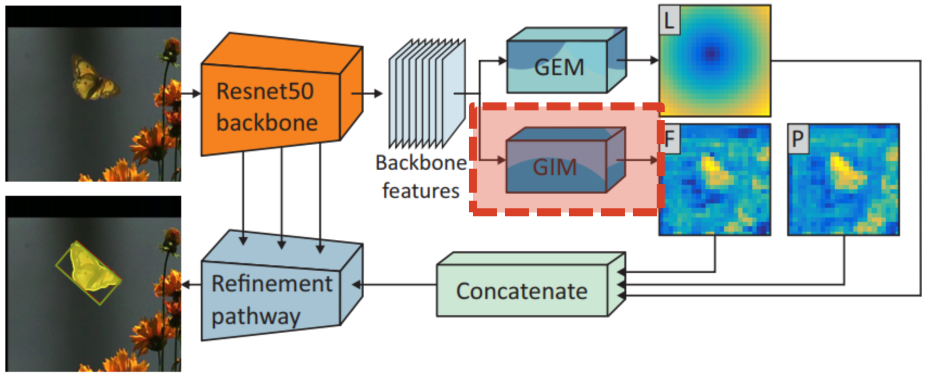


[1] Danelljan et al. ATOM: Accurate Tracking by Overlap Maximization. CVPR 2019

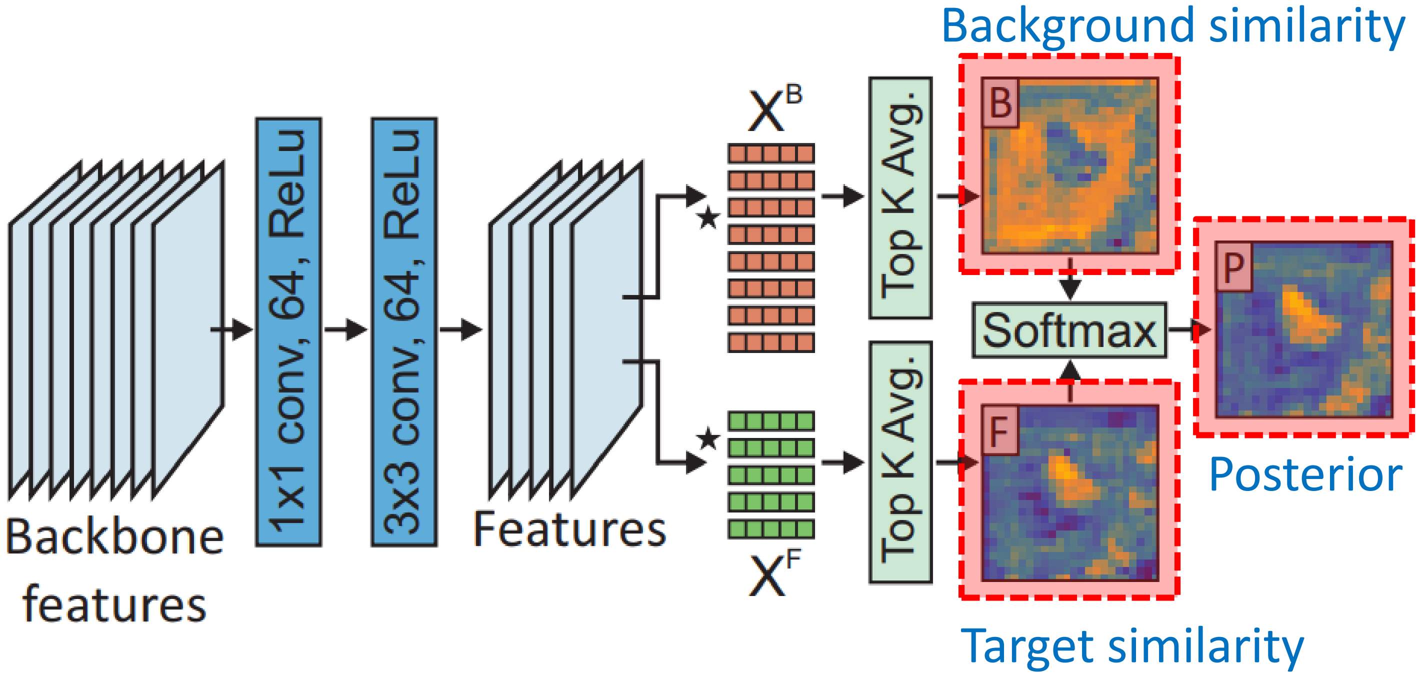
D3S: Geometrically Invariant Model (GIM)

