

Школа Анализа  
Данных Яндекса

Курс «Анализ изображений и видео, ч.2»

Лекция №6  
«Сопровождение объектов в видео»

Антон Конушин

Заведующий лабораторией компьютерной графики и мультимедиа  
ВМК МГУ

24 марта 2017 года



# Сопровождение объектов



- Есть видео (-поток или –последовательность) в которой запечатлены движущиеся объекты
- Мы хотим отследить движения объектов по всем кадрам и определить траектории их движения
  - «След объекта» - «Object track»
  - $X(1), X(2), X(3), \dots, X(N)$  – последовательность координат объектов в кадрах
- «Object tracking» (сопровождение объектов, отслеживание объектов, слежение за объектами)



## Два основных варианта задачи

---



- Visual object tracking (VOT)
  - Визуальное сопровождение объектов
- Multiple object tracking (MOT)
  - Множественное сопровождение объектов



# Visual object tracking



- Рассматривается отслеживание одного объекта
- Объект уже выделен на первом кадре
- «Model-free» - нет ничего, кроме одного изображения объекта на первом кадре, т.е. не можем детектировать объект
- «Short-term» - отслеживаем на коротких промежутках времени, не применяем повторное обнаружение
- Не используются будущие кадры, только предыдущие



# Multiple object tracking



- Задача «выделения и сопровождения множества объектов»
  - Нужно найти все объекты на всех кадрах
  - Определить сколько у нас разных «экземпляров» объектов
  - Найти на каких кадрах виден каждый экземпляр и где он именно
- Обобщение задачи «выделение объектов на изображении» на случай видео
- В отличие от VOT:
  - Работаем со множеством объектов
  - На длительных промежутках времени
  - Есть модель объектов (возможность повторного обнаружения)
  - Разрешено «заглядывать в будущее»



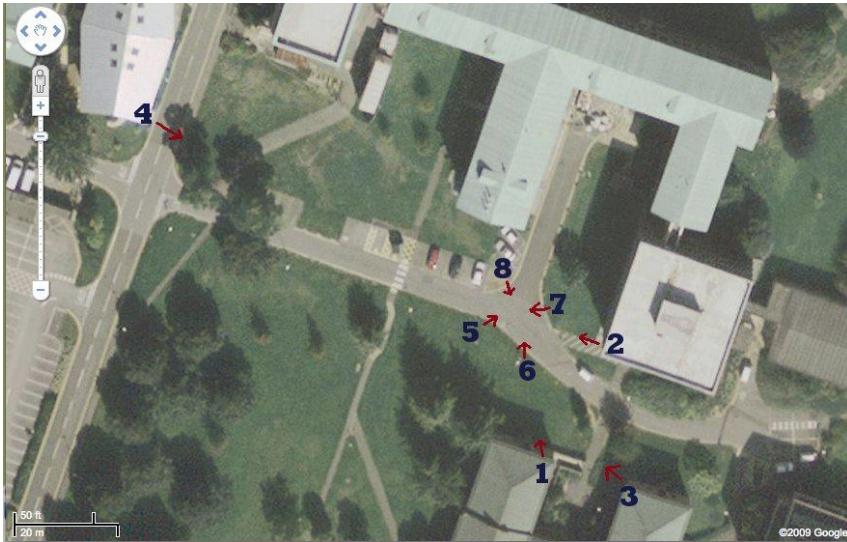
# Сложность задачи

- Вычислительная нагрузка
  - Нужно обрабатывать  $N$  кадров в секунду
- Изменение по времени
  - Вид объекта меняется от кадра к кадру из-за ракурса, изменения освещения, внутренних изменений (скейтбордист)
- Взаимодействие объектов
  - Перекрытия объектов
  - Визуальное сходство объектов
  - И т.д.





# Сложность оценки результата



- Для оценки качества работы алгоритмов слежения и настройки параметров требуются размеченные эталонные данные
- Подготовить эталонные данные для видео существенно сложнее, чем для изображений
  - Один эталонный пример для выделения объектов – 1 изображение
  - Один эталонный пример для отслеживания объектов - 1 видео
- Сейчас есть хорошие конкурсы, но объём данных по прежнему ограничен, особенно для МОТ



## Visual object tracking (VOT)



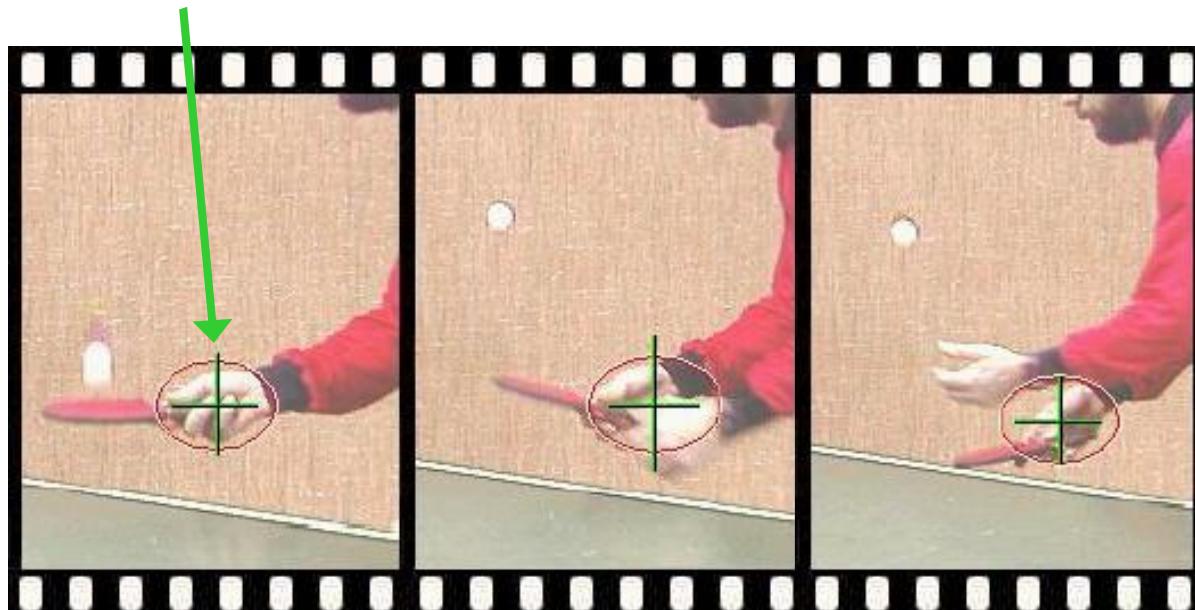
# Визуальное сопровождение



- Хотим сопровождать произвольный объект
- Поэтому нет возможности заранее обучить детектор объектов
- Придётся строить какое-то визуальное описание (дескриптор, упрощённую модель) объекта и искать на следующем кадре область изображения, похожую на искомый объект



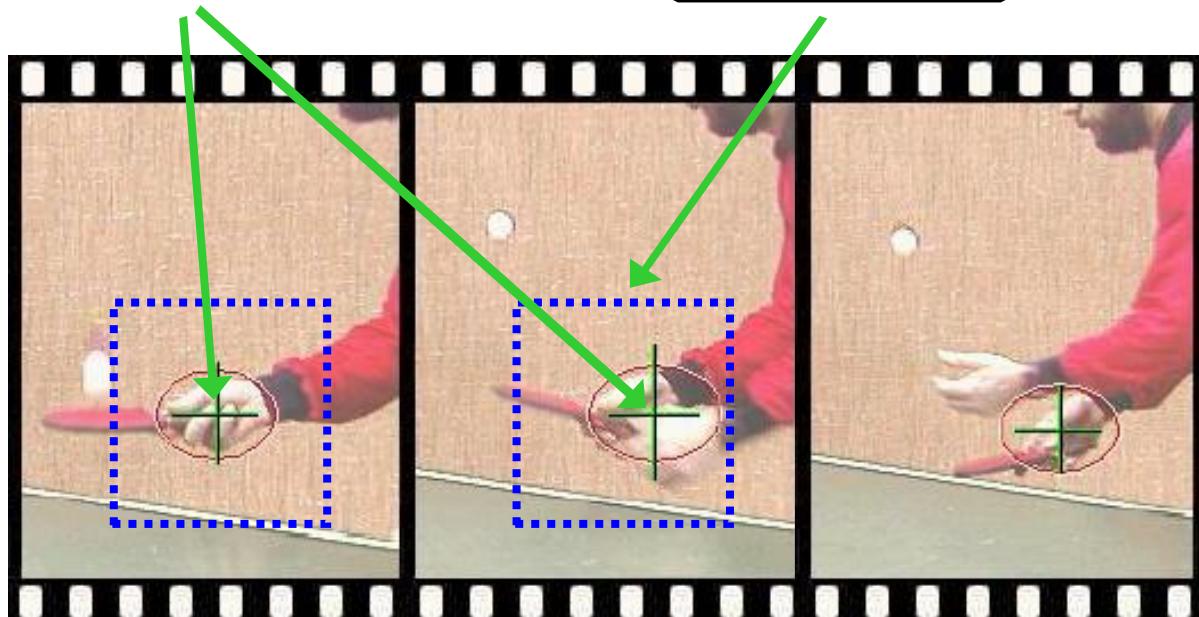
# Схема слежения



... Текущий кадр



# Схема слежения



Модель

Кандидат

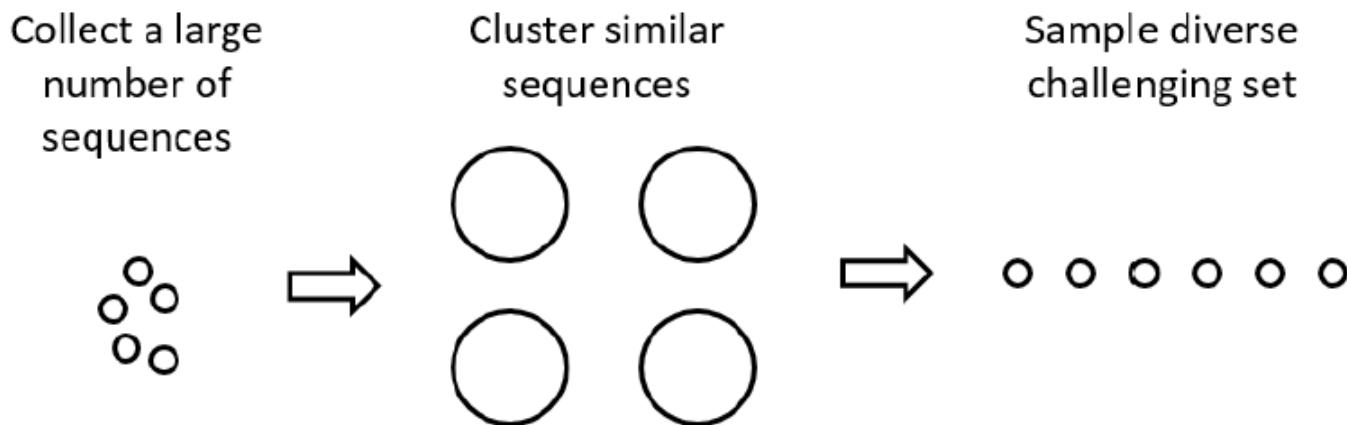
...      Текущий  
кадр      ...



# VOTChallenge 2013-2016 и далее



- Главный текущий конкурс – VOT Challenge  
<http://votchallenge.net/>
- Принципы:
  - Открытые реализации всех методов
  - Matlab-тулкит для экспериментальной оценки методов
  - Сравнение по точности и скорости
  - Небольшой, но разнообразный набор данных
  - Короткие последовательности (100 кадров)



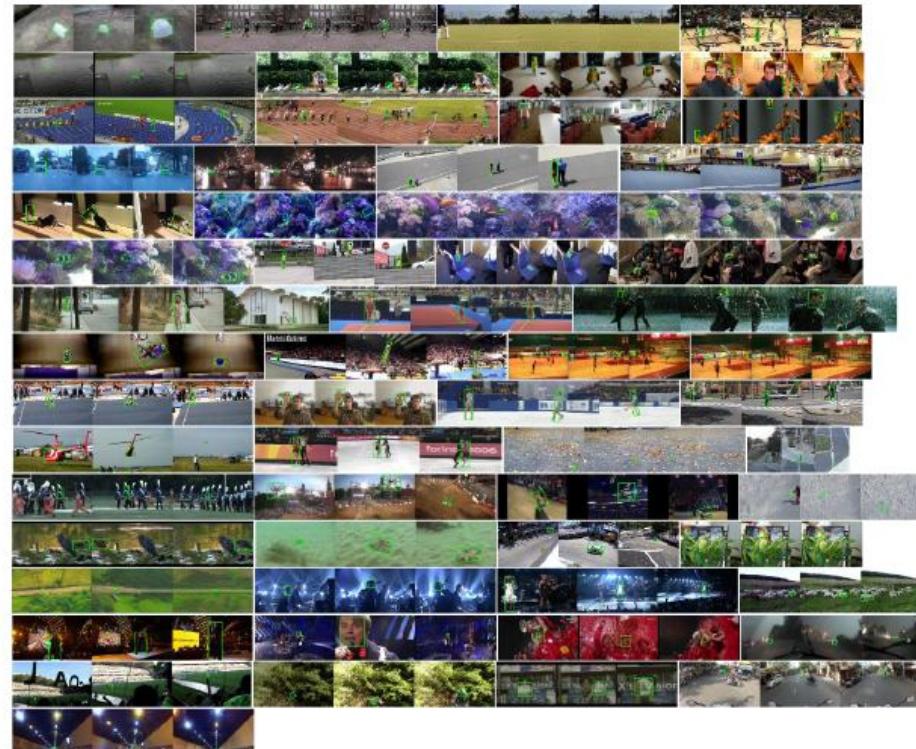


# VOT 2016

---



- The performance on VOT2015 dataset did not saturate in 2015 challenge
- Kept all 60 sequences from VOT2015 challenge
- NEW:  
*Objects re-annotated!*



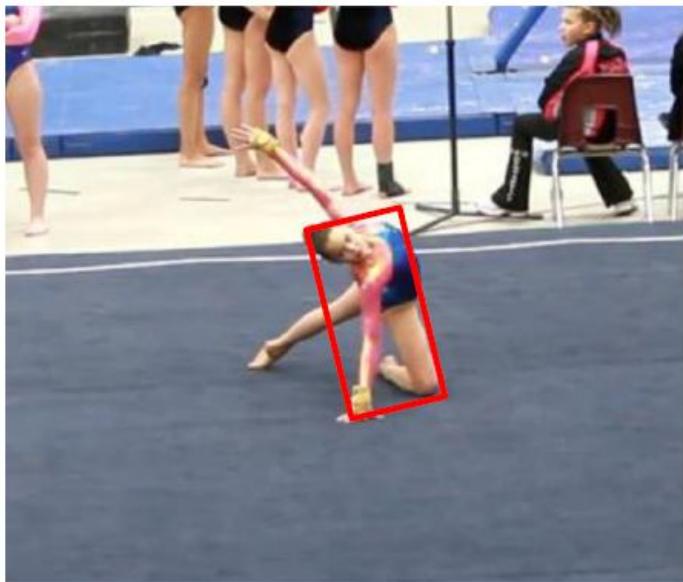


# Разметка объектов



## Automatic bounding box placement

1. Segment the target (semi-automatic)
2. Automatically fit a bounding box by optimizing a cost function



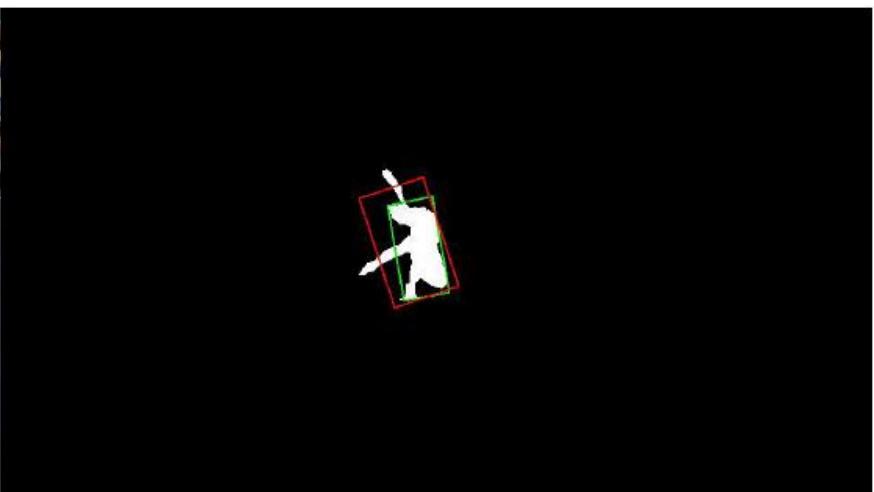
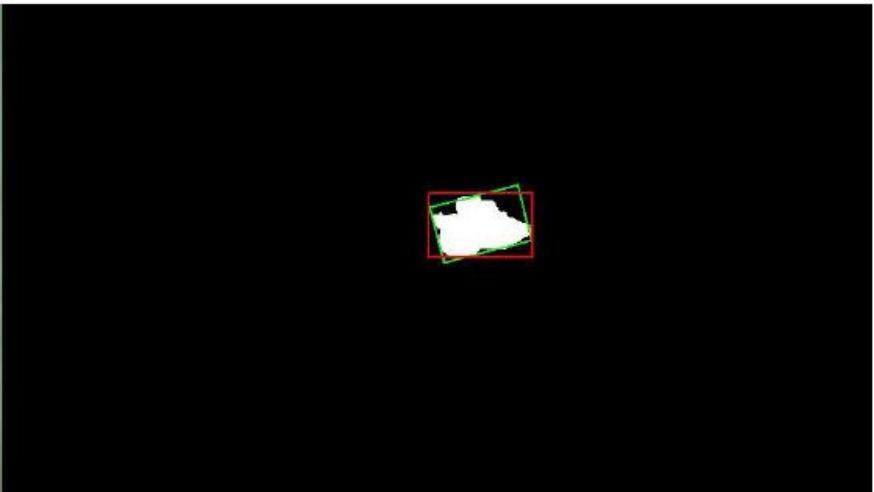
- Visual verification of the results
  - 12% reverted to the VOT2015 annotation



