



Visual Tracking

问题描述：

对于一个视频序列，在第一帧的时候给出物体所在位置
要求在后面的帧中能识别物体新的位置



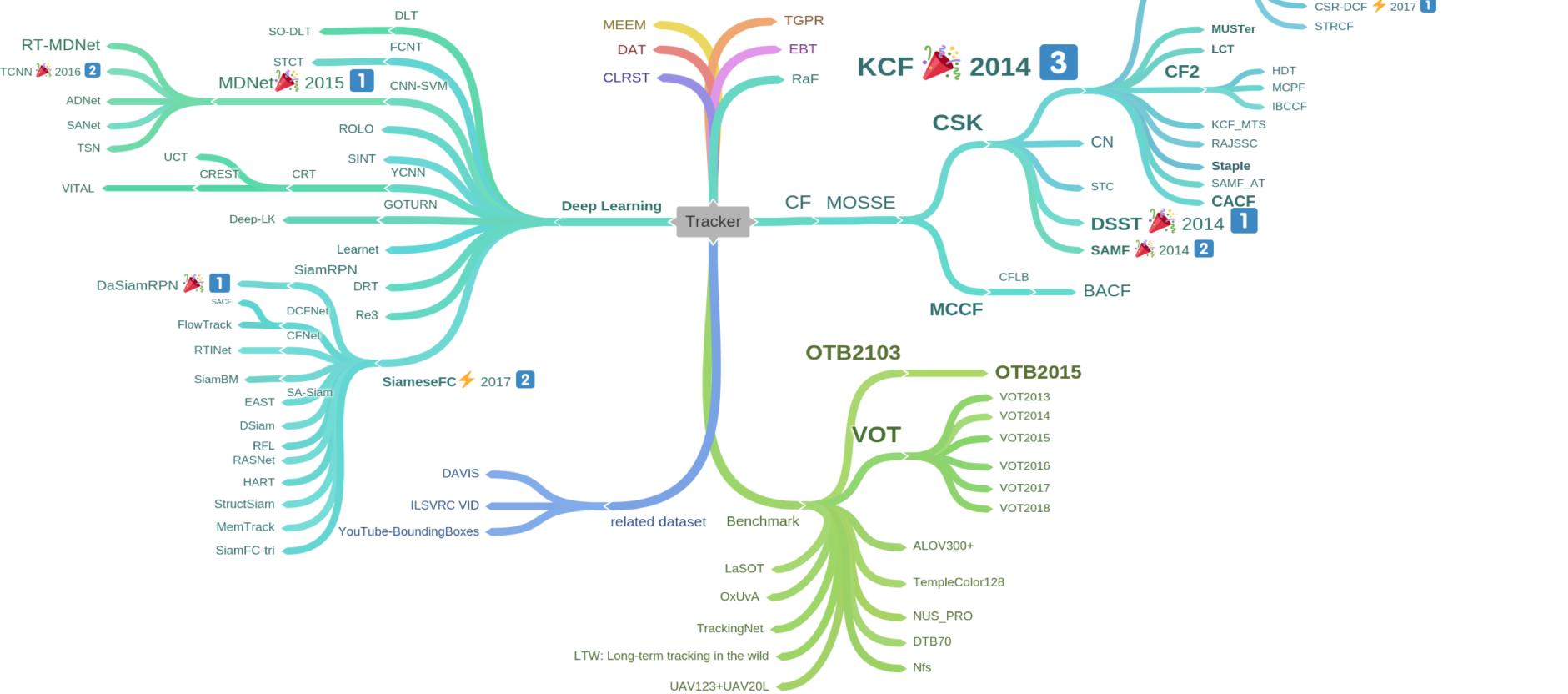
难点：

- 运动模糊
- 目标被遮挡
- 目标形变
- 目标尺寸变化
- 光照变化
- 缺乏训练样本

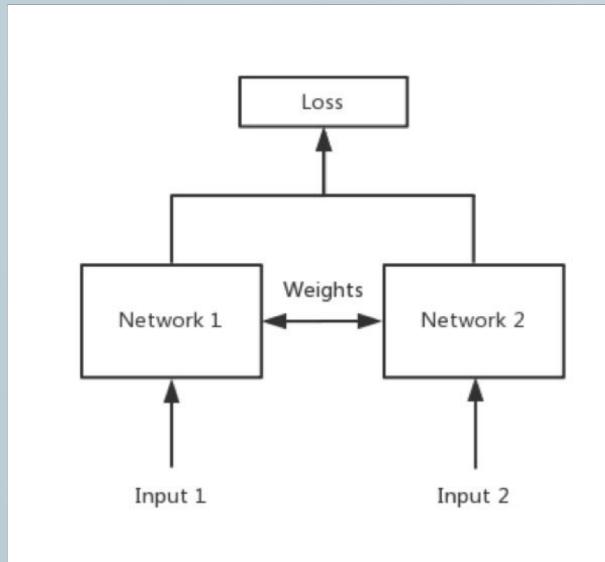
思路：

利用前后两帧的相似性

 denotes VOT *baseline* experiment
 denotes VOT *realtime* experiment



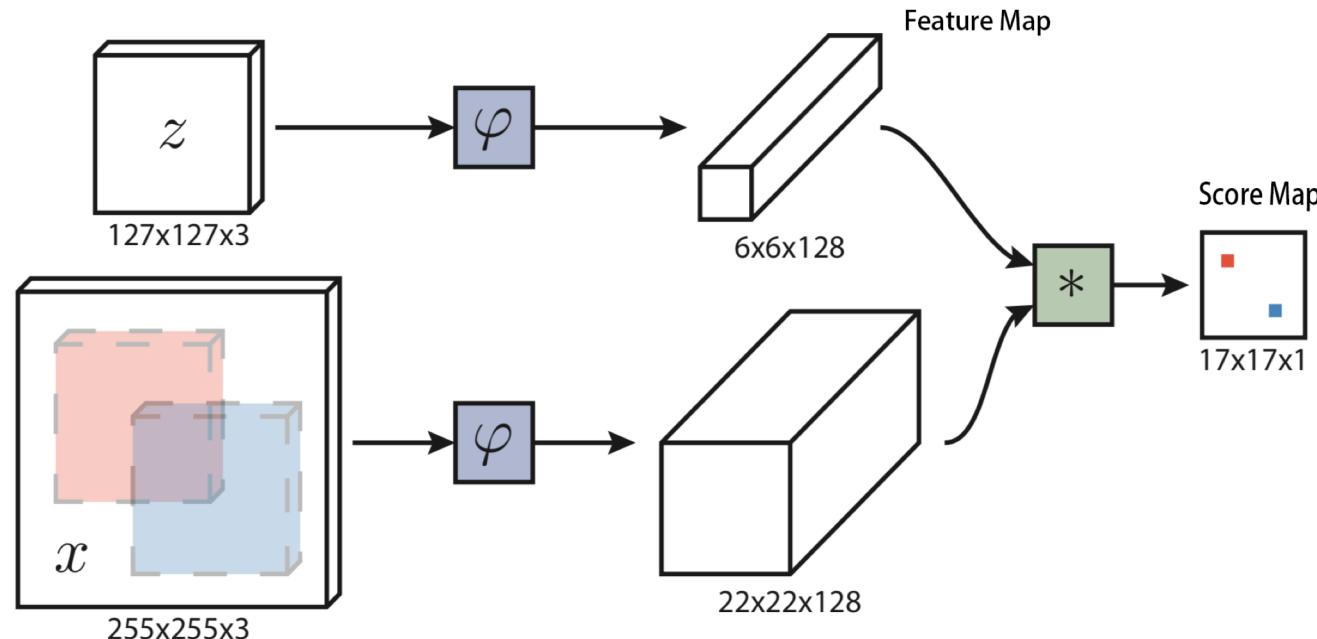
Siamese Network

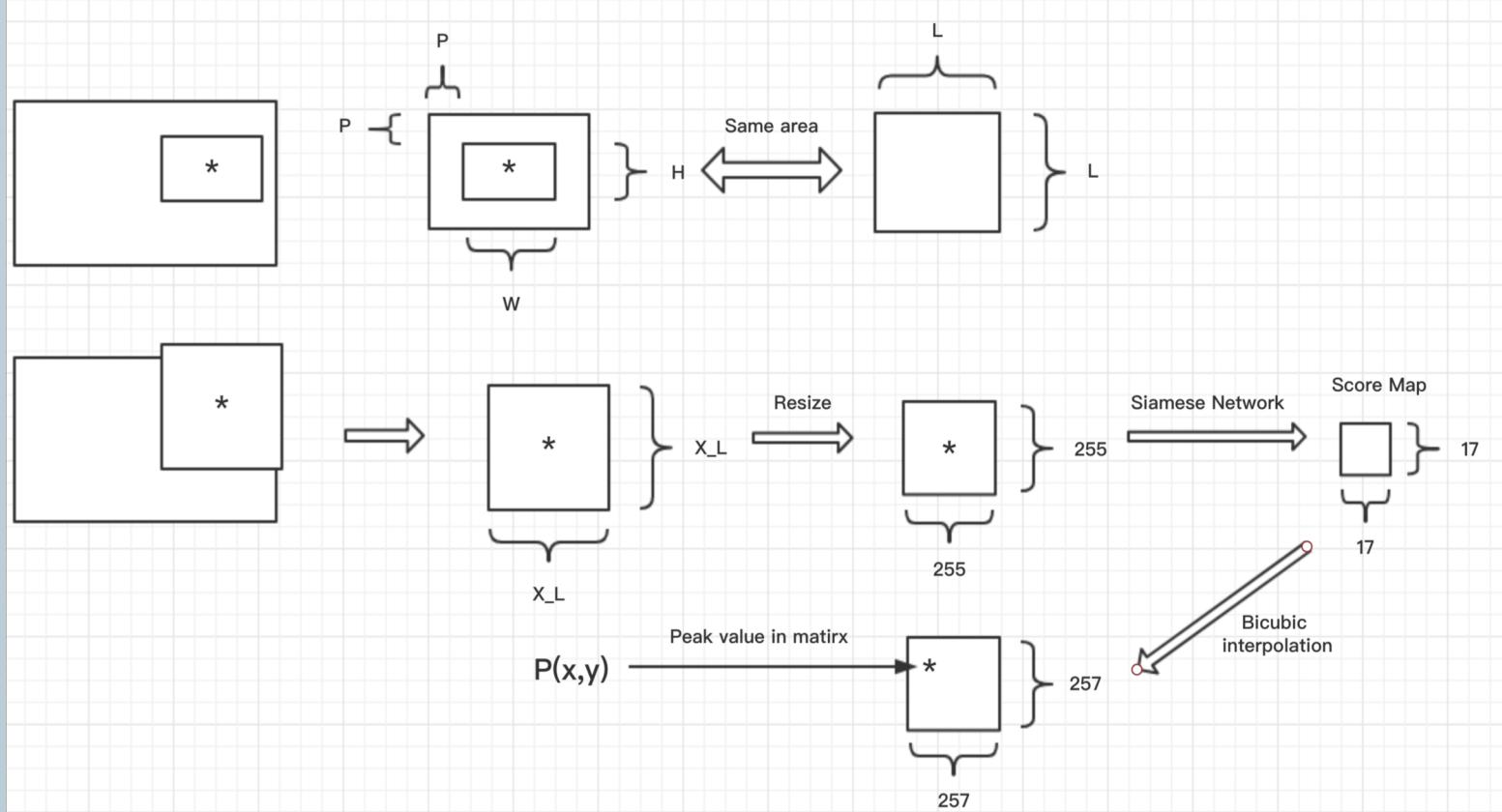


Siamese Network 称为孪生网络，左右两个神经网络共享相同的权值。简单来说就是衡量两个输入的相似程度。当X1和X2属于同一个类别时最小化损失函数。