Two sets of features extracted on the first frame



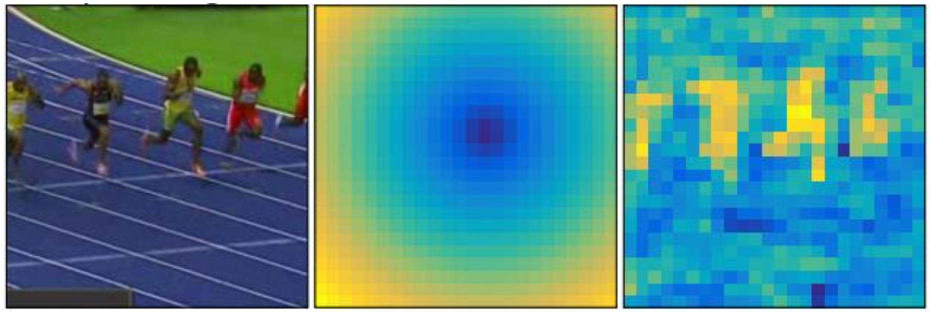


- Localization:
  - Per-pixel cosine similarity with X^B and X^F



D3S: Refinement Pathway

Input image GEM output GIM output

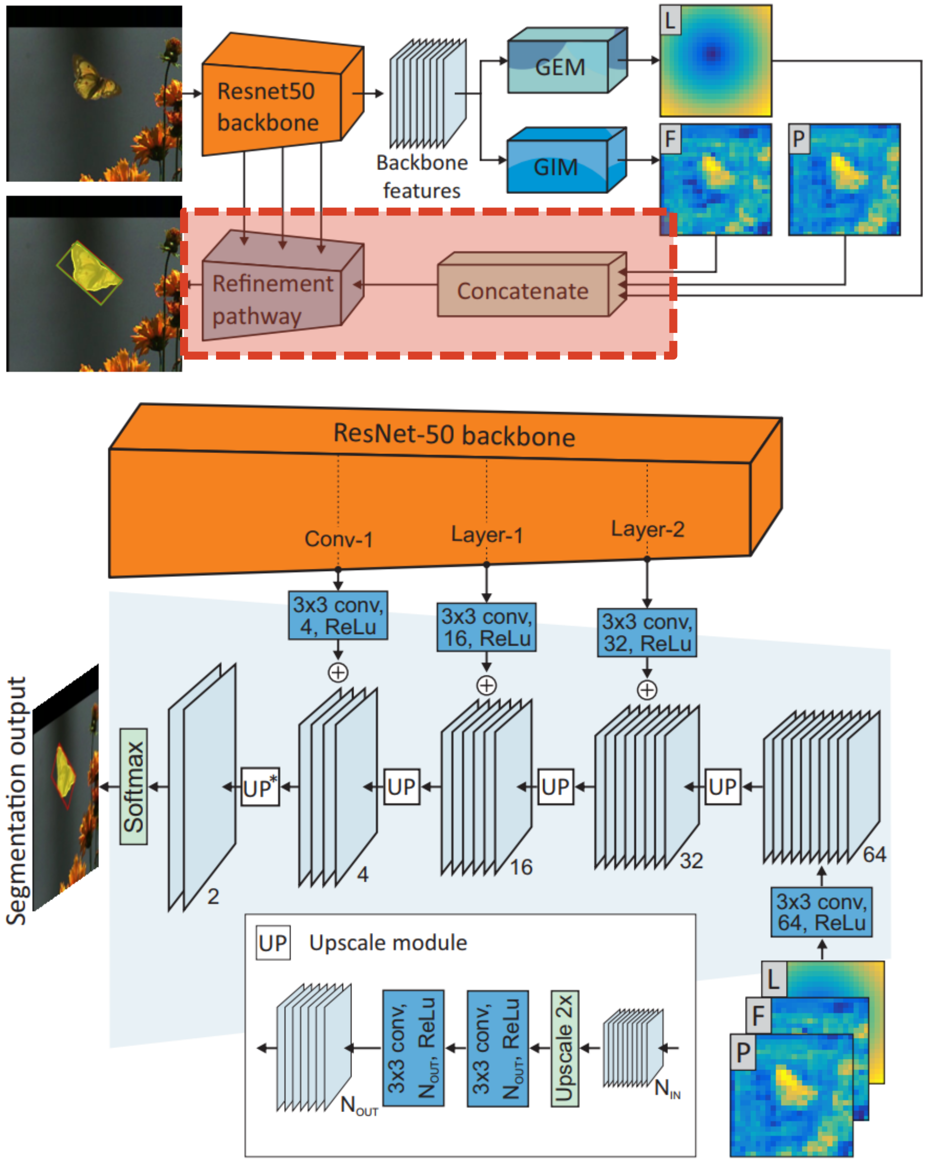


- Robust localization (selector)
- Not accurate (target center only)

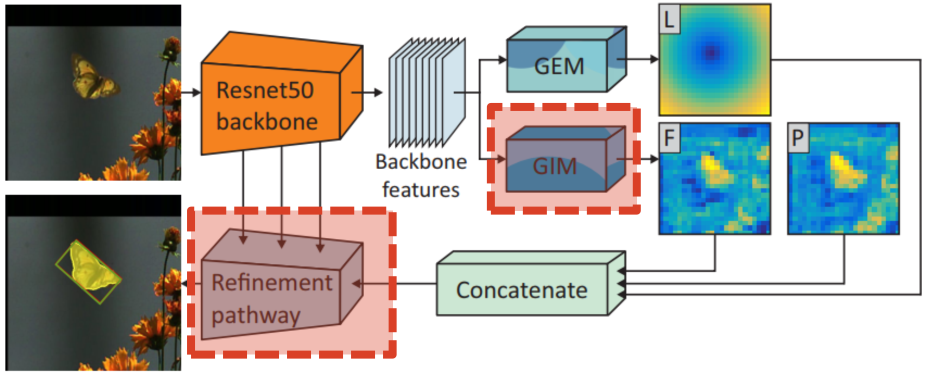
- Per-pixel segmentation
- Cannot distinguish similar targets

Low resolution

(Due to backbone reduction)