# Разметка объектов



- Average overlap between **VOT2015** and **VOT2016** BB: 0.74

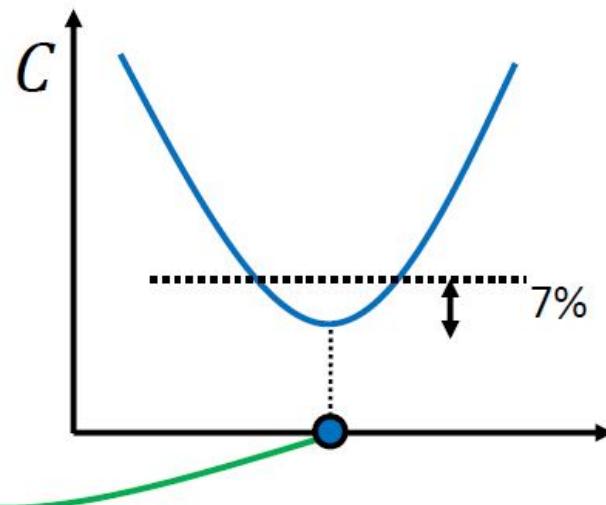
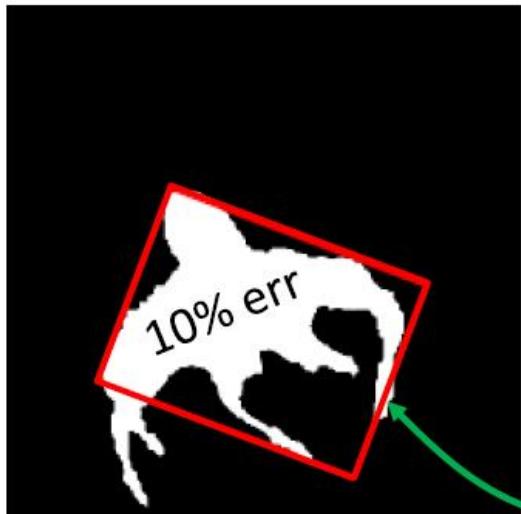




# Неточность разметки



- Segmentation uncertainty results in bounding box uncertainty



- Uncertainty: Average of overlaps between optimal bounding box and those within 7%  $C$  increase.



# Покадровая разметка



- Manually and automatically labeled each frame with VOT2013 visual attributes (same as VOT2015):
  - i. Occlusion (M)
  - ii. Illumination change (M)
  - iii. Object motion (A)
  - iv. Object size change (A)
  - v. Camera motion (M)
  - vi. Unassigned (A)

M ... manual annotation, A ... automatic annotation



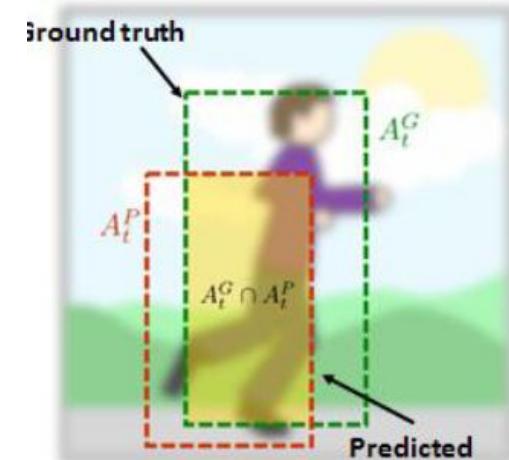
(i)	0	1	1	0
(ii)	0	0	0	0
(iii)	0	0	0	0
(iv)	1	1	1	0
(v)	0	0	0	0
(vi)	0	0	0	1



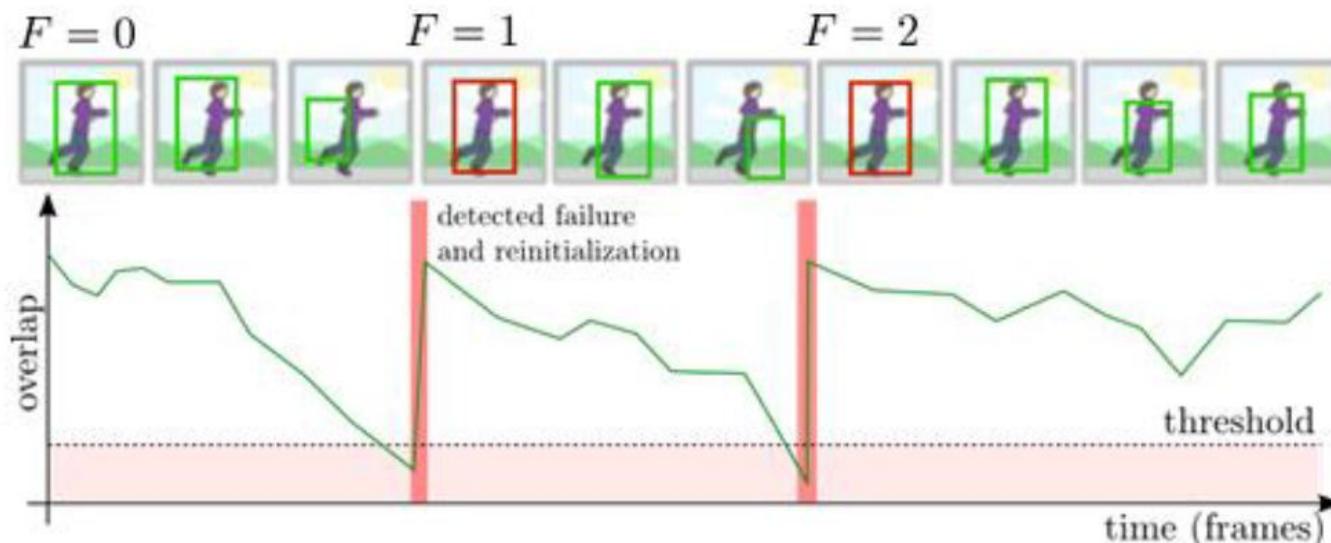
# Метрики качества

- Robustness:

*Number of times a tracker drifts off the target.*



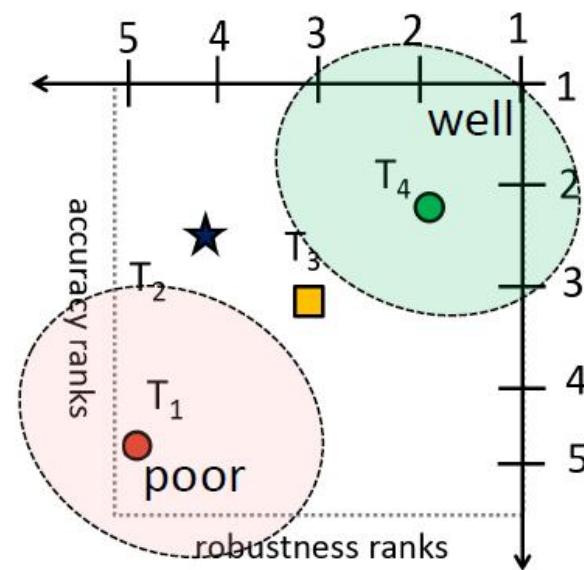
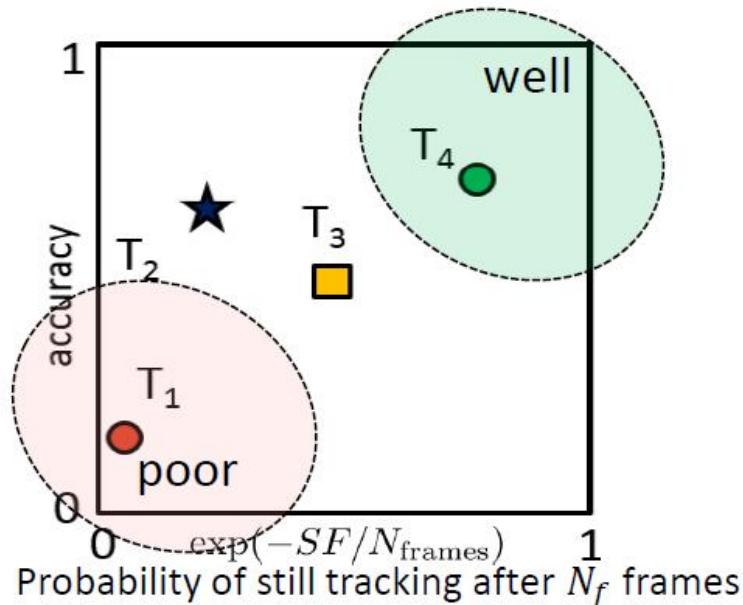
- Accuracy: *Average overlap during successful tracking.*





# Метрики качества

- Ranking methodology w.r.t. Accuracy and Robustness
- Assign equal rank to “equally” performing trackers:
  - Statistical significance of results and practical difference

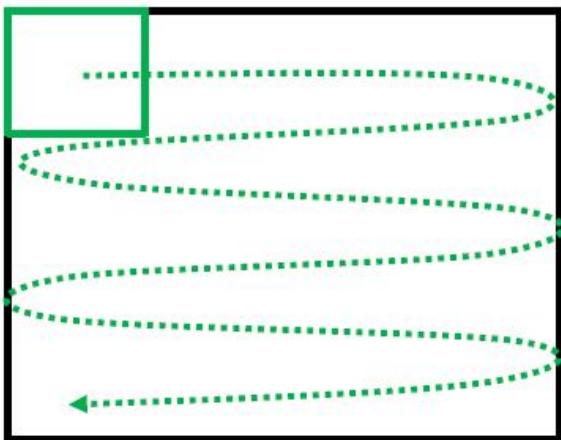


- A principled way to merge Accuracy and Robustness:
  - Expected average overlap (EAO)



# Оценка скорости работы

- Reduce the hardware bias in reporting tracking speed.
- Approach: The VOT2014 speed benchmark



600x600 image

Max operation in 30x30 window

Apply this filter to all pixels

Measure the time for filtering

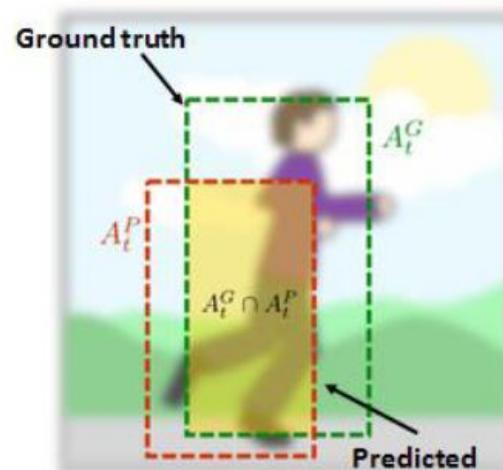
- Divide tracking time with time required to perform the filtering operation
- Equivalent Filter Operations (EFO)



# Проведение экспериментов



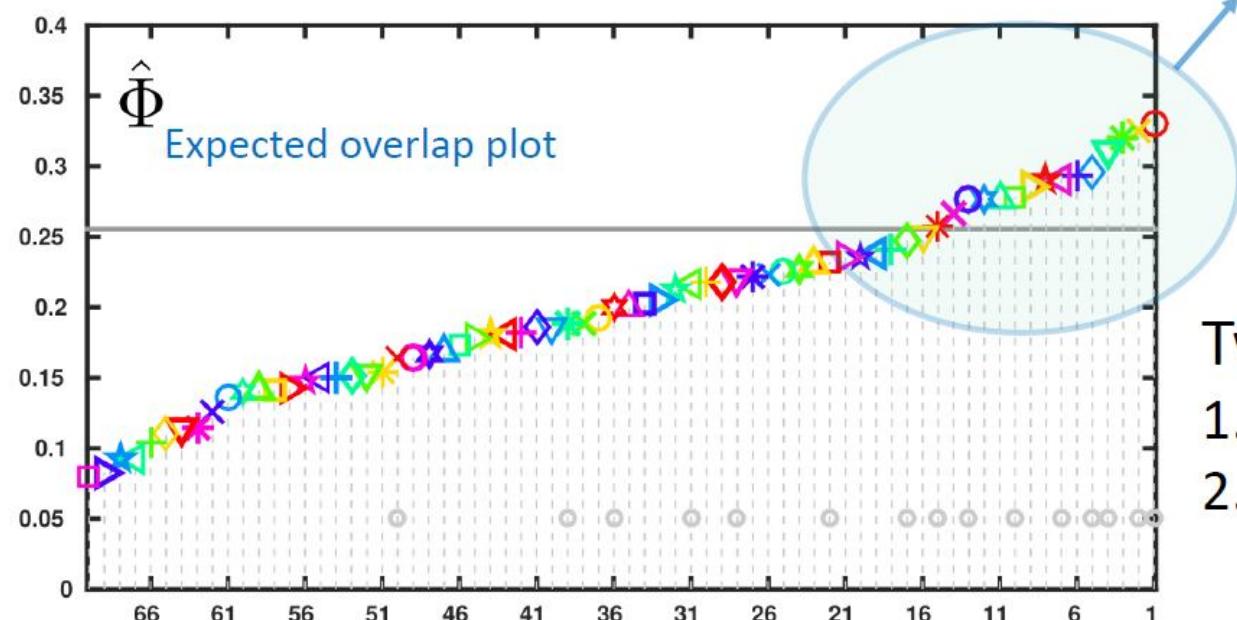
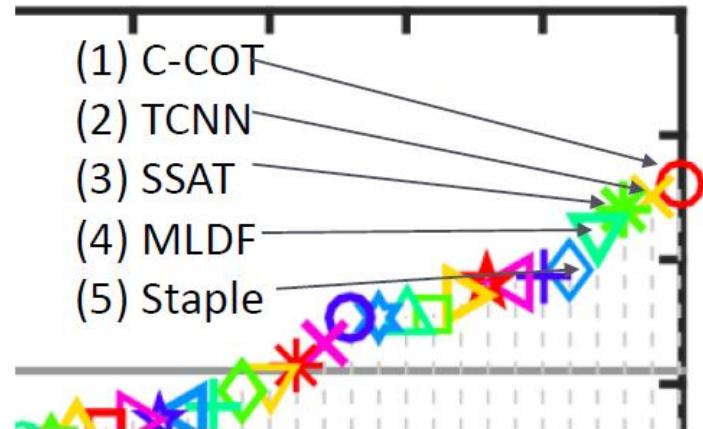
- Initialization on ground truth BBs
- Each tracker **run 15 times** on each sequence to obtain a better statistic on its performance.
- Reinitialization at overlap 0.





# Expected Average Overlap

Tracker	Type
C-COT	○
TCNN	×
SSAT	*
MLDF	▽



Two classes:

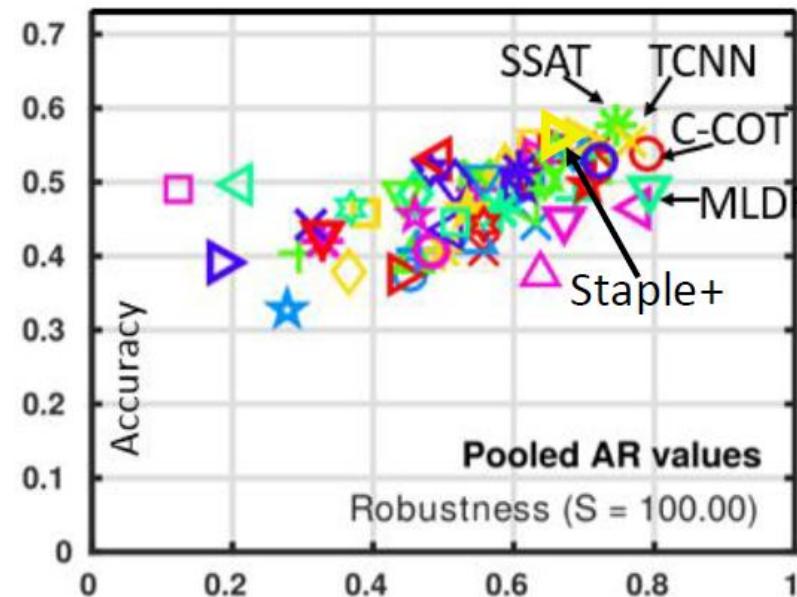
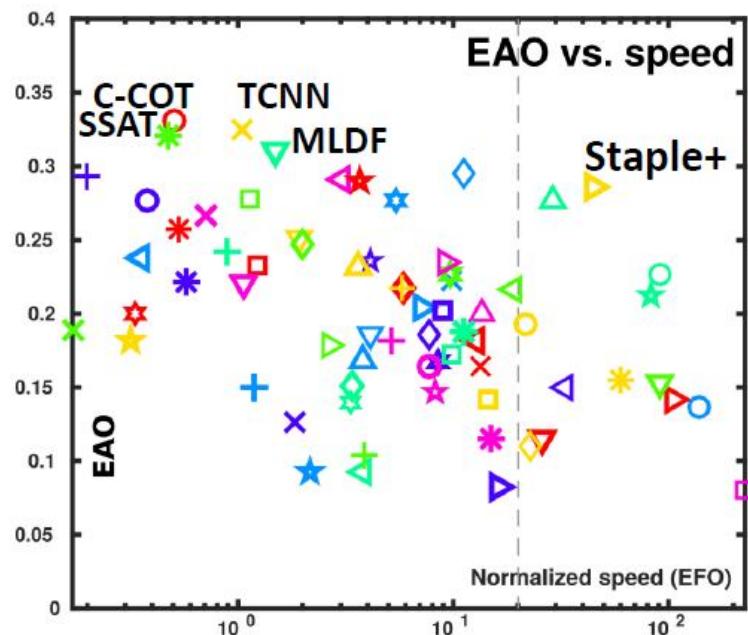
1. CNN-based
2. Correlation filters



# Скорость трекинга

- Top-performers slowest
  - Plausible cause: CNN
  - 
  - 
  -
- Real-time bound: Staple+
- Decent accuracy,
- Decent robustness

**Note:** the speed in some Matlab trackers has been significantly underestimated by the toolkit since it was measuring also the Matlab restart time. The EFOs of Matlab trackers are in fact higher than stated in this figure.