输入一对数据，输出数据是否属于同一类别。特点是接收两个输入而不是一个输入。

SiameseFC

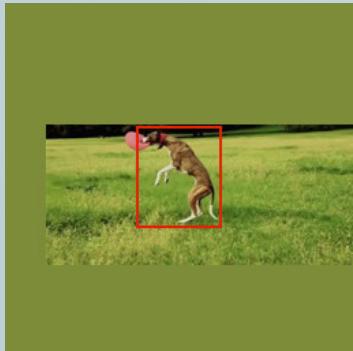




$$P = (W+H)/4$$

$$L^*L = (W+P)^*(H+P)$$

若有超出原图的部分我们用整张图的平均值进行填充



最终将得到一个 17×17 的Score Map，我们通过双三次插值算法将Score Map放大为 257×257 。此时，数值最大的点即为目标新的位置。

我们假定三个图像缩放比例：

{ 0.924556, 1.0, 1.0816 } 分别对应图像缩小、不变、放大。

我们对X进行三次操作，假设计算出要截的正方形边长为L，我们令L分别乘以这三个比例，其他操作不变。

最终可以得到三个Score Map，我们取三个里面最大值最大的那个，认为目标缩放了相应比例

尺寸自适应：采用多尺度检测方式实现尺度自适应，便于在GPU上加速

在线不更新：神经网络训练后完全固定，tracking时不反向传播，鲁棒性较强

检测区域较大：检测区域约为目标区域的4倍