D3S: Discriminative Single Shot Segmentation Tracker

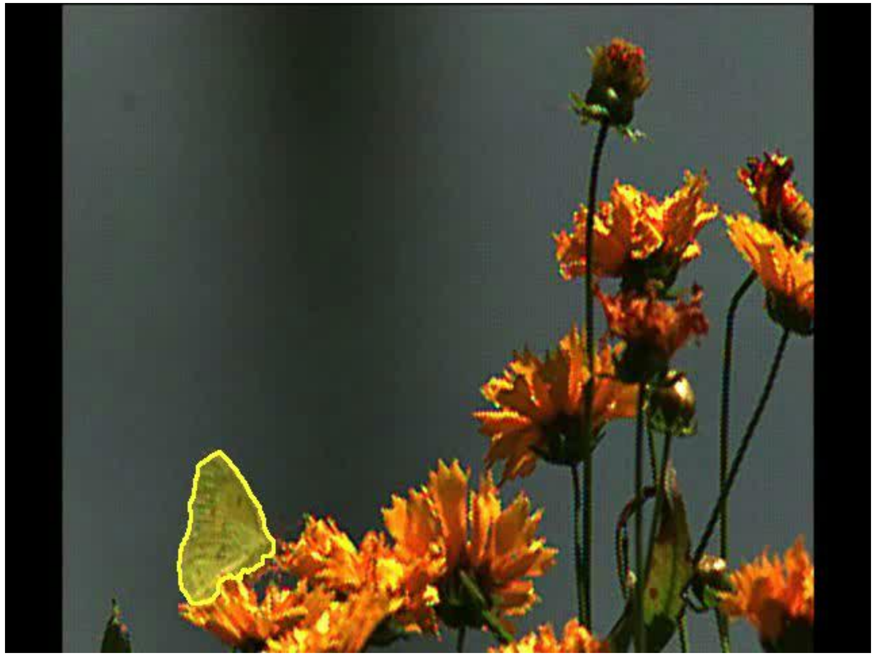


- Pre-trained for segmentation task only
  Backbone pre-trained on ImageNet
- YouTube-VOS [1]: 3471 videos
  with ground-truth segmentation masks
- 40 epochs with 1000 iterations
  batch size: 64 image pairs
- Pre-training: 20 hours on a single GPU (Nvidia 1080 GTX)
- Backbone is fixed during online tracking

[1] N. Xu et al. YouTube-VOS: a large-scale video object segmentation benchmark. arXiv:1809.03327, 2018

D3S: Tracking results (in 2020)

- State-of-the-art results on three tracking benchmarks
  VOT 2016 [1], VOT 2018 [2] and GOT-10k [3]
- Comparable to state-of-the-art trackers on TrackingNet [4]
- SOTA trackers:
  - Offline train for localization on large tracking datasets
- Generalization capability of a tracker
  Even though trained on segmentation task only



[1] Kristan et al. The Visual Object Tracking VOT2016 Challenge Results, ECCVW 2016
[2] Kristan et al. The sixth Visual Object Tracking VOT2018 challenge results, ECCVW 2018
[3] Huang et al. GOT-10k: A Large High-Diversity Benchmark for Generic Object Tracking in the Wild, TPAMI 2019
[4] Mueller et al. TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild, ECCV 2018

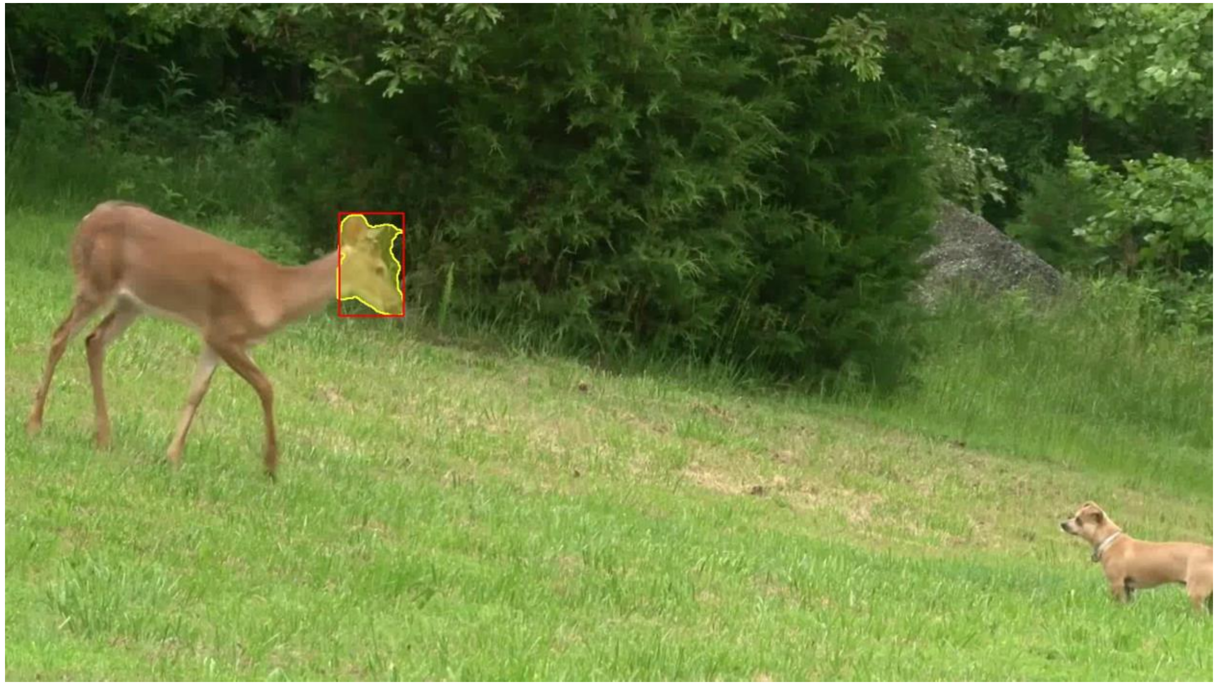
D3S: Video Segmentation

- Evaluated on two segmentation benchmarks: DAVIS2016 [1] and DAVIS2017 [2]
- Results comparable to the state-of-the-art segmentation methods
  - An order of magnitude faster (do not require heavy fine-tuning)
  - Not trained on DAVIS datasets
- Performance better than segmentation tracker (SiamMask), but still in real-time

	$\mathcal{J_M}^{16}$	$\mathcal{F_M}^{16}$	$\mathcal{J_M}^{17}$	$\mathcal{F_M}^{17}$	FPS
D3S	75.4	72.6	② 57.8	③ 63.8	② 25.0
SiamMask	71.7	67.8	54.3	58.5	① **55.0**
OnAVOS	① **86.1**	① **84.9**	① **61.6**	① **69.1**	0.1
FAVOS	② 82.4	79.5	54.6	61.8	0.8
VM	③ 81.0	-	③ 56.6	-	3.1
OSVOS	79.8	② 80.6	③ 56.6	② 63.9	0.1
PML	75.5	③ 79.3	-	-	3.6
OSMN	74.0	72.9	52.5	57.1	③ 8.0