# Примеры последовательностей



- Among the most challenging sequences

Matrix ( $A_f = 0.33, M_f = 57$ )



Rabbit ( $A_f = 0.31, M_f = 43$ )



Butterfly ( $A_f = 0.22, M_f = 45$ )



- Among the easiest sequences

Singer1 ( $A_f = 0.02, M_f = 4$ )



Octopus ( $A_f = 0.01, M_f = 5$ )



Sheep ( $A_f = 0.02, M_f = 15$ )





# Представление объектов



- Как будем описывать объект («модель объекта») ?
  - Шаблон (template)
    - Сюда попадает множество вариантов признаков
  - Набор фрагментов (parts)
  - Признаки объекта (гистограмма признаков)



# Сопоставление шаблонов

- Фиксируем изображение объекта (шаблон – pattern)
- Будем искать положение шаблона в новом кадре
  - Например, в окрестности предыдущего положения
- Попиксельно будем сравнивать шаблон и фрагмент нового кадра с помощью какой-нибудь метрики
- Например, SSD или NCC



$$\frac{1}{n-1} \sum_{x,y} \frac{(f(x,y) - \bar{f})(t(x,y) - \bar{t})}{\sigma_f \sigma_t}$$

(NCC) Normalized cross correlation





## Пример: пульт ТВ

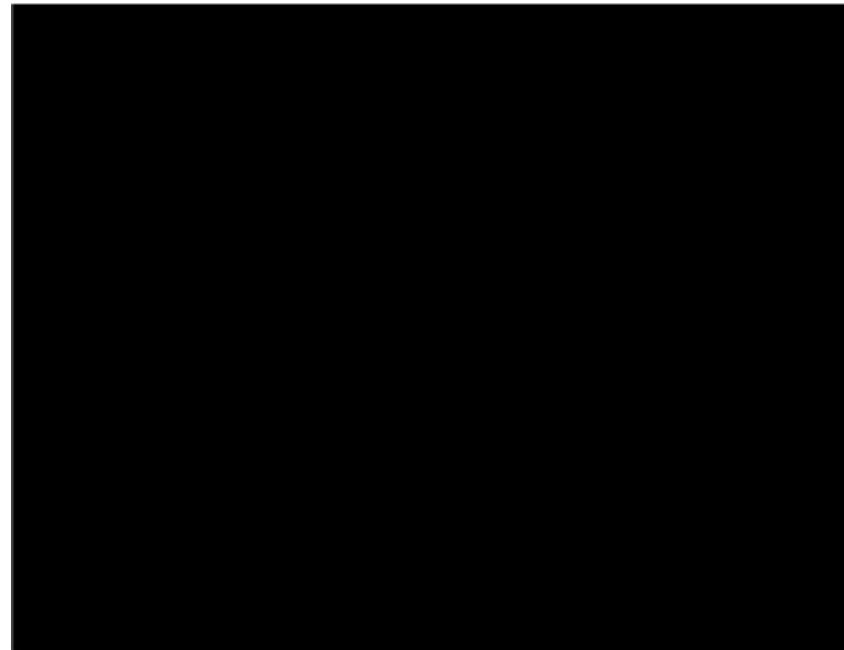


- Шаблон (слева), изображение (в центре), карта нормализованной корреляции (справа)
- Пик яркости (максимум корреляции) соответствует положению руки (искомого шаблона)



## Пример: пульт ТВ

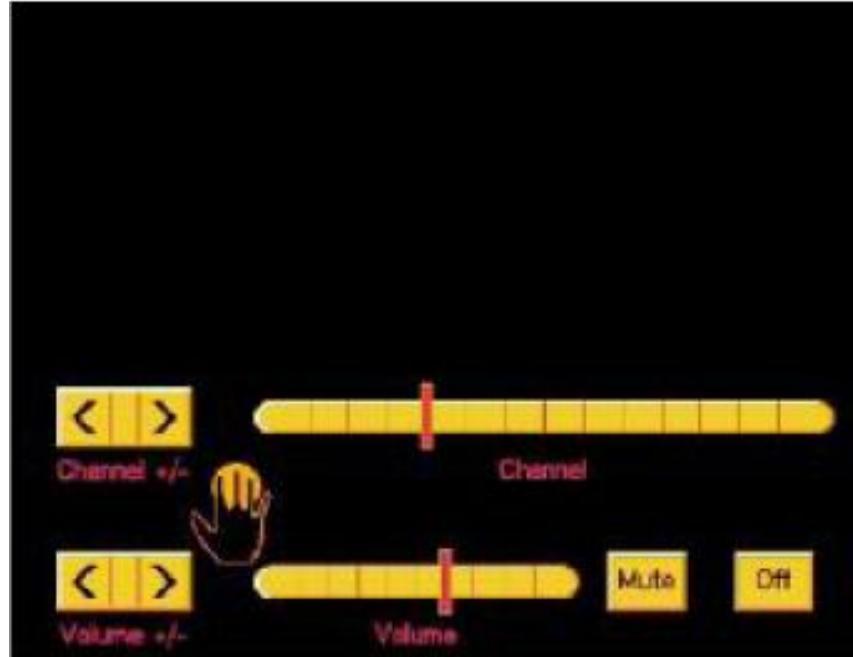
---



Credit: W. Freeman *et al*, “Computer Vision for Interactive Computer Graphics,” *IEEE Computer Graphics and Applications*, 1998

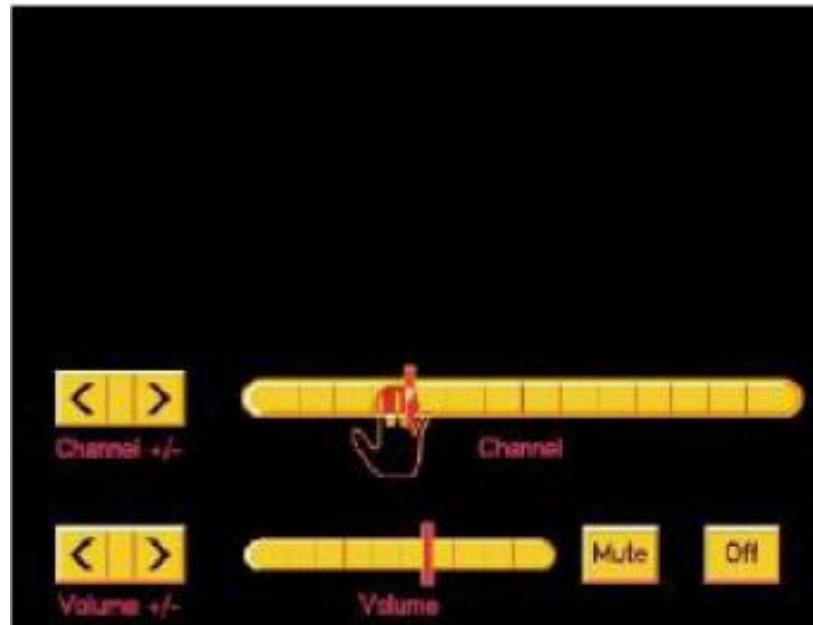


## Пример: пульт ТВ



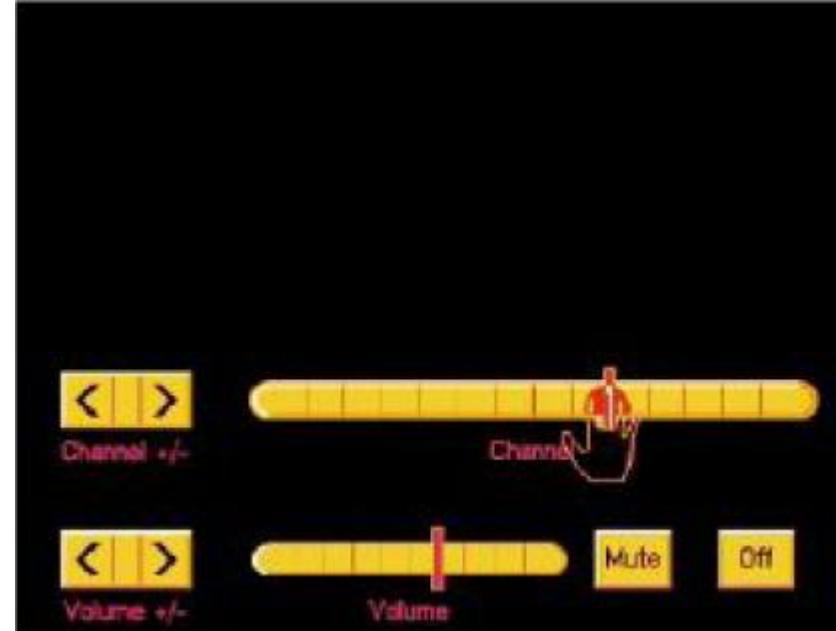


## Пример: пульт ТВ





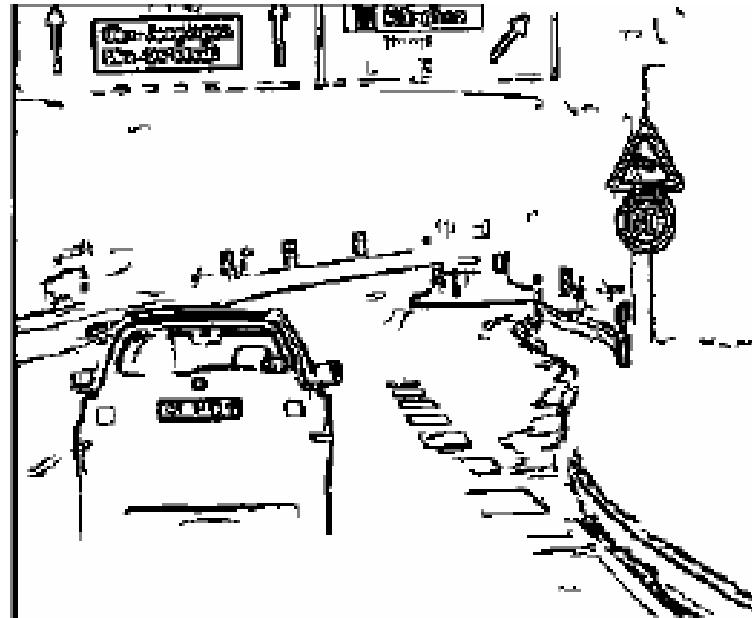
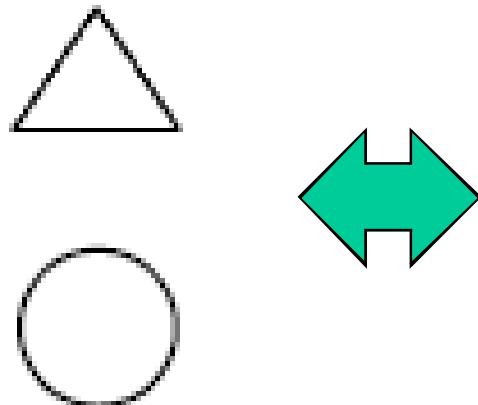
## Пример: пульт ТВ



Не смотря на свою простоту для некоторых задач (например, трекинг лица в хорошем разрешении) NCC достаточно, а скорость максимальна



# Края для сопоставления шаблонов



- Мы знаем, что в края – очень информативный признак, и они устойчивы к изменению освещения
- Попробуем использовать только края для поиска / отслеживания объекта
- Как эффективно сопоставлять карты краев?
  - Попиксельно явно нельзя!



# Метрики

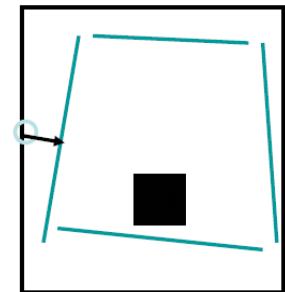
## • Chamfer Distance

- Для каждого пикселя  $a$  края шаблона  $A$  вычисляем расстояние до ближайшего пикселя  $b$  края изображения  $B$

$$r(a, B) = \min_{b \in B} \|a - b\|$$

- Суммируем все найденные расстояния

$$ChDist(A, B) = \sum_{a \in A} \min_{b \in B} \|a - b\|$$



## • Hausdorff Distance

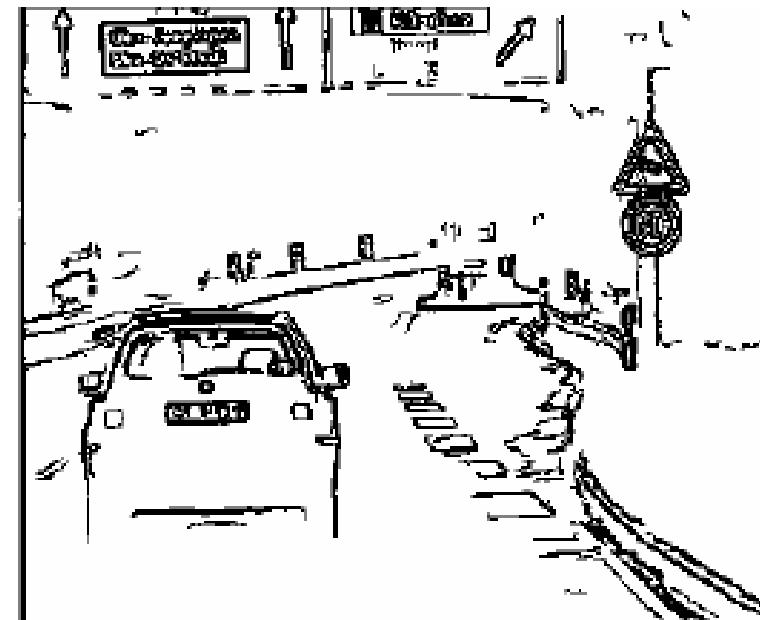
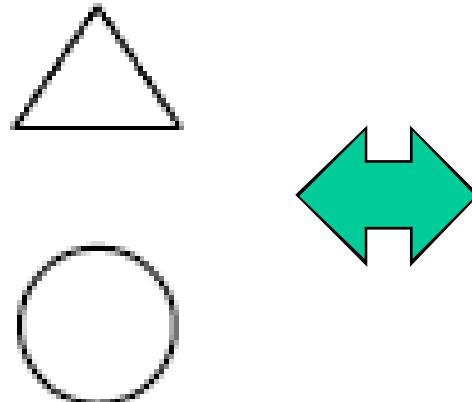
- Почти то же самое, но берём не сумму, а максимальное расстояния

$$HausDist(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

Какую метрику использовать заранее сказать нельзя, нужна экспериментальная проверка



# Поиск ближайших пикселей края



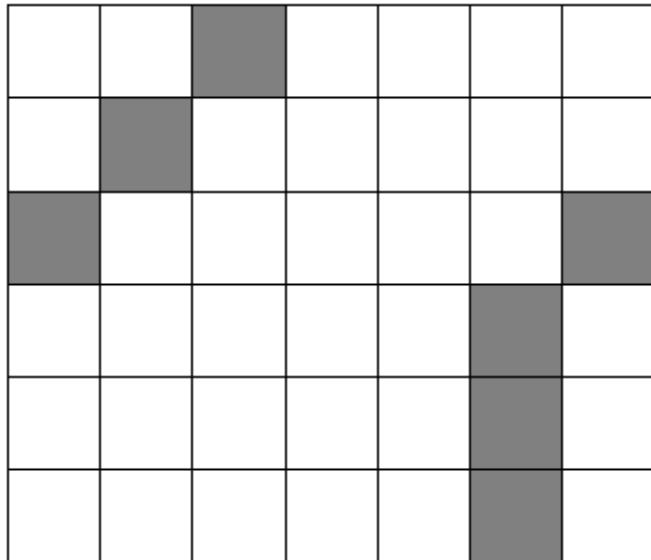
Вопрос: как найти ближайший пиксель края на изображении?