网络结构：

| Layer | Support | Chan. map | Stride | Activation size | | |
|-------|---------|-----------|--------|-----------------|------------|--------|
| | | | | for exemplar | for search | chans. |
| | | | | 127 × 127 | 255 × 255 | ×3 |
| conv1 | 11 × 11 | 96 × 3 | 2 | 59 × 59 | 123 × 123 | ×96 |
| pool1 | 3 × 3 | | 2 | 29 × 29 | 61 × 61 | ×96 |
| conv2 | 5 × 5 | 256 × 48 | 1 | 25 × 25 | 57 × 57 | ×256 |
| pool2 | 3 × 3 | | 2 | 12 × 12 | 28 × 28 | ×256 |
| conv3 | 3 × 3 | 384 × 256 | 1 | 10 × 10 | 26 × 26 | ×192 |
| conv4 | 3 × 3 | 384 × 192 | 1 | 8 × 8 | 24 × 24 | ×192 |
| conv5 | 3 × 3 | 256 × 192 | 1 | 6 × 6 | 22 × 22 | ×128 |

单点损失函数：

$$\ell(y, v) = \log(1 + \exp(-yv))$$

v表示Score Map中点的真实值，y表示点的标签，取值为 {1, -1} 对应正负样本

总损失函数：

$$L(y, v) = \frac{1}{|\mathcal{D}|} \sum_{u \in \mathcal{D}} \ell(y[u], v[u])$$

y的取值：

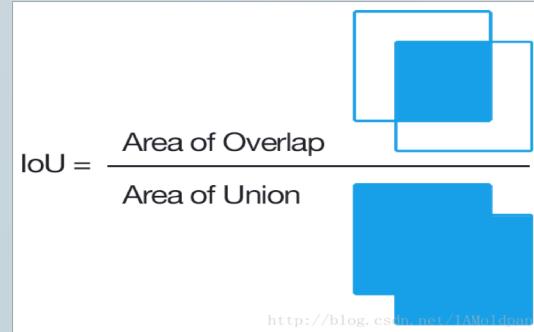
$$y[u] = \begin{cases} +1 & \text{if } k\|u - c\| \leq R \\ -1 & \text{otherwise} \end{cases} .$$