[1] Perazzi et al. A benchmark dataset and evaluation methodology for video object segmentation. CVPR 2016
[2] Pont-Tuset et al. The 2017 davis challenge on video object segmentation. arXiv:1704.00675, 2017

D3S: Qualitative Examples



Tracking part of an object

Similar targets

Lukežič, Matas, Kristan, D3S -- A Discriminative Single Shot Segmentation Tracker, CVPR2020

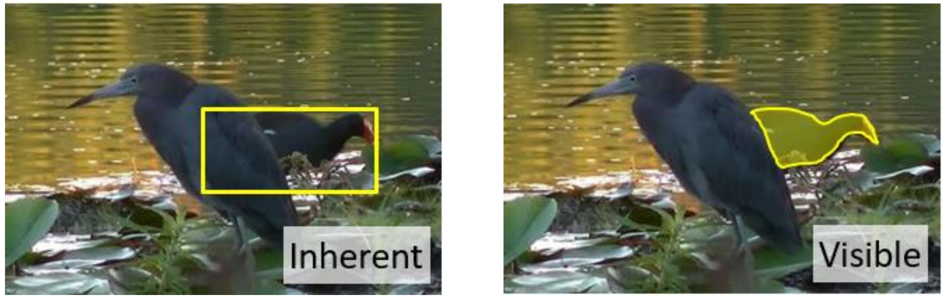
D3S: Qualitative Examples

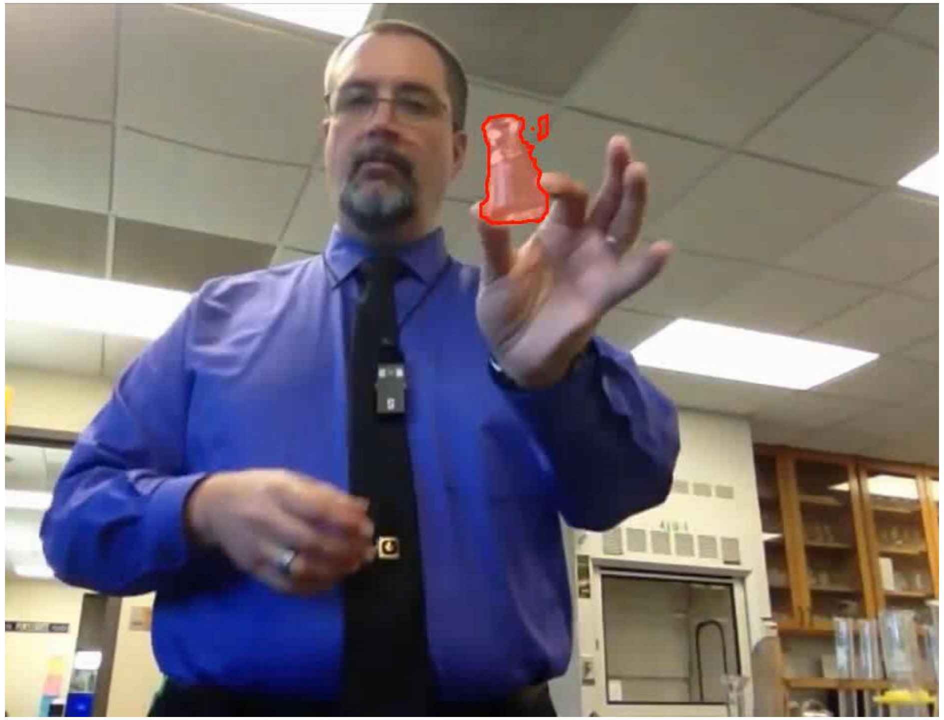
- Target deformation
- Scale and aspect change

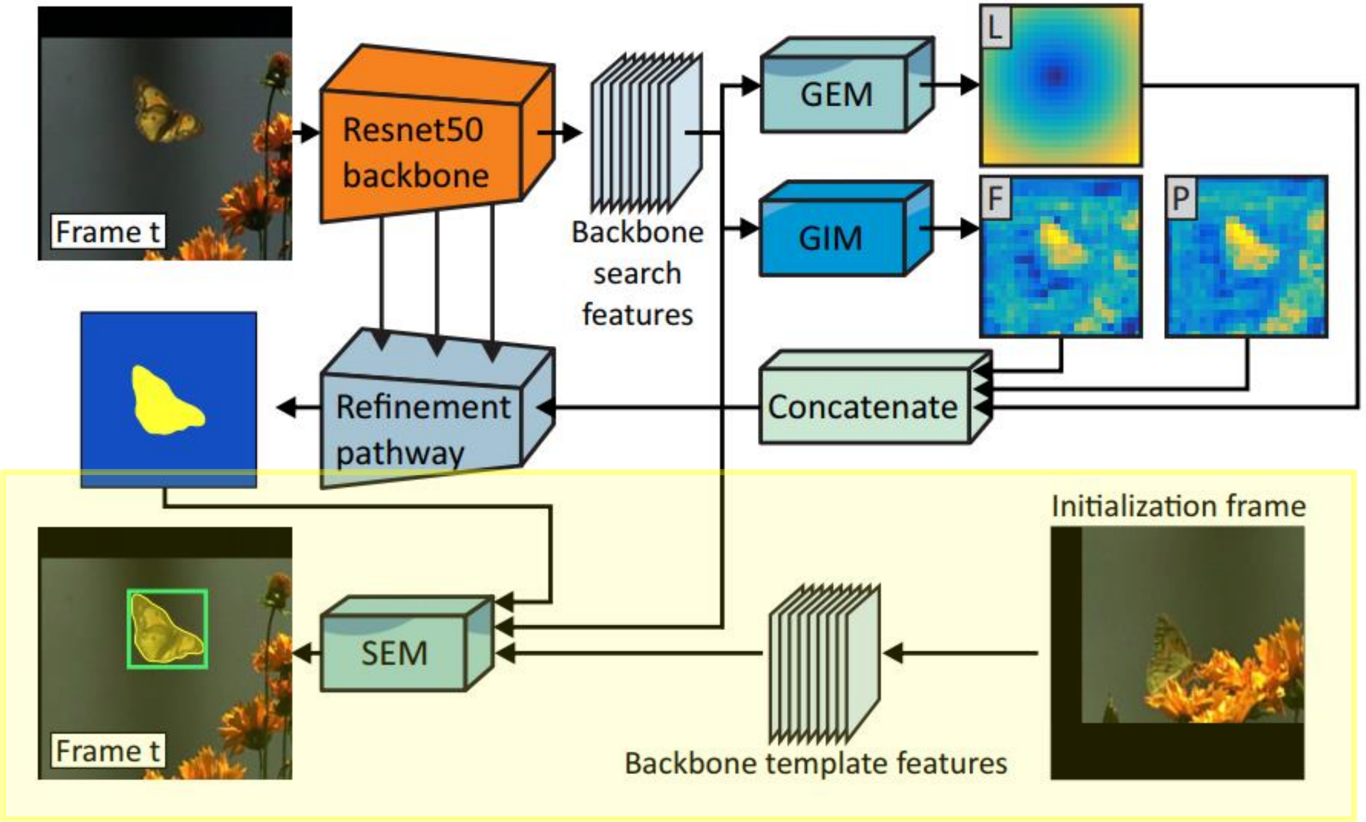
Lukežič, Matas, Kristan, D3S -- A Discriminative Single Shot Segmentation Tracker, CVPR2020