# Distance Transform



«Дистанктное преобразование»



2	1	0	1	2	3	2
1	0	1	2	3	2	1
0	1	2	3	2	1	0
1	2	3	2	1	0	1
2	3	3	2	1	0	1
3	4	3	2	1	0	1

Для каждого пикселя вычисляется расстояние до ближайшего пикселя края

- Многопроходный алгоритм (пометить соседей, потом их соседей и т.д.)
- Двухпроходный алгоритм



# Применение DT

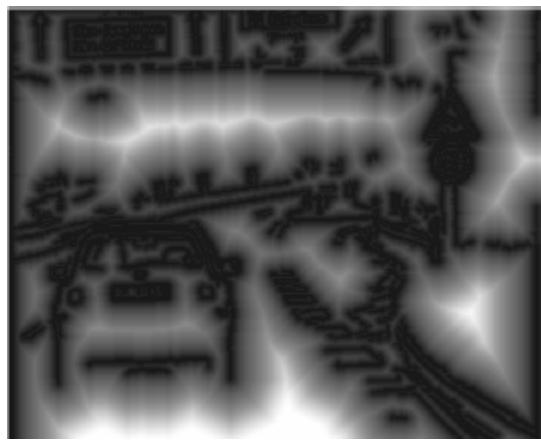
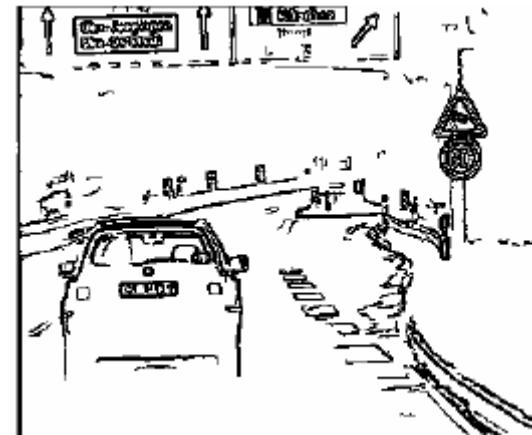
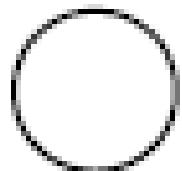
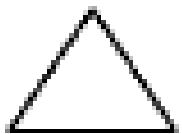


2	1	0	1	2	3	2
1	0	1	2	3	2	1
0	1	2	3	2	1	0
1	2	3	2	1	0	1
2	3	3	2	1	0	1
3	4	3	2	1	0	1

- Совмещаем шаблон и карту DT
- Вычисляем ошибку, суммирую все значения в пикселях краев (для Chamfer distance)



# Пример поиска с помощью DT





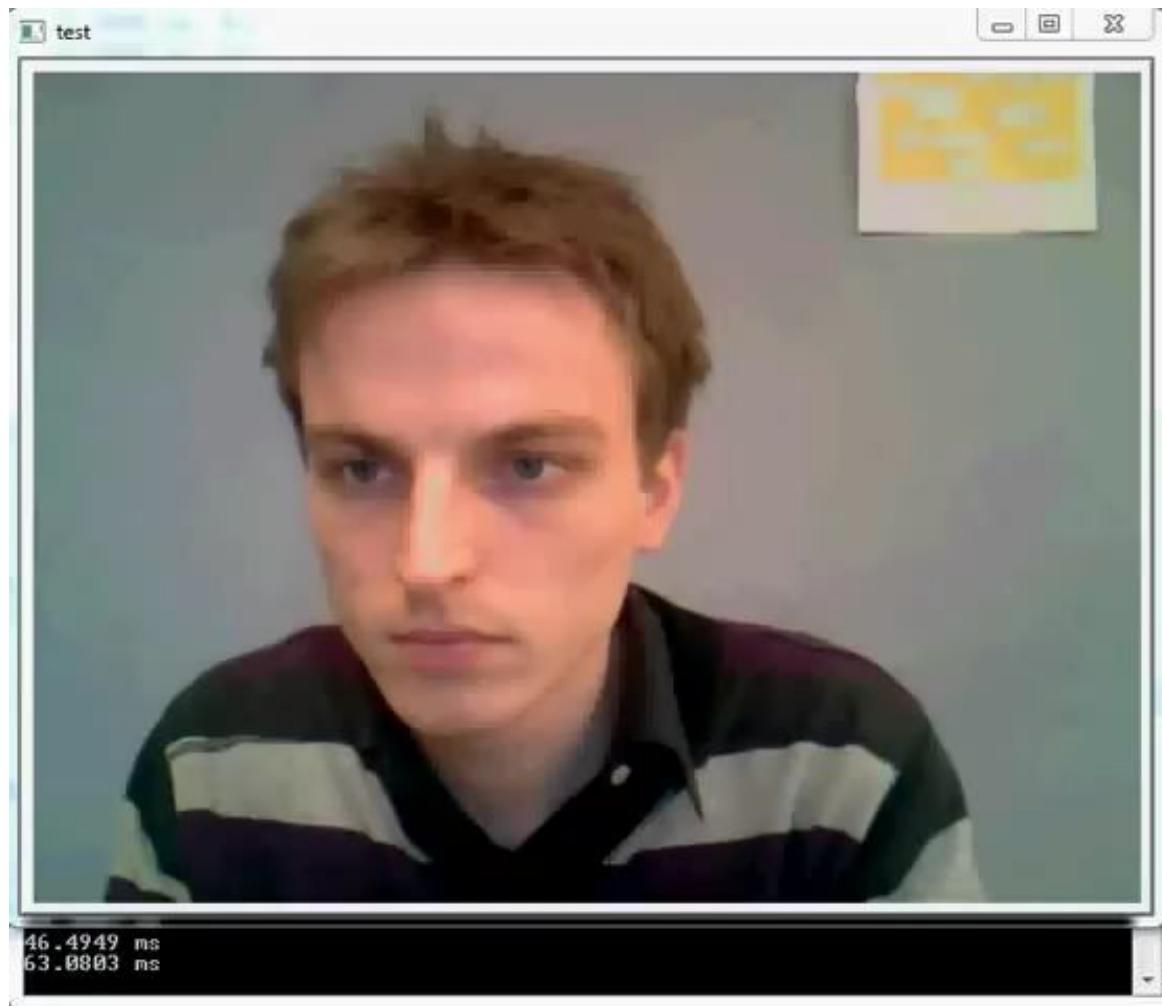
# Сопровождение



- Карта краёв шаблона используется для дальнейшего сравнения
- Вычисляется метрика Хаусдорфа на основании DT
- Шаблон обновляется как набор краёв, ближайших к краям шаблона предыдущего кадра
- HUTTENLOCHER, D., NOH, J., AND RUCKLIDGE, W.. Tracking nonrigid objects in complex scenes. ICCV 1993



# Пример работы





# Множество точек

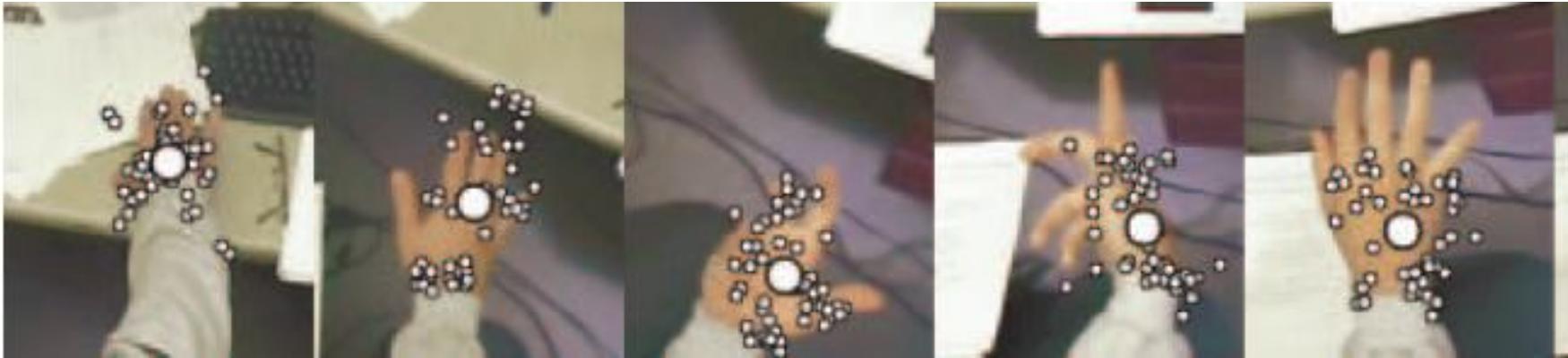
---

- Мы умеем вычислять движение отдельных точек с помощью локального метода оценки оптического потока Lucas-Kanade
- Но каждая точка в отдельности может быстро сбиться из-за ошибок в вычислении оптического потока и его несоответствия движению точек сцены
- Решение – использование «стай точек» («flock of features»)





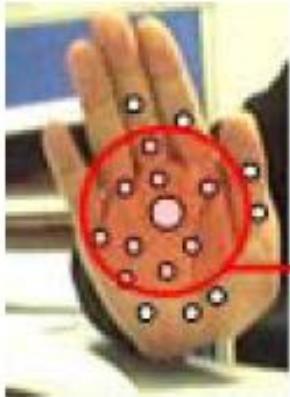
## Пример: отслеживание руки



- Стая – множество контрольных точек, удовлетворяющих 2м условиям:
  - Никакие две контрольные точки не совпадают (порог на близость)
  - Никакая контрольные точки не уходят далеко от медианного центра (порог на удаление)



# Пример: отслеживание руки



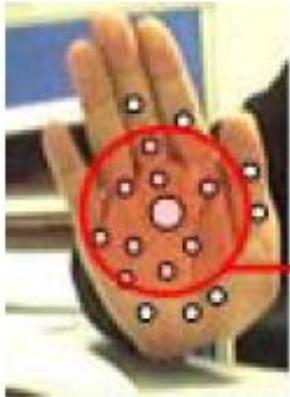
- Feature
- Feature mean position
- Skin color collect area



- Шаг 1: инициализация
  - Находим 100 контрольных точек с помощью метода поиска локальных особенностей (Harris corners) в рамке руки
  - Вычисляем медиану
  - Вычисляем цветовую статистику в окрестности центра
    - Одна гауссиана (или гистограмма)
    - Это модель кожи
  - Разметить в рамке руки все пиксели, похожие на кожу



# Пример: отслеживание руки



- Feature
- Feature mean position
- Skin color collect area



- Шаг 2: слежение
  - Отслеживаем контрольные точки
  - Если точка нарушает условия стаи, её удаляем
- Шаг 3: инициализация новых контрольных точек
  - Ищем особенности (Harris corners)
  - Если точка не на коже, то отбрасываем её

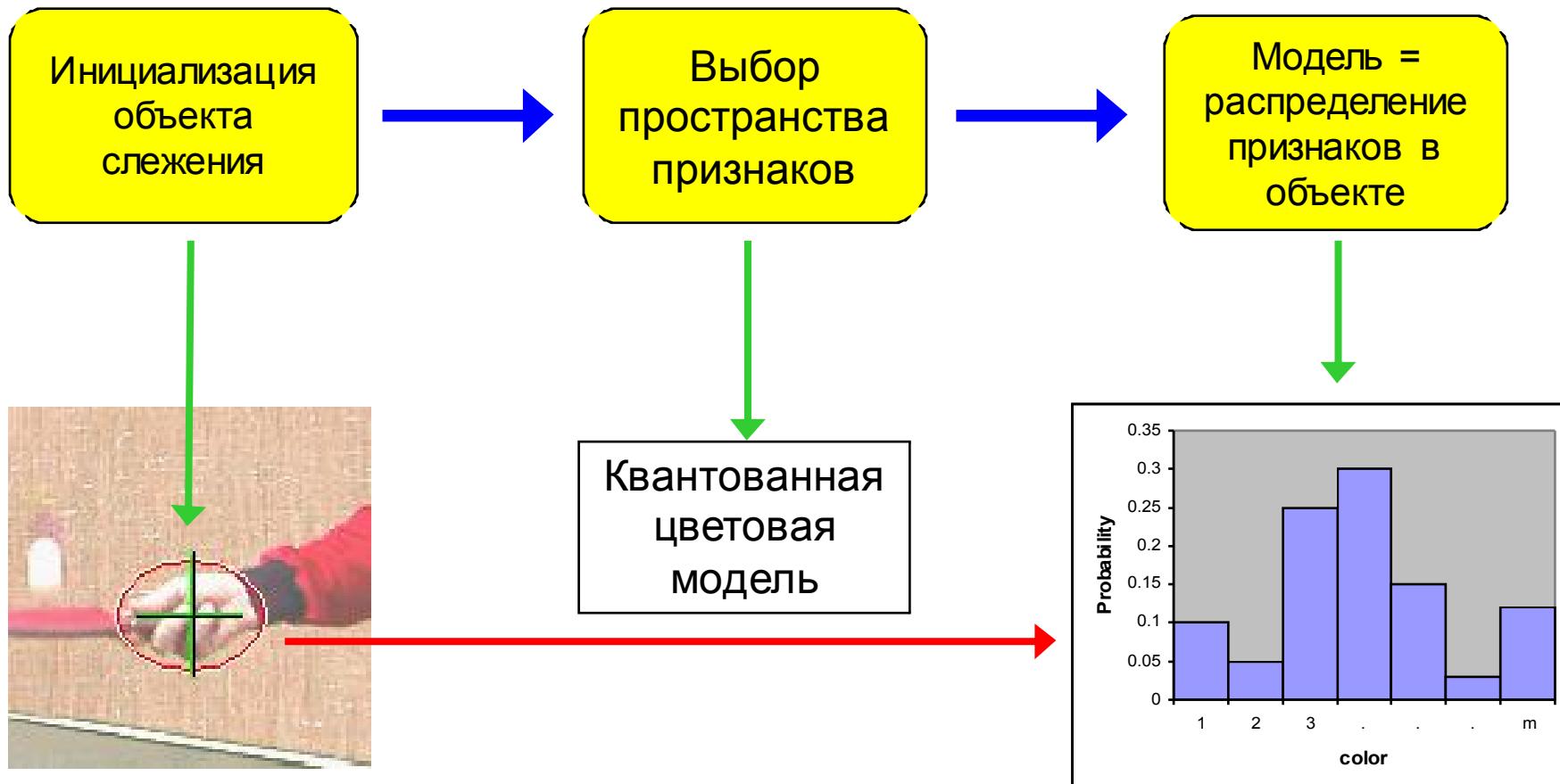


# Пример работы





# Mean-Shift («Сдвиг среднего»)



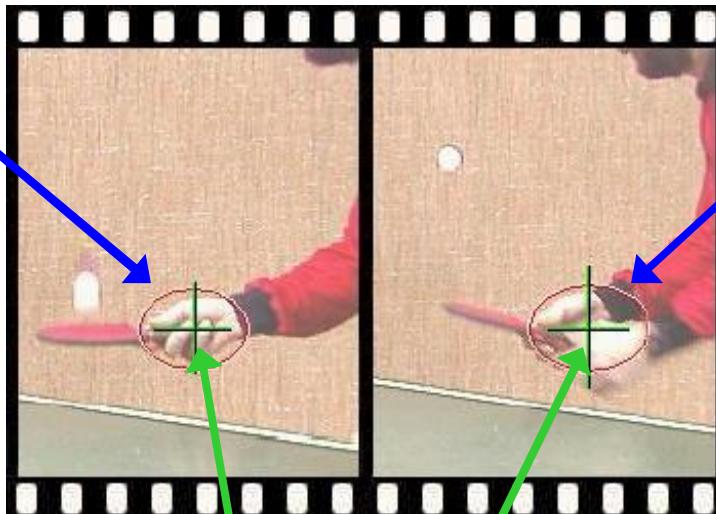
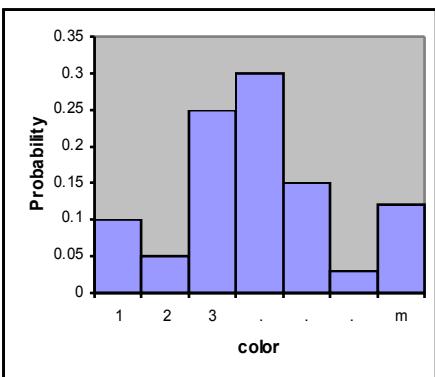
- Вместо жесткого шаблона вычислим вектор-признак по области объекта
- Например, гистограмму распределения цветов



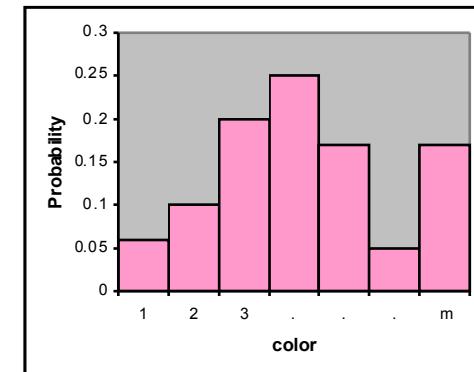
# Слежение сдвигом среднего



Модель  
(центр в 0)