即当前点u到目标点c的距离满足上面的要求就认为是正样本，反之是负样本。R 和 k 是常数

$$\arg \min_{\theta} \mathbb{E}_{(z, x, y)} L(y, f(z, x; \theta))$$

评测指标：

IoU (重叠度)



Precision

若预测点与目标点距离小于阈值threshold(这里取20)则认为预测成功，
Precision = 预测成功次数/总次数

Precision AUC

ROC曲线下方面积

TPR , 表示预测为正的正确结果TP在所有正样本T中的占比

FPR , 表示预测为正的错误结果FP在所有负样本F中的占比

ROC曲线是由FPR为X轴坐标 , TPR为Y轴坐标的曲线 ,

我们通过改变阈值的值来获得多组FPR和TPR的值

| Tracker | accuracy | # failures | overlap | speed (fps) |
|-------------------------|----------|------------|---------|-------------|
| MDNet [9] | 0.5620 | 46 | 0.3575 | 1 |
| EBT [41] | 0.4481 | 49 | 0.3042 | 5 |
| DeepSRDCF [6] | 0.5350 | 60 | 0.3033 | < 1 * |
| SiamFC-3s (ours) | 0.5335 | 84 | 0.2889 | 86 |
| SiamFC (ours) | 0.5240 | 87 | 0.2743 | 58 |
| SRDCF [42] | 0.5260 | 71 | 0.2743 | 5 |
| sPST [43] | 0.5230 | 85 | 0.2668 | 2 |
| LDP [12] | 0.4688 | 78 | 0.2625 | 4 * |
| SC-EBT [44] | 0.5171 | 103 | 0.2412 | — |
| NSAMF [45] | 0.5027 | 87 | 0.2376 | 5 * |
| StruckMK [3] | 0.4442 | 90 | 0.2341 | 2 |
| S3Tracker [46] | 0.5031 | 100 | 0.2292 | 14 * |
| RAJSSC [12] | 0.5301 | 105 | 0.2262 | 2 * |
| SumShift [46] | 0.4888 | 97 | 0.2233 | 17 * |
| DAT [47] | 0.4705 | 113 | 0.2195 | 15 |
| SO-DLT [7] | 0.5233 | 108 | 0.2190 | 5 |

C++ 版本 :

```
0 -- Precision: 46.62 -- Precisions AUC: 16.82 -- IOU: 37.03 -- Speed: 2.49
0 -- Precision: 24.24 -- Precisions AUC: 8.78 -- IOU: 18.35 -- Speed: 2.77
0 -- Precision: 44.00 -- Precisions AUC: 14.79 -- IOU: 35.38 -- Speed: 3.12
1 -- Precision: 99.66 -- Precisions AUC: 36.54 -- IOU: 62.28 -- Speed: 2.07
1 -- Precision: 98.99 -- Precisions AUC: 36.93 -- IOU: 69.20 -- Speed: 2.50
1 -- Precision: 100.00 -- Precisions AUC: 39.69 -- IOU: 74.39 -- Speed: 3.06
2 -- Precision: 100.00 -- Precisions AUC: 39.96 -- IOU: 72.84 -- Speed: 1.42
2 -- Precision: 100.00 -- Precisions AUC: 37.83 -- IOU: 67.77 -- Speed: 1.84
2 -- Precision: 100.00 -- Precisions AUC: 42.96 -- IOU: 68.97 -- Speed: 2.49
Precision (20.00 px): 90.97 -- Precisions AUC: 35.25 -- IOU: 63.58 -- Speed: 2.06
```

python 版本 :

```
0 -- Precision: 46.62 -- Precisions AUC: 16.82 -- IOU: 37.03 -- Speed: 0.46
1 -- Precision: 99.66 -- Precisions AUC: 36.54 -- IOU: 62.28 -- Speed: 0.47
2 -- Precision: 100.00 -- Precisions AUC: 39.96 -- IOU: 72.84 -- Speed: 0.47
Precision (20 px): 92.18 -- Precisions AUC: 35.63 -- IOU: 64.60 -- Speed: 0.47
```



Thank you