D3S$_2$ published recently

- More advanced architecture







Lukežič, Matas, Kristan, A Discriminative Single-Shot Segmentation Network for Visual Object Tracking, IEEE TPAMI, 2021

Transformers

- Transformers have emerged with the seminal paper in 2017[1]
- An example of the most trivial "attention operation" (Scaled dot-product attention)

3xwxh

key K whx3

query Q whx3

value V whx3

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$\tilde{V}$ whx3

3xwxh

QK^T

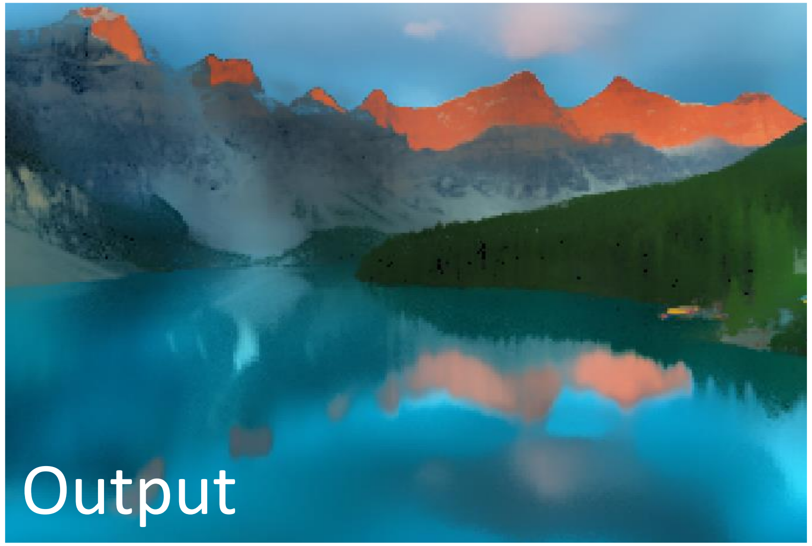
Similarity between each key and value

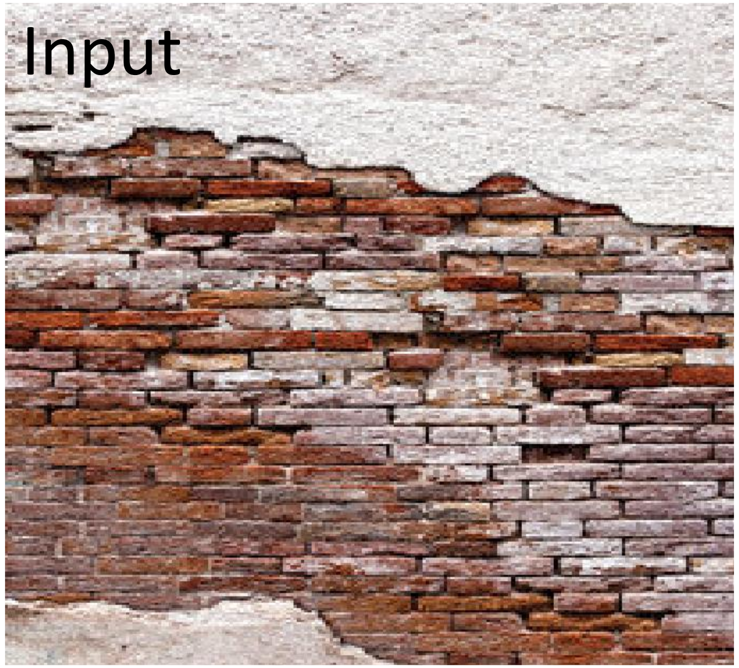
whxwh

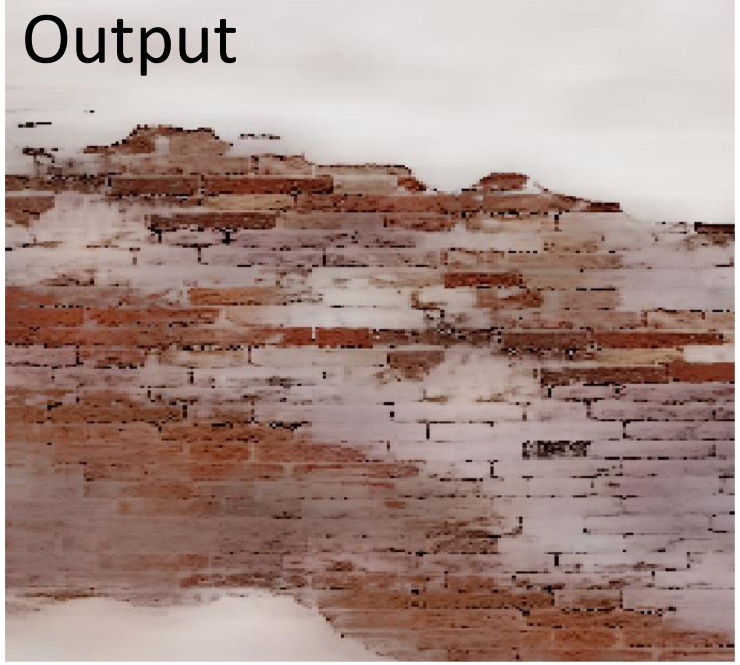
Attention essentially computes $\tilde{V}$ as a reconstruction of pixels in V.