Кандидат  
(центр в y)



$$\vec{q} = \{q_u\}_{u=1..m} \quad \sum_{u=1}^m q_u = 1$$

Сходство:  $f(y) = f[\vec{q}, \vec{p}(y)]$

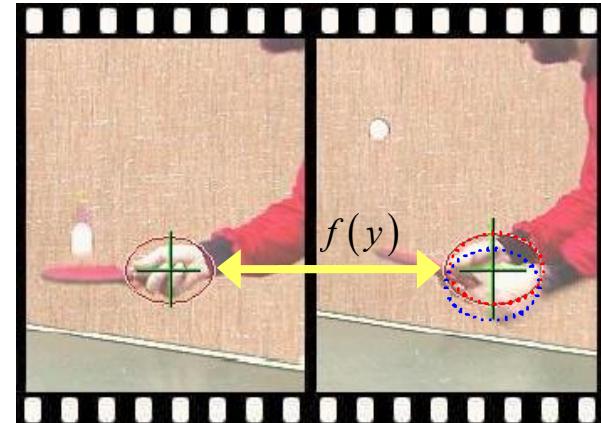
$$\vec{p}(y) = \{p_u(y)\}_{u=1..m} \quad \sum_{u=1}^m p_u = 1$$

Будем сравнивать гистограммы опорной модели и положений-кандидатов на новом кадре



# Слежение сдвигом среднего

Сходство:  $f(y) = f[\vec{p}(y), \vec{q}]$



- Методы сопоставления шаблонов страдают от проблемы негладкости целевой функции
  - Небольшие смещения могут привести к резким скачкам ошибки сопоставления
  - Поэтому мы не можем использовать, например, метод градиентного спуска для поиска оптимального положения
- Как мы решали эту проблему при сопоставлении точек?
- Mean-Shift – развитие идеи для сопровождения

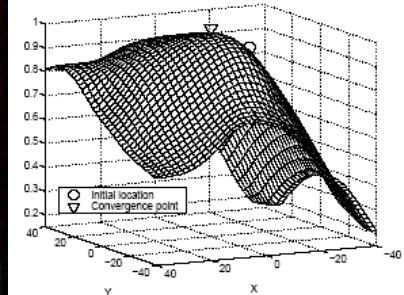
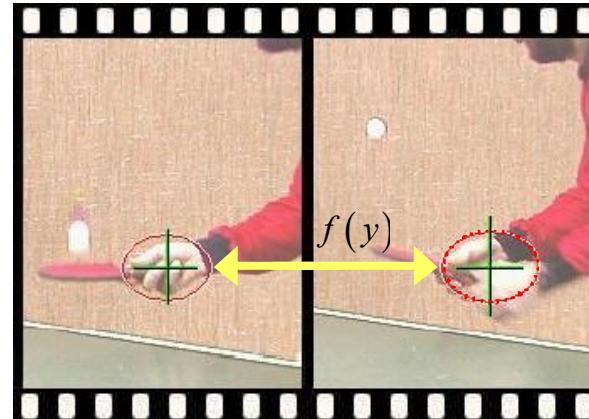
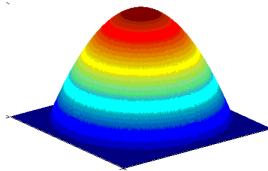


# Слежение сдвигом среднего



Сходство:  $f(y) = f[\vec{p}(y), \vec{q}]$

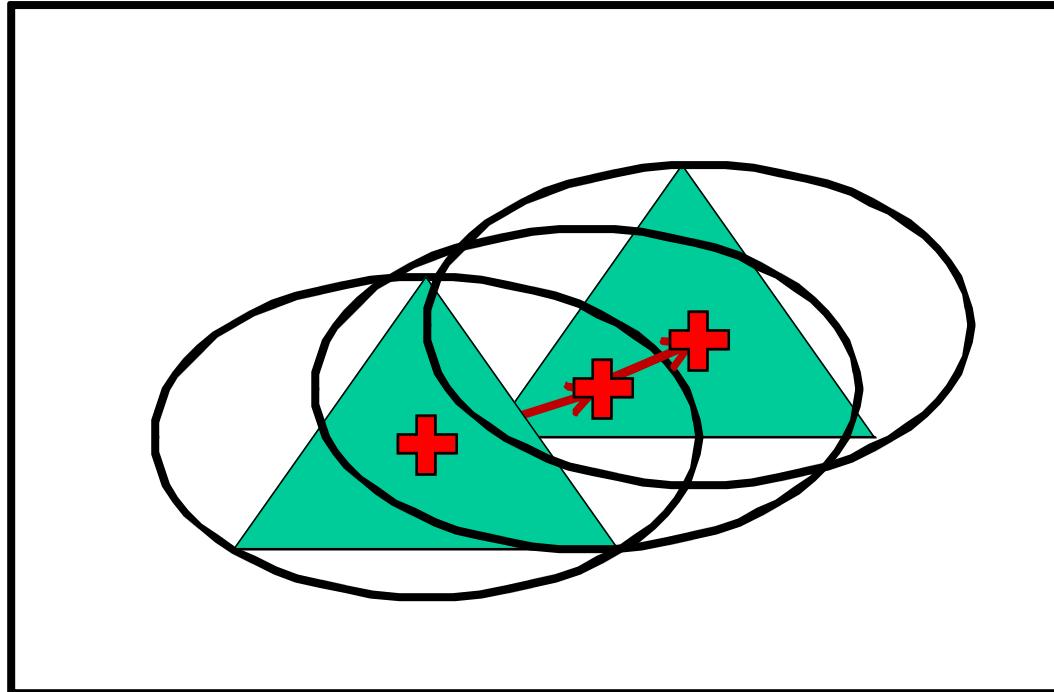
Изотропное ядро:



- Покроем цель изотропным ядром. Веса на границе цели близки к нулю
- Теперь небольшие смещения приводят к небольшим изменениям ошибки сопоставления
- Ошибка сопоставления становится гладкой (по сдвигу от оптимального положения)
- Теперь мы можем не перебирать все положения в окрестности, а воспользоваться методом градиентного спуска



# Иллюстрация сдвига среднего



Для определенных ядер можно показать, что метод градиентного спуска превращается в «сдвиг среднего» :

Взвешенная  
сумма точек

$$y_1 = \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i}$$

расчет весов

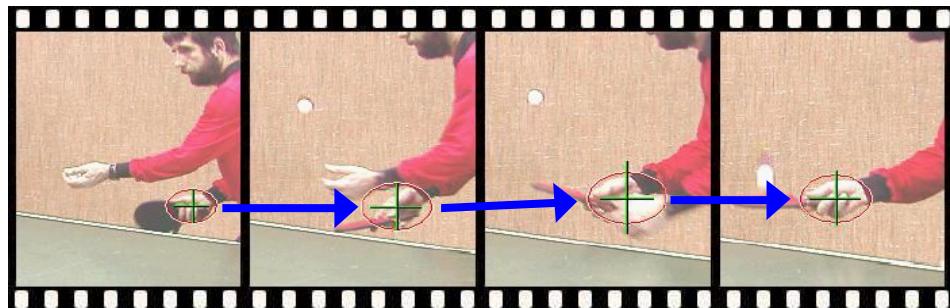
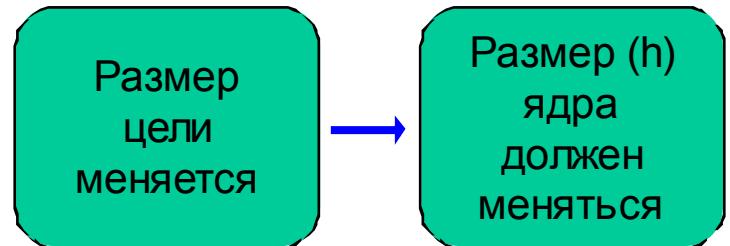
$$w_i = \sum_{u=1}^m \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_0)}} \delta [b(\mathbf{x}_i) - u]$$



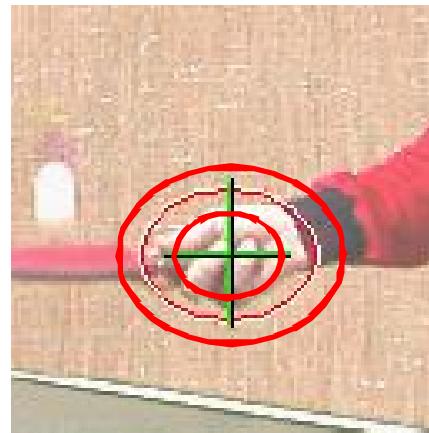
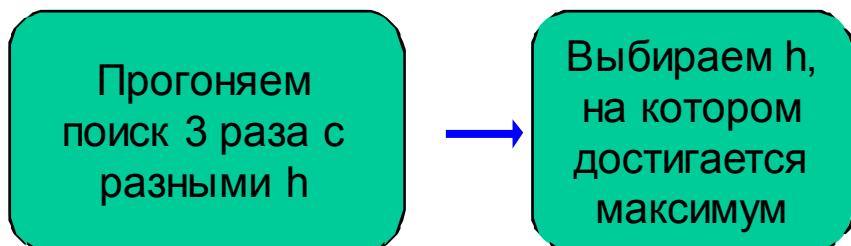
# Слежение сдвигом среднего



## Проблема :



## Решение:



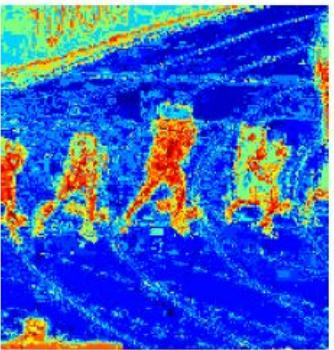
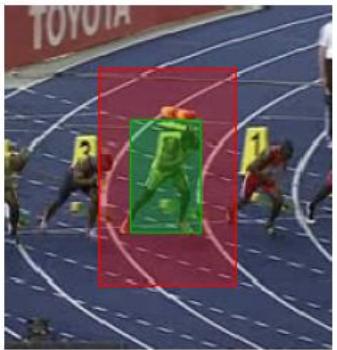


## Пример работы

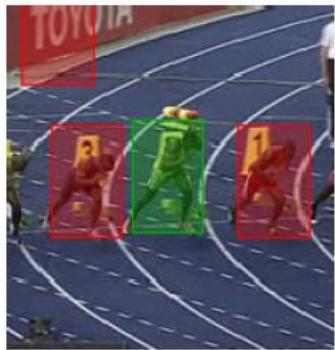




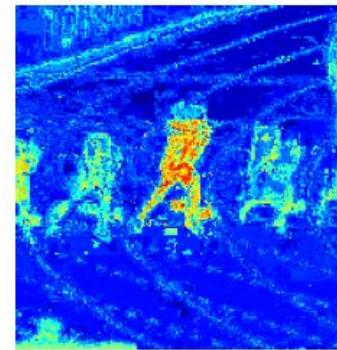
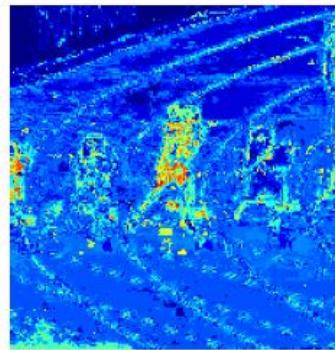
# Развитие цветовых моделей



(a) Object-surrounding:  $P(\mathbf{x} \in \mathcal{O} | O, S, b_{\mathbf{x}})$ .



(b) Object-distractors:  $P(\mathbf{x} \in \mathcal{O} | O, D, b_{\mathbf{x}})$ .



(c) Combined.

$$P(\mathbf{x} \in \mathcal{O} | O, S, b_{\mathbf{x}}) = \begin{cases} \frac{H_O^I(b_{\mathbf{x}})}{H_O^I(b_{\mathbf{x}}) + H_S^I(b_{\mathbf{x}})} & \text{if } I(\mathbf{x}) \in I(O \cup S) \\ 0.5 & \text{otherwise,} \end{cases} \quad (2)$$

$$P(\mathbf{x} \in \mathcal{O} | O, D, b_{\mathbf{x}}) = \begin{cases} \frac{H_O^I(b_{\mathbf{x}})}{H_O^I(b_{\mathbf{x}}) + H_D^I(b_{\mathbf{x}})} & \text{if } I(\mathbf{x}) \in I(O \cup D) \\ 0.5 & \text{otherwise.} \end{cases} \quad (3)$$

Итог – линейная комбинация

H. Possegger, T. Mauthner, and H. Bischof. In Defense of Color-based Model-free Tracking. In CVPR, 2015.



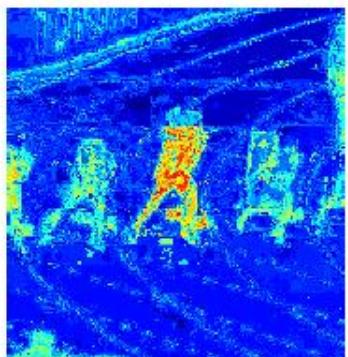
# Слежение



$$O_t^* = \arg \max_{O_{t,i}} (s_v(O_{t,i})s_d(O_{t,i})), \quad \text{-задача слежения}$$

$$s_v(O_{t,i}) = \sum_{\mathbf{x} \in O_{t,i}} P_{1:t-1} (\mathbf{x} \in \mathcal{O} | b_{\mathbf{x}}), \quad \text{-визуальная схожесть}$$

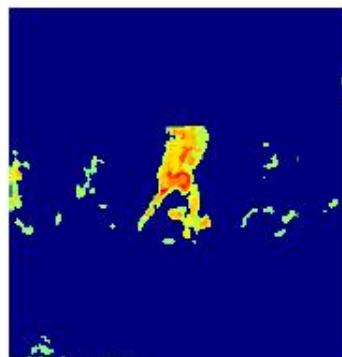
$$s_d(O_{t,i}) = \sum_{\mathbf{x} \in O_{t,i}} \exp \left( -\frac{\|\mathbf{x} - \mathbf{c}_{t-1}\|^2}{2\sigma^2} \right), \quad \text{-близость к предыдущему положению}$$



(a) Likelihood map  $L$ .



(b) Regions  $O$  and  $S$ .

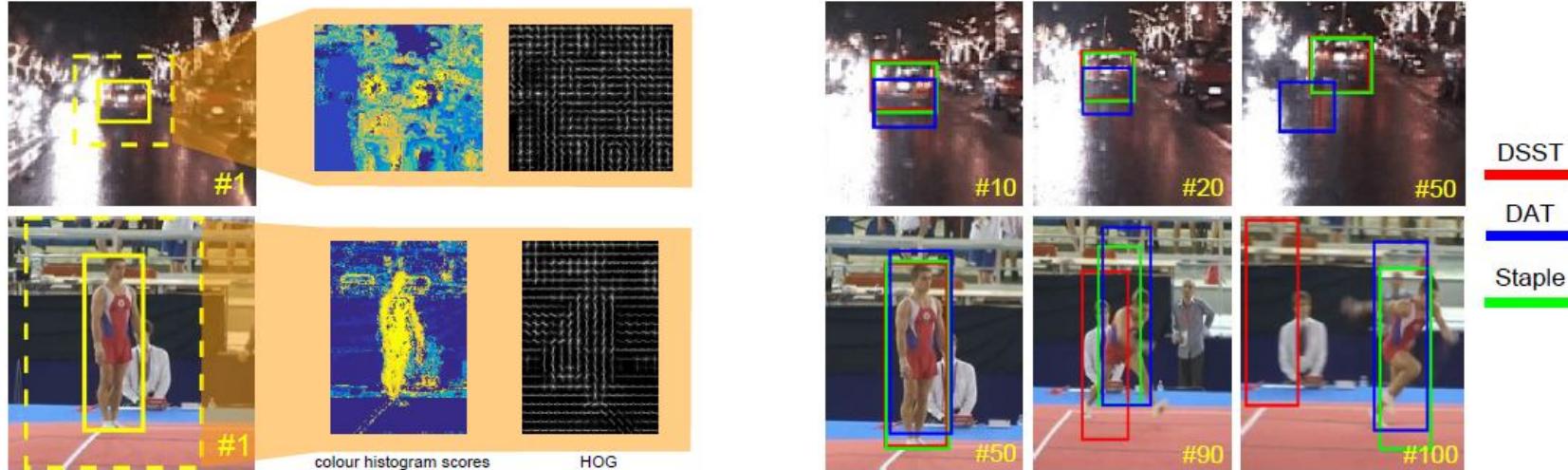


(c) Segmentation.

- Сегментация  
объекта и  
определение нового  
размера bbox



# Staple



- ▶ Colour distributions rely on pixel values to discriminate target from background → no concept of locality: robust to shape changes, sensitive to blur and poor illumination.
- ▶ Template models rely on spatial configuration →robust to blur and poor illumination, sensitive to shape changes.

Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.H.S.: Staple: Complementary learners for real-time tracking. In: CVPR. (2016)



---

## Staple = Sum of Template and Pixel-wise Learners.

- ▶ Combination of score functions  $f_{\text{tmpl}}$  and  $f_{\text{hist}}$  evaluated on complementary cues

$$f(x) = \gamma_{\text{tmpl}} f_{\text{tmpl}}(x) + \gamma_{\text{hist}} f_{\text{hist}}(x)$$

- ▶  $f_{\text{tmpl}}(x; h)$ : linear function of HOG feature image  $\phi_x$ .
- ▶  $f_{\text{hist}}(x; \beta)$ : average of a score image computed from the histogram model  $\beta$ .
- ▶ Two ridge regression problems at each  $t$  depending on previous images/locations  $\mathcal{X}_t$  :

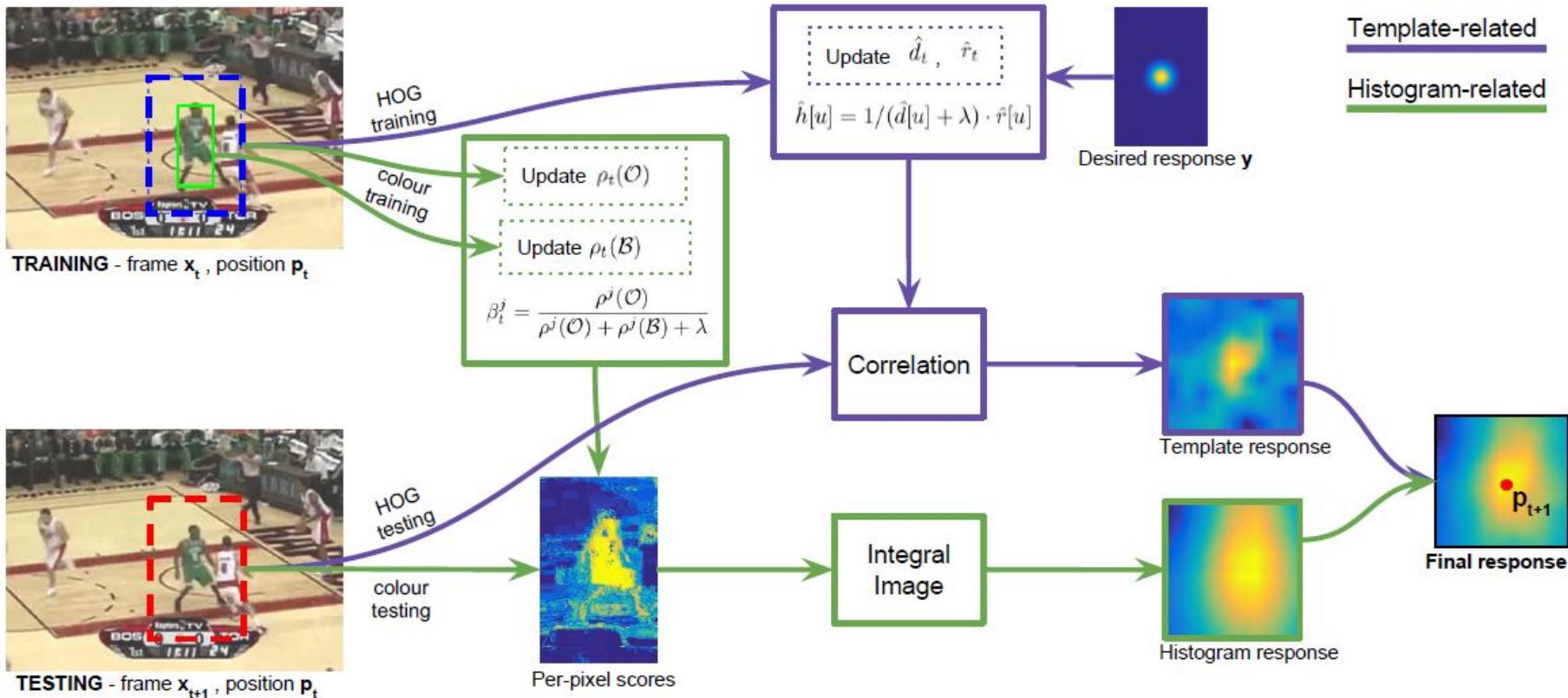
$$h_t = \arg \min_h \{ L_{\text{tmpl}}(h; \mathcal{X}_t) + \frac{1}{2} \lambda_{\text{tmpl}} \|h\|^2 \}$$

$$\beta_t = \arg \min_{\beta} \{ L_{\text{hist}}(\beta; \mathcal{X}_t) + \frac{1}{2} \lambda_{\text{hist}} \|\beta\|^2 \}$$

- ▶ Correlation (in Fourier domain) and Integral Image → dense responses with fast sliding window search.



# Схема работы





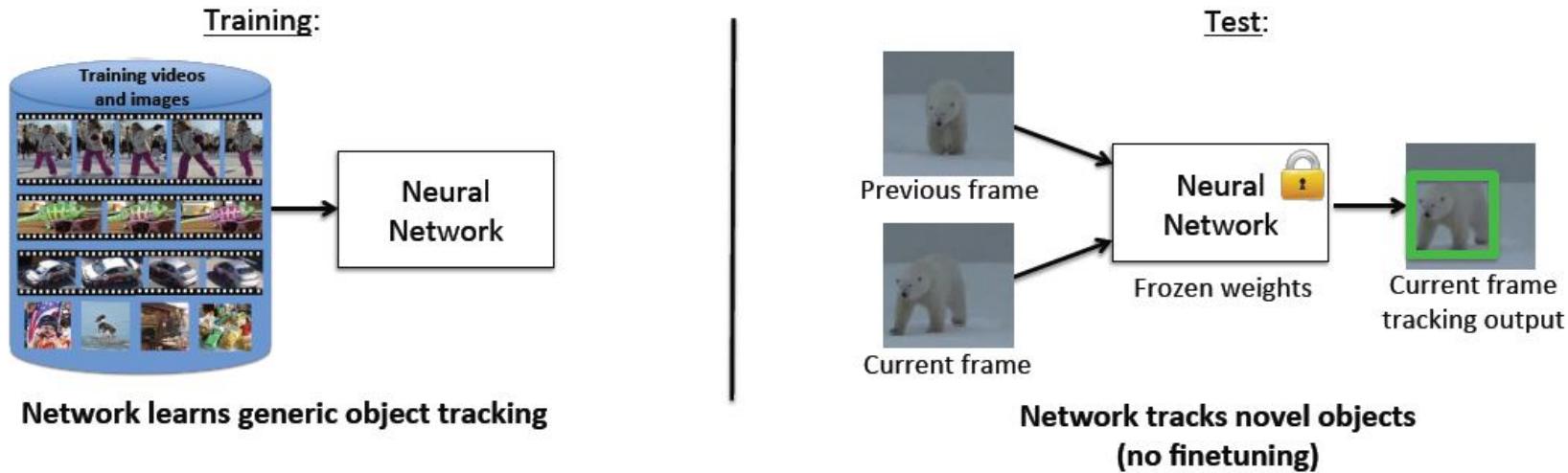
# Результаты на VOT

Tracker	Year	Where	Accuracy	# Failures	Rank	Speed (fps)
MDNet	2015	ICCV	0.583	0.69	14.31	1
DeepSRDCF	2015	VOT	0.528	1.05	19.16	< 1
SRDCF	2015	ICCV	0.521	1.24	21.01	5
<b>Staple</b>	2016	CVPR	0.533	1.39	21.64	<b>80</b>
SO-DLT	2015	arXiv	0.535	1.78	22.71	5
NSAMF	2015	VOT	0.490	1.29	22.93	5
EBT	2015	arXiv	0.453	1.02	23.01	5
sPST	2015	ICCV	0.508	1.48	23.04	2
RAJSSC	2015	VOT	0.518	1.63	23.53	2
SC-EBT	2015	ICML	0.523	1.86	23.70	-

Table 2 : VOT15 top 10 (of 63)



# GOTURN, Generic Object Tracking Using Regression Networks

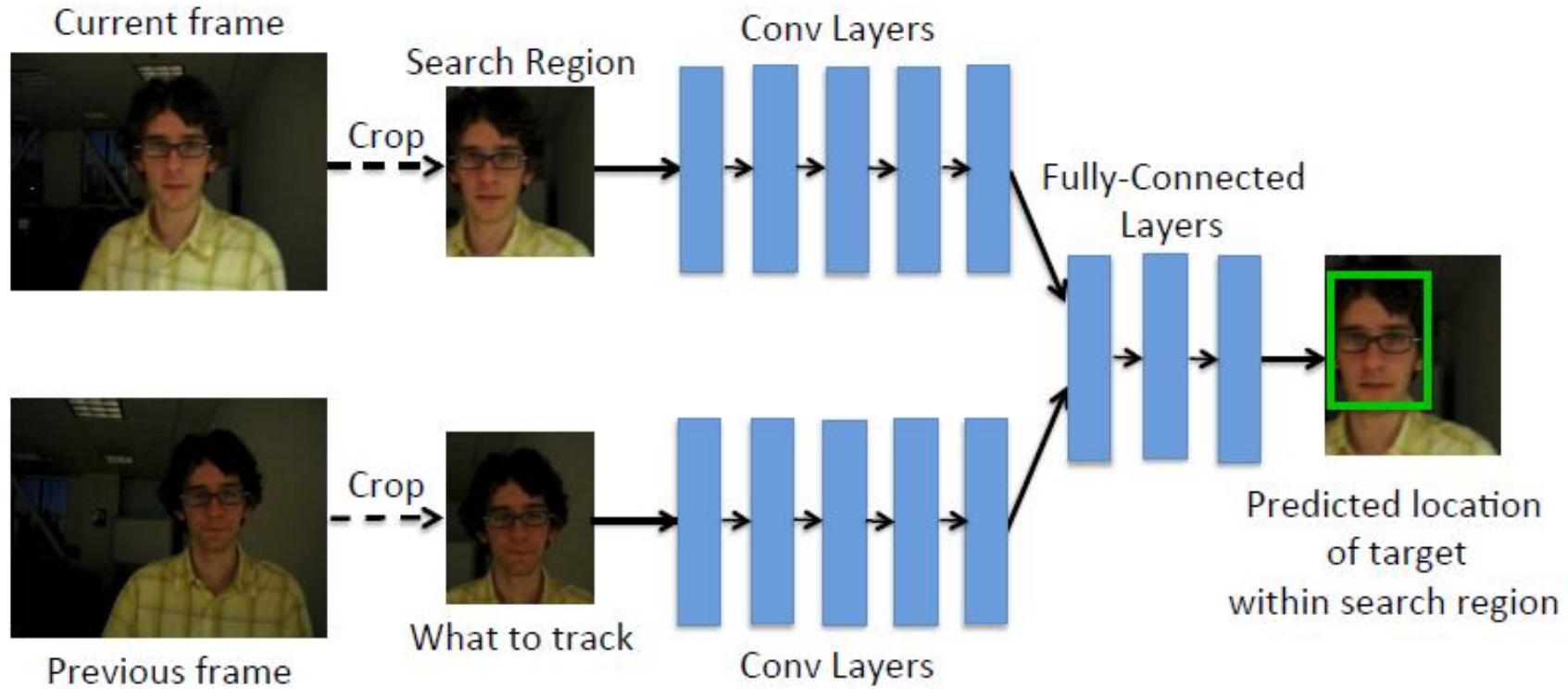


**Fig. 1.** Using a collection of videos and images with bounding box labels (but no class information), we train a neural network to track generic objects. At test time, the network is able to track novel objects without any fine-tuning. By avoiding fine-tuning, our network is able to track at 100 fps

[Learning to Track at 100 FPS with Deep Regression Networks, David Held, Sebastian Thrun, Silvio Savarese, European Conference on Computer Vision \(ECCV\), 2016 \(In press\)](#)



# Архитектура сети





# Обучение на видео и изображениях



Previous  
video frame  
centered on  
object



Current video frame,  
shifted, with  
ground-truth  
bounding box

Image  
centered on  
object



Shifted image  
with ground-truth  
bounding box

Сэмплинг малых смещений чаще, чем больших

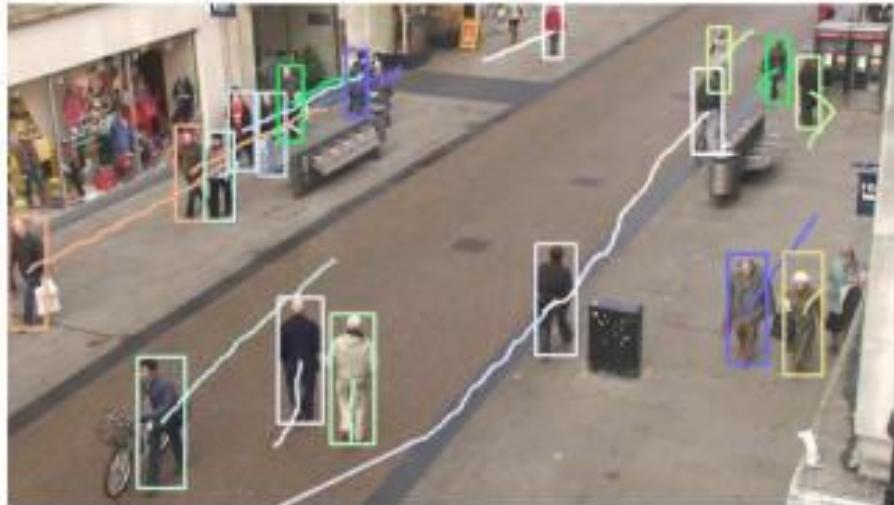


# Резюме

---



- VOT-методы пригодны для сопровождения произвольного объекта на коротких временных отрезках
- Как это часто бывает, ищем компромисс надежность vs скорость
- Лучшие методы очень медленные (~1 кадр / сек и медленнее)
- Достаточно много подходов
- Лучшие методы комбинируют признаки и классификаторы
- Сейчас развиваются CNN-методы



## Multiple object tracking



# Multiple object tracking



- Работаем со множеством объектов
- На длительных промежутках времени
- Варианты:
  - Есть модель объектов (возможность повторного обнаружения)
    - » Detection Based Tracking (DBT)
  - Нет модели объектов, все объекты только на первом кадре выделяются
    - » Detection Free Tracking (DFT)



# Online vs Offline трекинг

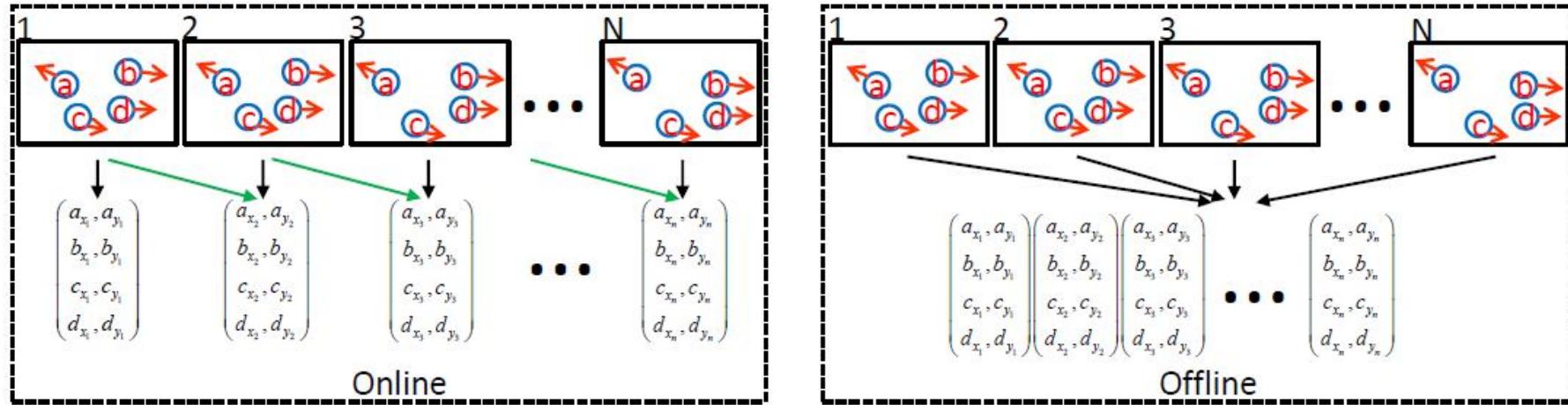


Fig. 4: Illustration of online and offline tracking. Best viewed in color



# Постановка задачи

Задача:

$$\hat{\mathbf{S}}_{1:t} = \arg \max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t} | \mathbf{O}_{1:t}). \quad \mathbf{O} - \text{наблюдения, } \mathbf{S} - \text{состояния объектов}$$

Вероятностный вывод:

Predict:  $P(\mathbf{S}_t | \mathbf{O}_{1:t-1}) = \int P(\mathbf{S}_t | \mathbf{S}_{t-1}) P(\mathbf{S}_{t-1} | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1}$

Update:  $P(\mathbf{S}_t | \mathbf{O}_{1:t}) \propto P(\mathbf{O}_t | \mathbf{S}_t) P(\mathbf{S}_t | \mathbf{O}_{1:t-1})$

Модель наблюдения  
(observation model)

Модель движения  
(dynamic model)

Сведение к минимизации энергии:

$$\begin{aligned}\hat{\mathbf{S}}_{1:t} &= \arg \max_{\mathbf{S}_{1:t}} P(\mathbf{S}_{1:t} | \mathbf{O}_{1:t}) \\ &= \arg \max_{\mathbf{S}_{1:t}} \frac{1}{Z} \exp(-C(\mathbf{S}_{1:t} | \mathbf{O}_{1:t})) \\ &= \arg \min_{\mathbf{S}_{1:t}} C(\mathbf{S}_{1:t} | \mathbf{O}_{1:t}),\end{aligned}$$



# Метрики



Table 6: An overview of evaluation metrics for MOT. The up arrow (*resp.* down arrow) indicates that the performance is better if the quantity is greater (*resp.* smaller)