[1] Vaswani et al., Attention is all you need, NIPS 2017

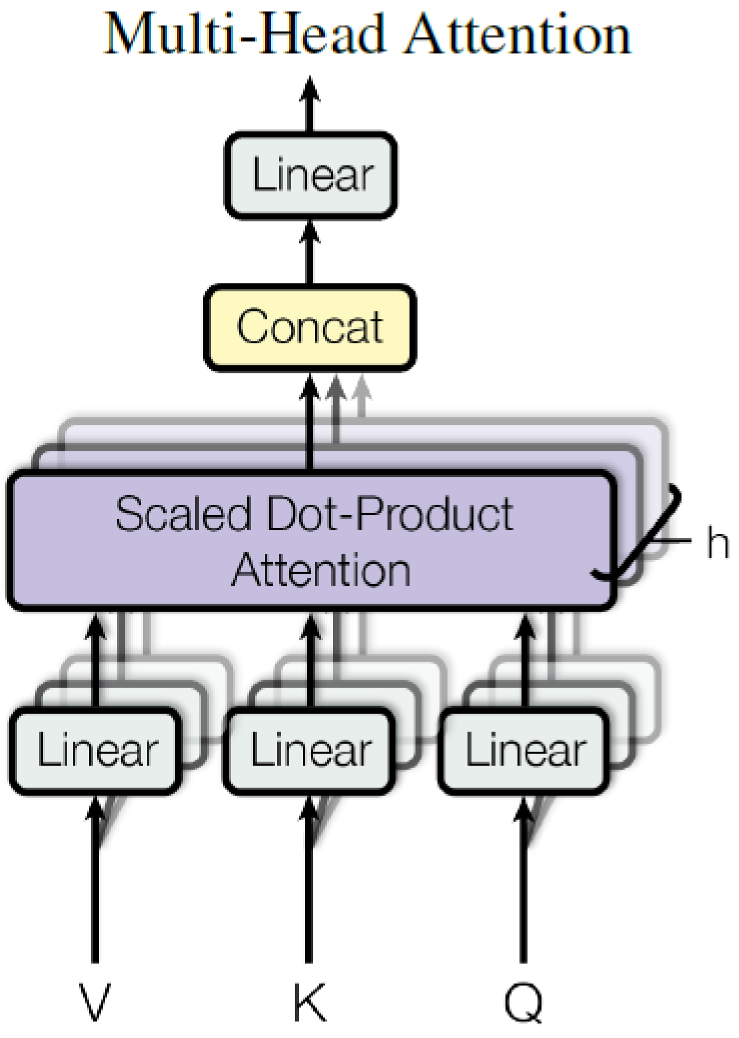
Local attention as image denoising