Type	Concern	Metric	Description	Note
Detection	Accuracy	Recall	correctly matched detections over ground-truth detections	↑
		Precision	correctly matched detections over result detections	↑
		FAF/FPPI	number of false alarms averaged over a sequence	↓
		MODA	take the miss detection, false positive rate into account	↑
	Precision	MODP	the overlap between true positives and ground truth	↑
Tracking	Accuracy	MOTA	take the false negative, false positive and mismatch rate into account	↑
		IDS	the number of times that a tracked trajectory changes its matched ground-truth identity	↓
	Precision	MOTP	overlap between the estimated positions and the ground truth averaged over the matches	↑
		TDE	difference between the ground-truth annotation and the tracking result	↓
	Completeness	MT	percentage of ground-truth trajectories which are covered by tracker output for more than 80% in length	↑
		ML	percentage of ground-truth trajectories which are covered by tracker output for less than 20% in length	↓
		PT	1.0 - MT - ML	-
		FM	the number of times that a ground-truth trajectory is interrupted in tracking result	↓
	Robustness	RS	the ratio of tracks which are correctly recovered from short occlusion	↑
		RL	the ratio of tracks which are correctly recovered from long occlusion	↑



# Наборы данных

Data set	Multi-view	Ground truth	Web link
PETS 2009	✓	✓	<a href="http://www.cvg.rdg.ac.uk/PETS2009/a.html">www.cvg.rdg.ac.uk/PETS2009/a.html</a>
PETS 2006	✓	✗	<a href="http://www.cvg.rdg.ac.uk/PETS2006/data.html">www.cvg.rdg.ac.uk/PETS2006/data.html</a>
PETS 2007	✓	✓	<a href="http://www.pets2007.net/">www.pets2007.net/</a>
CAVIAR	✓	✓	<a href="http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/">http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/</a>
Trecvid 2008	✓	✗	<a href="http://www-nlpir.nist.gov/projects/tv2008/">www-nlpir.nist.gov/projects/tv2008/</a>
TUD	✗	✓	<a href="http://www.d2.mpi-inf.mpg.de/datasets">www.d2.mpi-inf.mpg.de/datasets</a>
Caltech Pedestrian	✗	✓	<a href="http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/">www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/</a>
UBC Hockey	✗	✗	<a href="http://www.cs.ubc.ca/~okumak/research.html">www.cs.ubc.ca/~okumak/research.html</a>
Lids AVSS 2007	✗	✓	<a href="http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html">www.eecs.qmul.ac.uk/~andrea/avss2007_d.html</a>
ETH pedestrian	✓	✓	<a href="http://www.vision.ee.ethz.ch/~aess/dataset/">www.vision.ee.ethz.ch/~aess/dataset/</a>
ETHZ Central	✗	✓	<a href="http://www.vision.ee.ethz.ch/datasets/">www.vision.ee.ethz.ch/datasets/</a>
Town Centre	✗	✓	<a href="http://www.robots.ox.ac.uk/ActiveVision/Research/Projects/2009bbenfold_headpose/project.html#datasets">www.robots.ox.ac.uk/ActiveVision/Research/Projects/2009bbenfold_headpose/project.html#datasets</a>
Zara	✗	✗	<a href="https://graphics.cs.ucy.ac.cy/research/downloads/crowd-data">https://graphics.cs.ucy.ac.cy/research/downloads/crowd-data</a>
UCSD	✗	✗	<a href="http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm">http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm</a>
UCF Crowds	✗	✗	<a href="http://www.crcv.ucf.edu/data/crowd.php">www.crcv.ucf.edu/data/crowd.php</a>



# Основной подход для DBT



## «Tracking by detection»



### Шаг 1: выделение объектов

- Вычитание фона
- Детекторы объектов

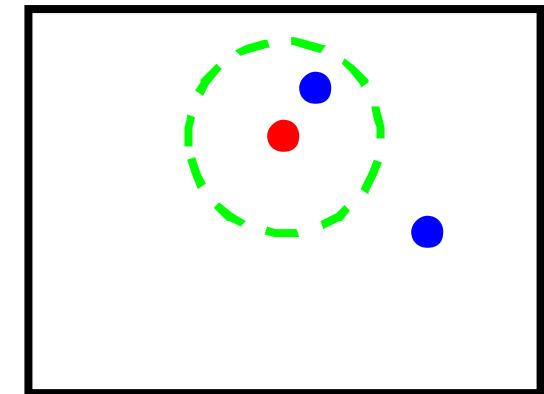
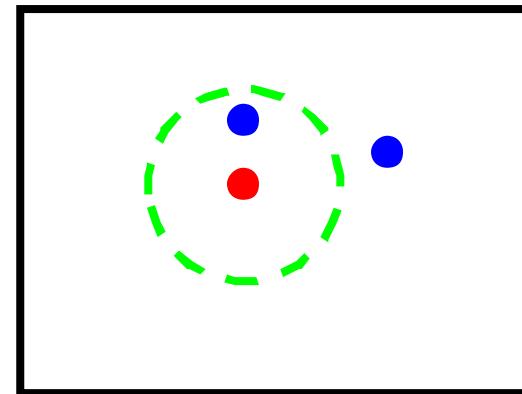
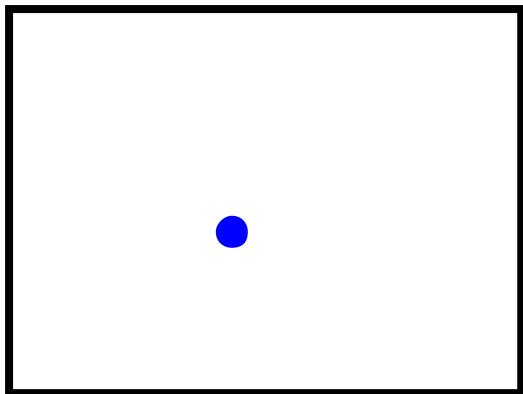


### Шаг 2: ассоциация обнаружений между кадрами для построения траекторий

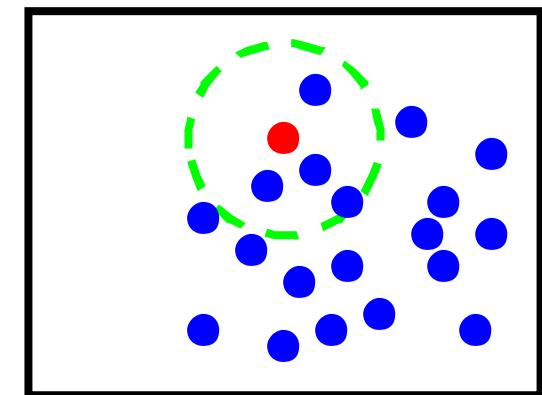
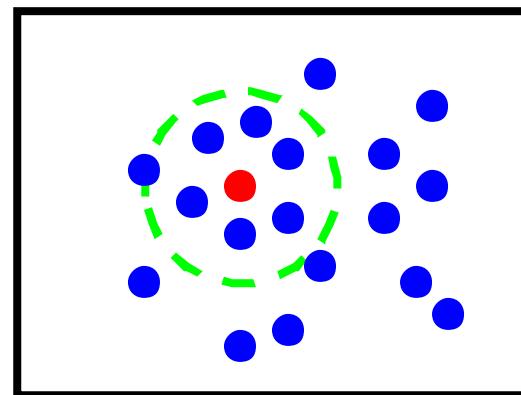
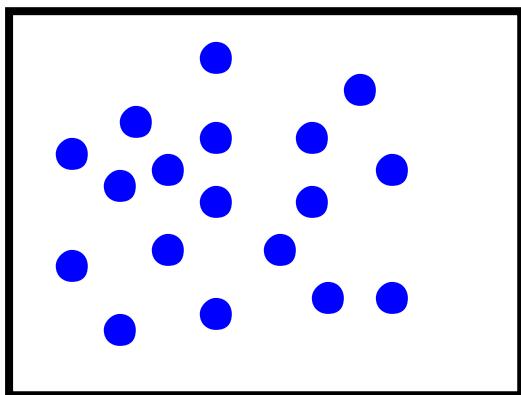
Какой простейший метод ассоциации обнаружений?



# Простейшая стратегия



Сопоставим ближайшее наблюдение следу



Простейшая стратегия в более сложных случаях не срабатывает



# Пример работы





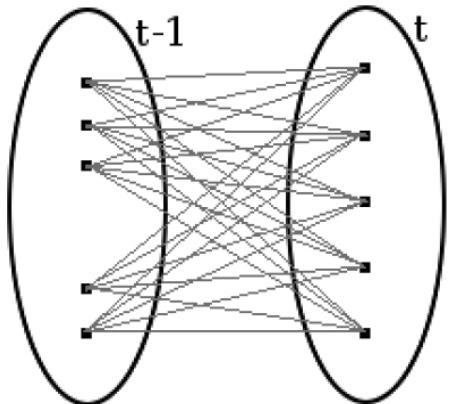
# Ассоциация обнаружений



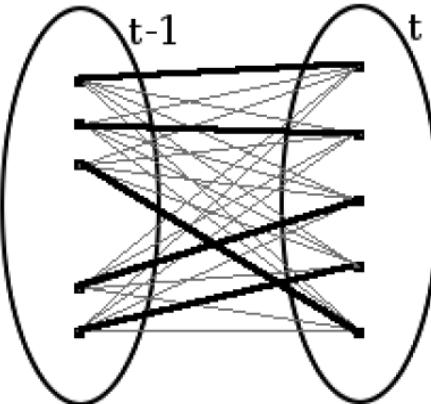
- Какие возможны ситуации при ассоциации обнаружений?
  - Сопоставление двух обнаружений на кадрах  $t-1$  и  $t$
  - Появление нового объекта в поле зрения
  - Пропадание объекта из поля зрения
  - Ложное обнаружение
  - Пропуск объекта (ошибка детектора)



# Алгоритмы ассоциации



(a)



(b)

Для случая 2x кадров  
имеем задачу о  
назначениях

- Для каждого возможного соответствия между кадрами  $t-1$  и  $t$  можем вычислить «стоимость» установления соответствия
- Получим матрицу стоимости назначений
- Можем решить Венгерским алгоритмом
- Какие факторы влияют на «стоимость»?



# SORT

---



- Алгоритм:
  - Нейросетевой детектор объектов
  - Фильтр Калмана для предсказания движения объекта
  - Критерий «стоимости» - IoU для предсказанного bbox и обнаружения
  - Венгерский алгоритм для сопоставления

A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft. [Simple Online and Realtime Tracking](#). In arXiv:1602.00763, 2016.



# Результаты SORT

**Table 2.** Performance of the proposed approach on MOT benchmark sequences [6].

Test Sequences		MOTA $\uparrow$	MOTP $\uparrow$	FAF $\downarrow$	MT $\uparrow$	ML $\downarrow$	FP $\downarrow$	FN $\downarrow$	ID sw $\downarrow$	Frag $\downarrow$
TBD [20]	Batch	15.9	70.9	2.6%	6.4%	47.9%	14943	34777	1939	1963
ALEXTRAC [5]	Batch	17.0	71.2	1.6%	3.9%	52.4%	9233	39933	1859	1872
DP_NMS [23]	Batch	14.5	70.8	2.3%	6.0%	40.8%	13171	34814	4537	3090
SMOT [1]	Batch	18.2	71.2	1.5%	2.8%	54.8%	8780	40310	1148	2132
NOMT [11]	Batch	<b>33.7</b>	71.9	<b>1.3%</b>	12.2%	44.0%	7762	32547	<b>442</b>	<b>823</b>
RMOT [4]	Online	18.6	69.6	2.2%	5.3%	53.3%	12473	36835	684	1282
TC_ODAL [17]	Online	15.1	70.5	2.2%	3.2%	55.8%	12970	38538	637	1716
TDAM [18]	Online	33.0	<b>72.8</b>	1.7%	<b>13.3%</b>	39.1%	10064	<b>30617</b>	464	1506
MDP [12]	Online	30.3	71.3	1.7%	13.0%	38.4%	9717	32422	680	1500
SORT (Proposed)	Online	<b>33.4</b>	72.1	<b>1.3%</b>	11.7%	<b>30.9%</b>	<b>7318</b>	32615	1001	1764



# Элементы обработки

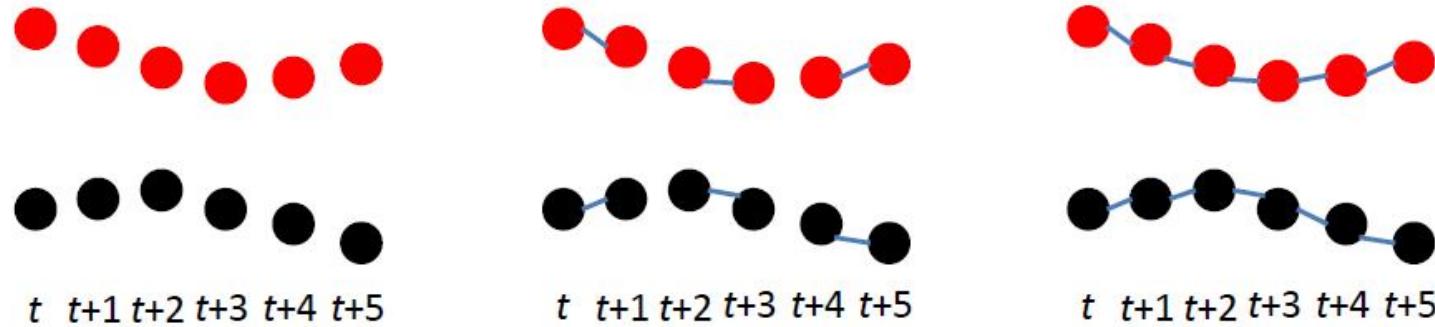


Fig. 2: Detection responses (left), tracklets (center), and trajectories (right) are shown in continuous 6 frames. Different colors encode different targets. Best viewed in color

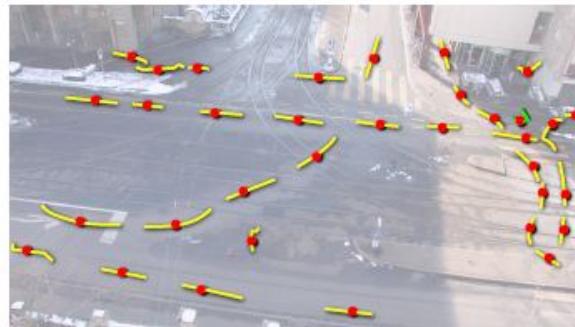
- «Треклет» - фрагмент траектории
- Можем построить треклет с помощью алгоритма визуального сопровождения
- Обычно – vot быстрее, детектор точнее
- Выход – ищем компромисс между скоростью и надежностью
  - Трекер можно использовать для предсказания движения объектов



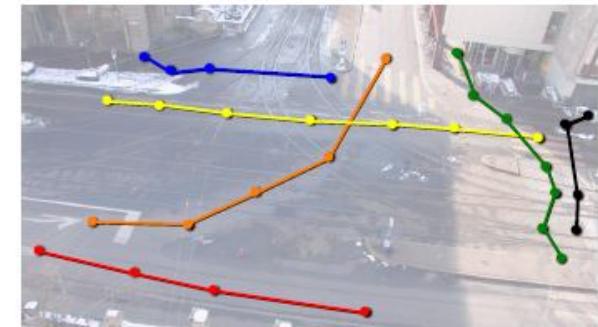
# Визуализация



(1) Object detection



(2) Single-target tracking

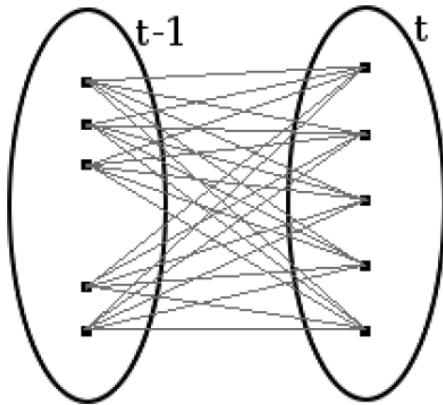


(3) Multi-target tracking

Figure 1: We (1) compute object detections, (2) from which we initialize a visual tracker to obtain short trajectories, which we use for (3) global data association.