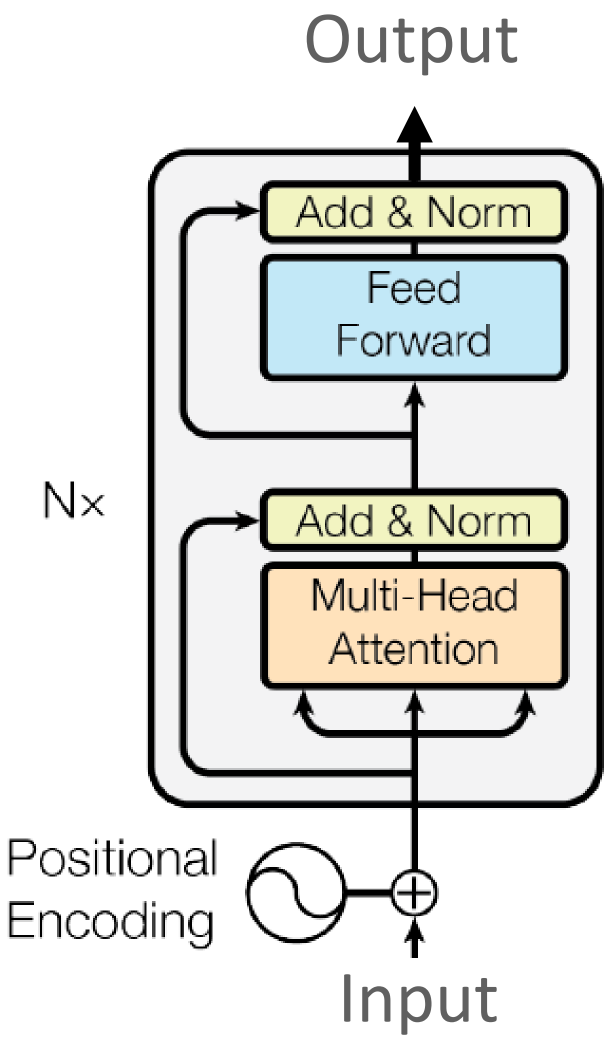




The main (en)coding block

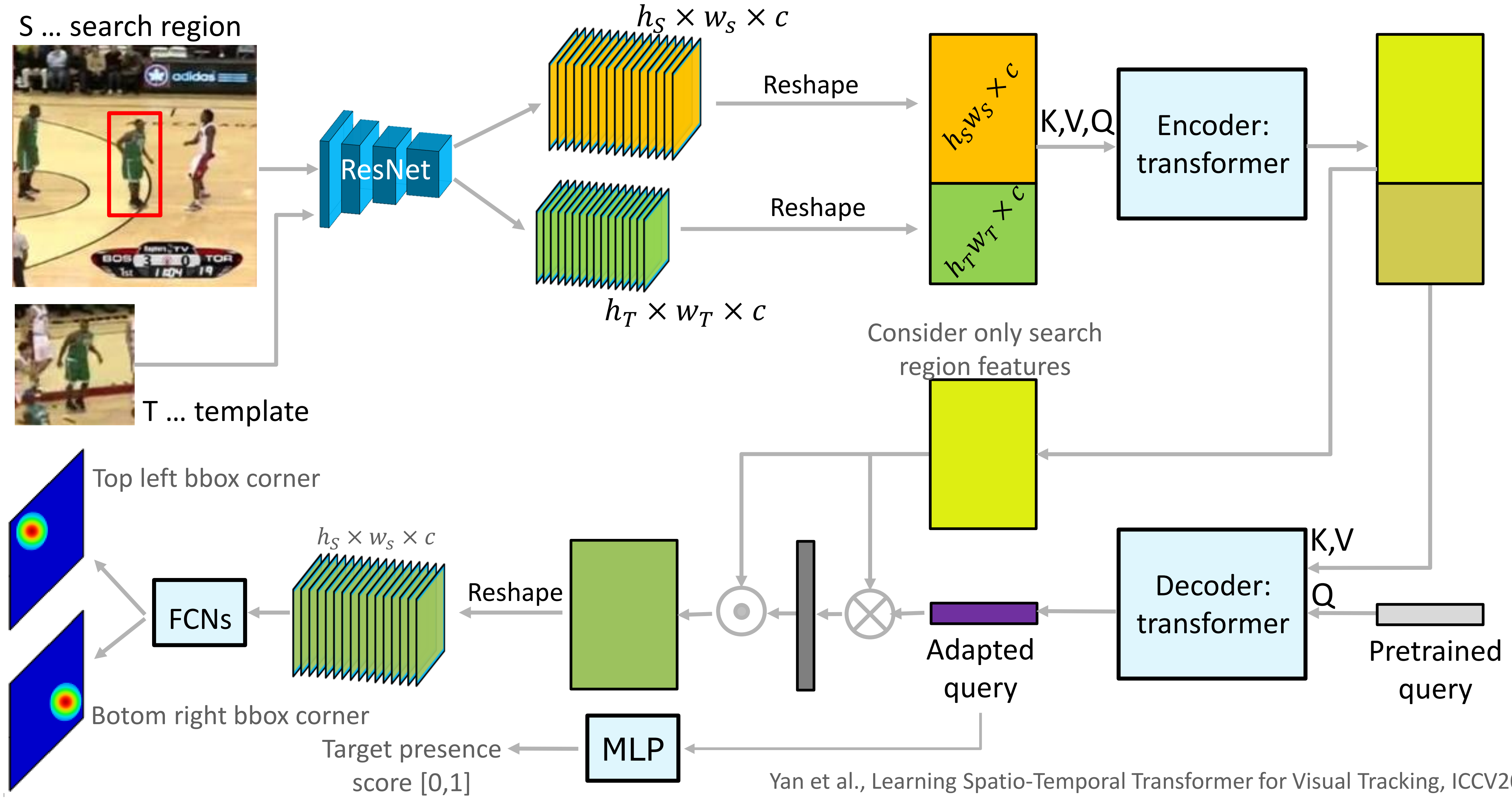


The transformer block:



Vaswani et al., Attention is all you need, NIPS 2017

Recent transformer tracker: STARK



Yan et al., Learning Spatio-Temporal Transformer for Visual Tracking, ICCV2021

STARK in action

- Large search range (partial occlusion handled well) & accurate bbox

… Ground truth mask … Tracker bounding box

STARK in action

- The forward pass filter construction in decoder not robust to distractors

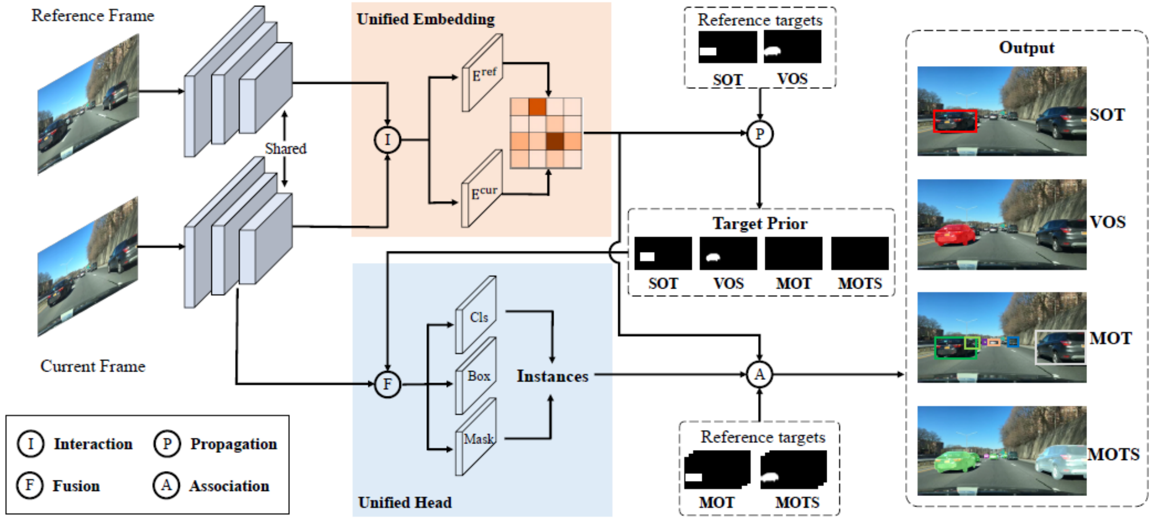
STARK

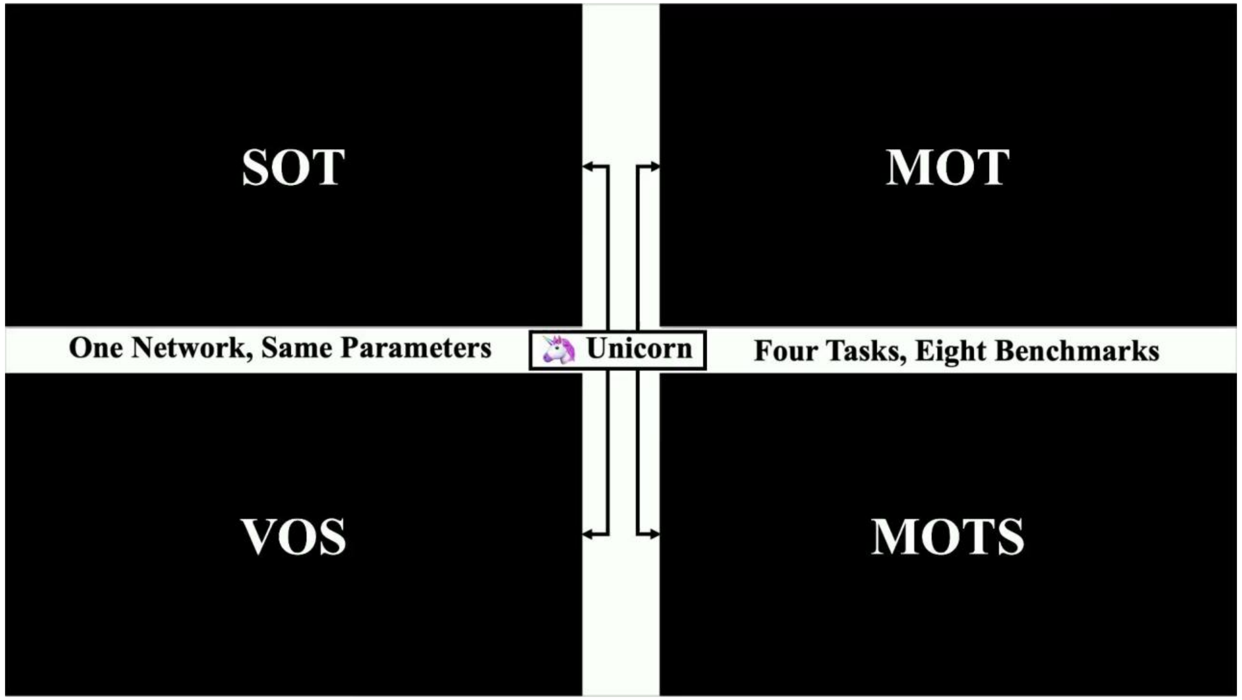
$D3S_2$

... Ground truth mask ... Tracker bounding box

[$D3S_2$] Lukežič, Matas, Kristan, A Discriminative Single-Shot Segmentation Network for Visual Object Tracking, TPAMI 2021

Recent works aim at multiple "tasks"





- Codename: "Unicorn"
- Attempting to unify the tracking tasks
- A single backbone handling different input/output specifications
- Allows learning a common network from MANY datasets with many tasks

Yan et al., [Towards Grand Unification of Object Tracking](), ECCV 2022

Deep learning for tracking – summary

- Various architectures for localization overviewed
  - CNN patch classifier (MDNet)
  - CNN backbone trained for localization by correlation (SiamFc)
  - CNN pre-trained features + a deep DCF (ATOM)
- Bounding box estimation
  - Regression (MDNet)
  - Region proposals, i.e., regression to several hypotheses (SiamRPN)
  - CNN for overlap optimization (a modified IoUNet in ATOM)
- Beyond bounding boxes (D3S, SiamMask)
  - Closing the gap between tracking and video segmentation
- Transformers (STARK, TransT, MixFormer), more recent works (Unicorn)...

Consider this a glimpse
(much more approaches exist)

References

MDNet branch:

- Nam and Han, Learning Multi-Domain Convolutional Neural Networks for Visual Tracking, CVPR2016
- Jung, Son, Baek, Han, Real-Time MDNet, ECCV2018

Siamese networks:

- SiamFc:
    - Bertinetto et al., Fully-Convolutional Siamese Networks for Object Tracking, ECCV VOT2016
    - Zhang et al., Learning the Model Update for Siamese Trackers, ICCV2019
- SiamRPN:
    - Li et al., SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks, CVPR2019
    - Li et al., High Performance Visual Tracking with Siamese Region Proposal Network, CVPR2018
    - Wang et al. Fast Online Object Tracking and Segmentation: A Unifying Approach. CVPR 2019

Deep DCF:

- Danelljan et al., ATOM: Accurate Tracking by Overlap Maximization, CVPR2019
- Bhat et al., Learning Discriminative Model Prediction for Tracking, ICCV201

Single-shot segmentation networks:

- Lukežič, et al, A Discriminative Single-Shot Segmentation Network for Visual Object Tracking, IEEE TPAMI 2021

Transformers:

- Yan et al., Learning Spatio-Temporal Transformer for Visual Tracking, ICCV2021
- Chen et al., Transformer Tracking, CVPR 2021