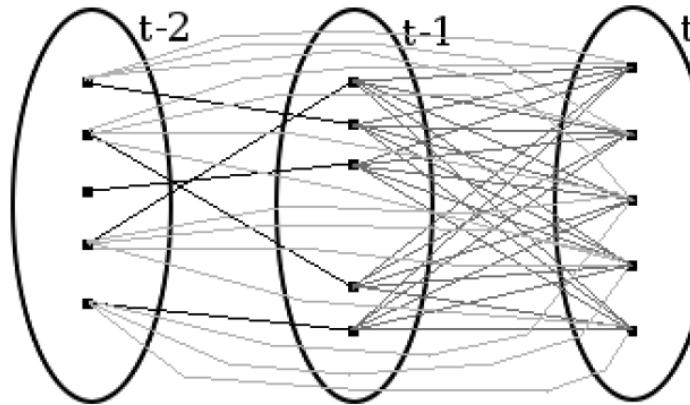


# Алгоритмы ассоциации



(a)

Двухкадровые



(c)

Многокадровые

Подходы к установлению соответствий:

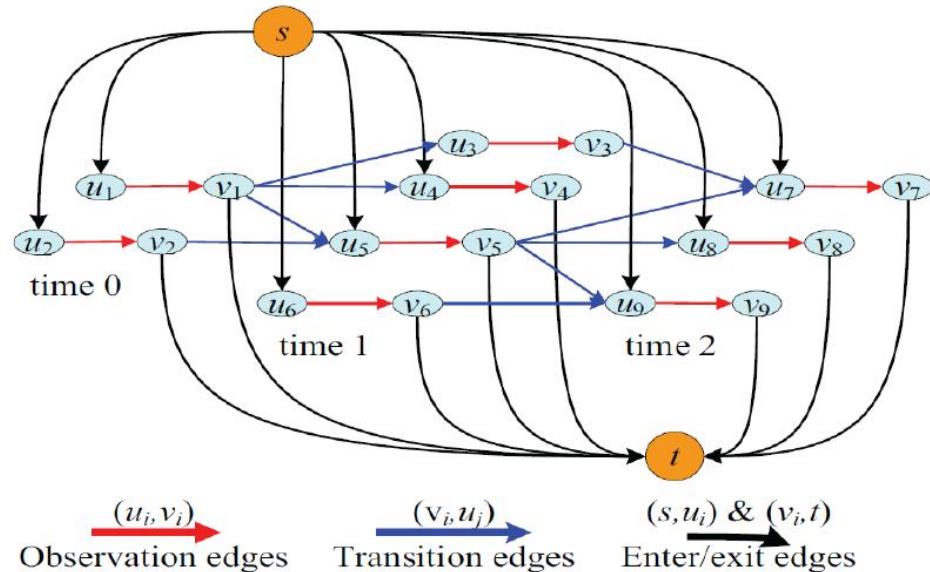
- Перебор разных вариантов
- Энергетическая (или статистическая) формулировка:
  - Для каждого ограничения сформулируем «штраф»
  - Составим функцию «энергии» сопоставления
  - Будем оптимизировать энергию с помощью какого-нибудь метода оптимизации



# Примеры подходов

$$\mathcal{T}^* = \operatorname{argmin}_{\mathcal{T}} \sum_i C_{\text{in}}(i) f_{\text{in}}(i) + \sum_i C_{\text{out}}(i) f_{\text{out}}(i) + \sum_i C_{\text{det}}(i) f(i) + \sum_{i \neq j} C_{\text{t}}(i, j) f(i, j)$$

Сведение к задаче  
линейного  
программирования





# Пример алгоритма<sup>1)</sup>



- Шаг 1. Поиск голов на ключевых кадрах
- Шаг 2. Построение треклетов
  - визуальное сопровождение (пр. – стая точек)
  - получаем гипотезы движения объектов между ключевыми кадрами (треклеты)
- Шаг 3. Объединение треклетов в траектории
  - Алгоритм МСМС DA
    - Построение выборки из распределения
    - Алгоритм Метрополиса – Гастингса
    - Элемент с максимальной вероятностью
- Шаг 4. Восстановление положения на промежуточных кадрах

<sup>1)</sup>Benfold B., Reid I. Stable Multi-Target Tracking in Real-Time Surveillance Video // Computer Vision and Pattern Recognition. 2011



# Визуализация

---





# Пример современного алгоритма



## **Stable Multi-Target Tracking in Real-Time Surveillance Video**

**CVPR 2011**

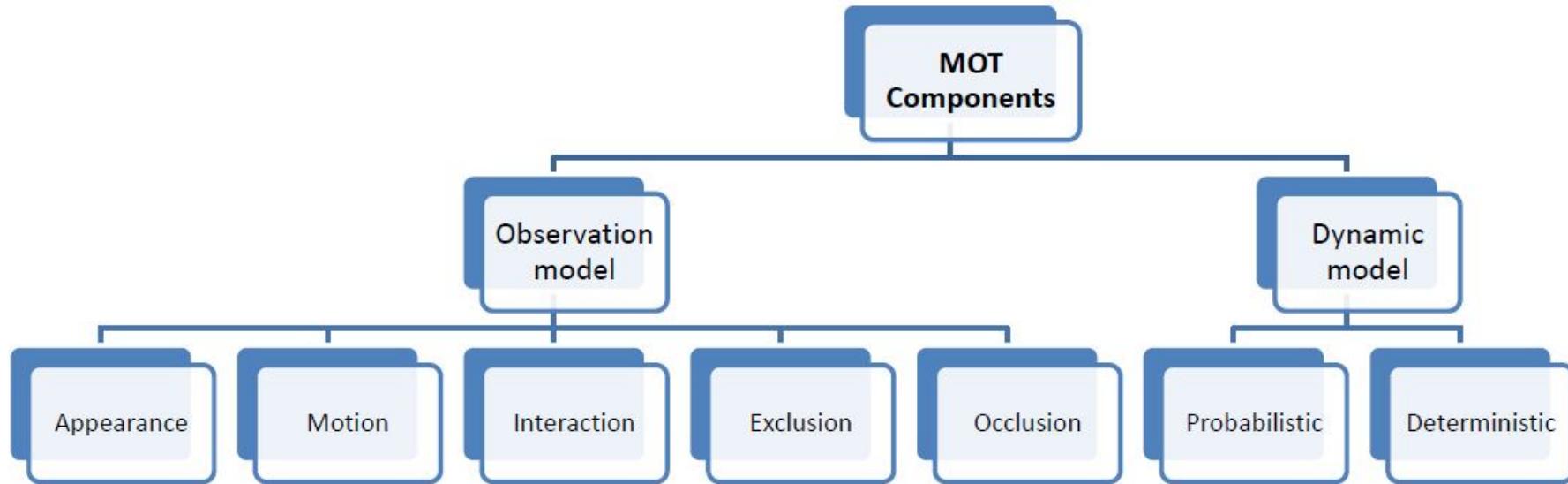
Ben Benfold and Ian Reid

Active Vision Group  
University of Oxford

B Benfold and I D Reid Stable Multi-Target Tracking in Real-Time  
Surveillance Video. CVPR 2011



# Компоненты MOT





# Примеры признаков

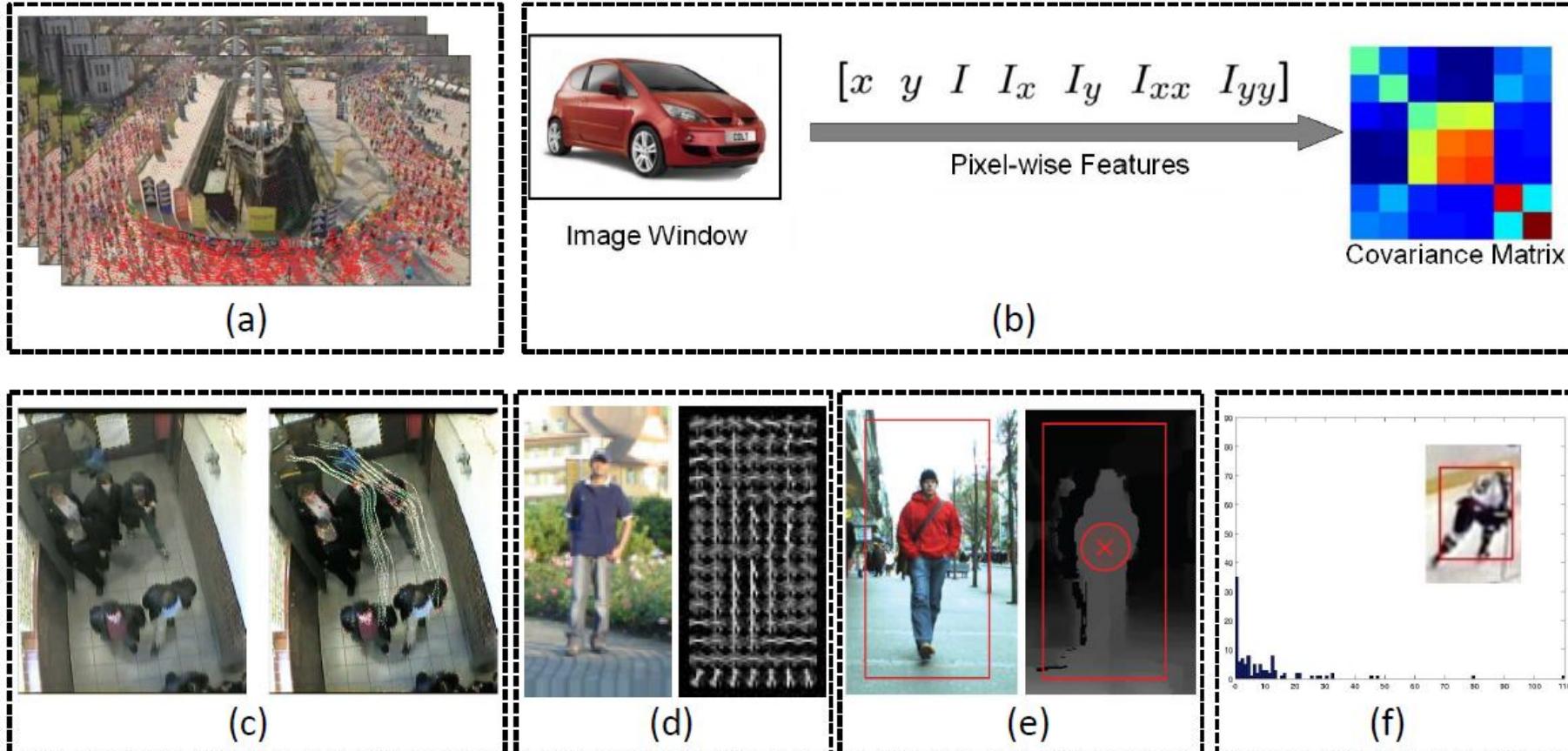


Fig. 6: Some exemplar features. (a) Image showing optical flow (Ali and Shah 2008), (b) image showing covariance matrix, (c) image showing point feature (Brostow and Cipolla 2006), (d) image showing gradient based features (Dalal and Triggs 2005), (e) image showing depth (Mitzel et al. 2010) and (f) image showing color feature (Okuma et al. 2004). Best viewed in color



# Appearance через CNN

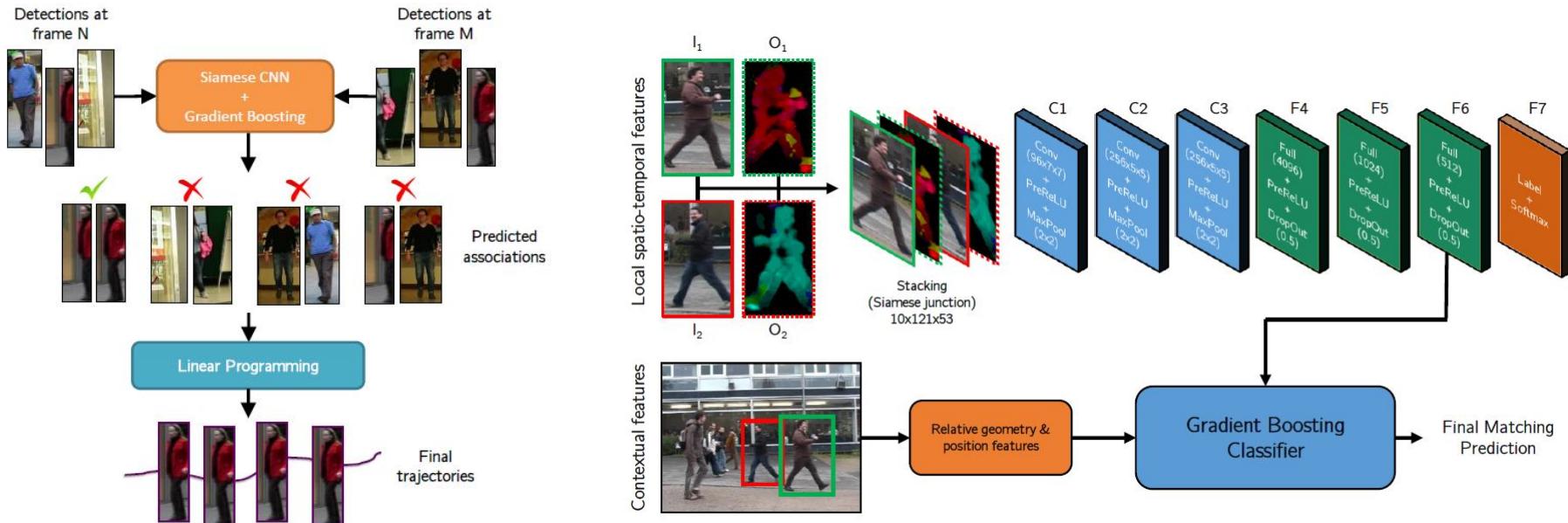
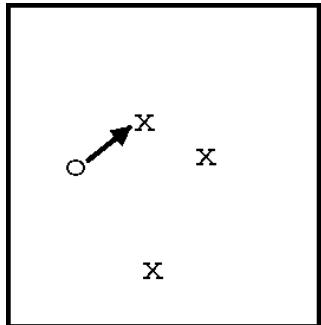


Figure 2: Proposed two-stage learning architecture for pedestrian detection matching.

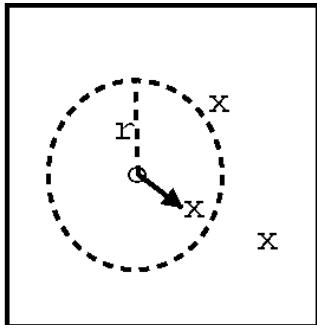
Laura Leal-Taixé, Cristian Canton-Ferrer, Konrad Schindler  
Learning by tracking: Siamese CNN for robust target association  
IEEE International Conference on Computer Vision and Pattern Recognition  
Workshops (CVPRW). Deep Vision: Deep Learning for Computer Vision. 2016



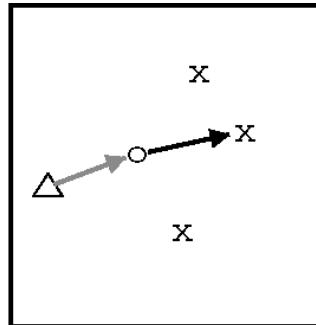
# Примеры кинематических факторов



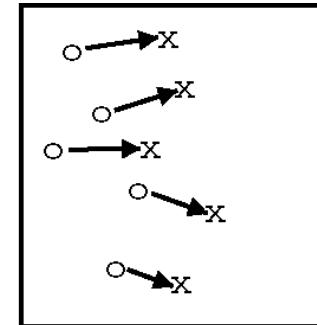
(a)



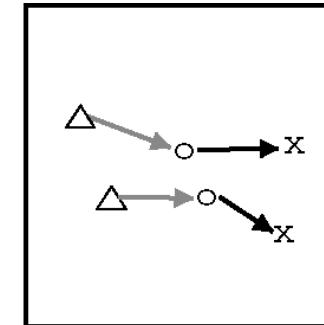
(b)



(c)



(d)



(e)

- (a) близость
- (b) максимальная скорость
- (c) малое изменение вектора скорости
- (d) общее движение
- (e) «жесткость»



# Модели движения

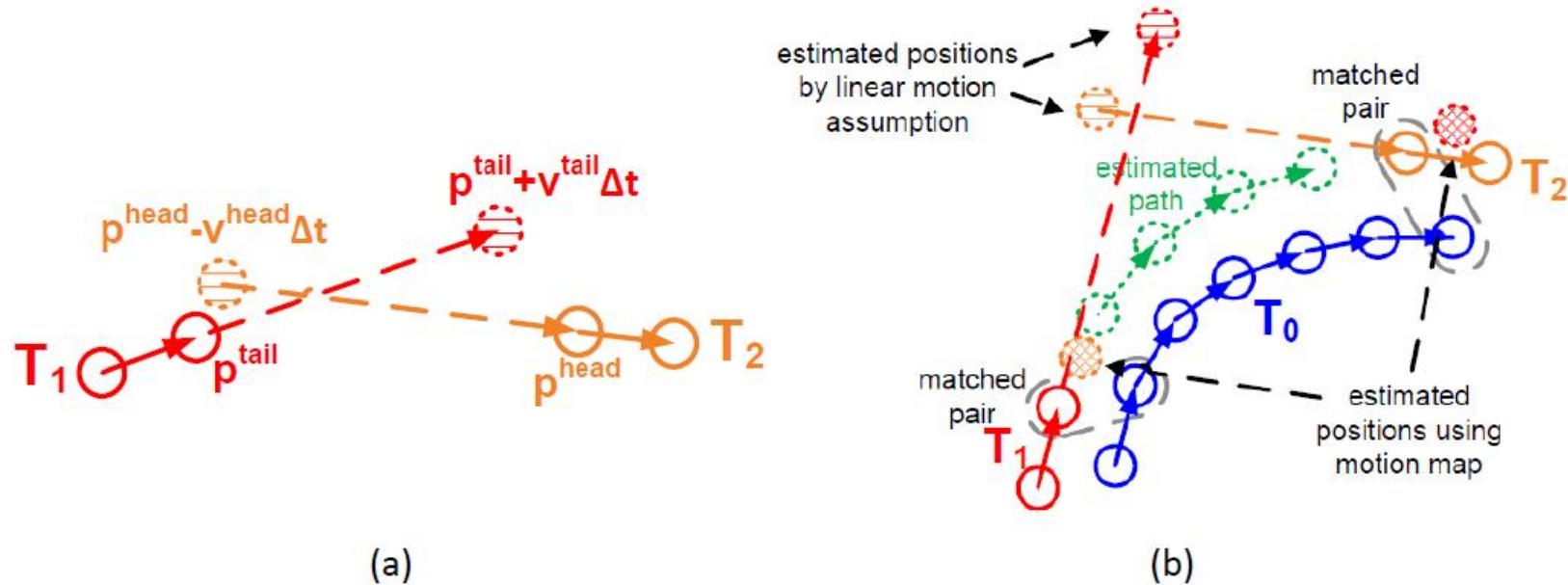


Fig. 8: An image comparing the linear motion model (a) with the non-linear motion model (b) (Yang and Nevatia 2012a). Best viewed in color



# Согласование траекторий

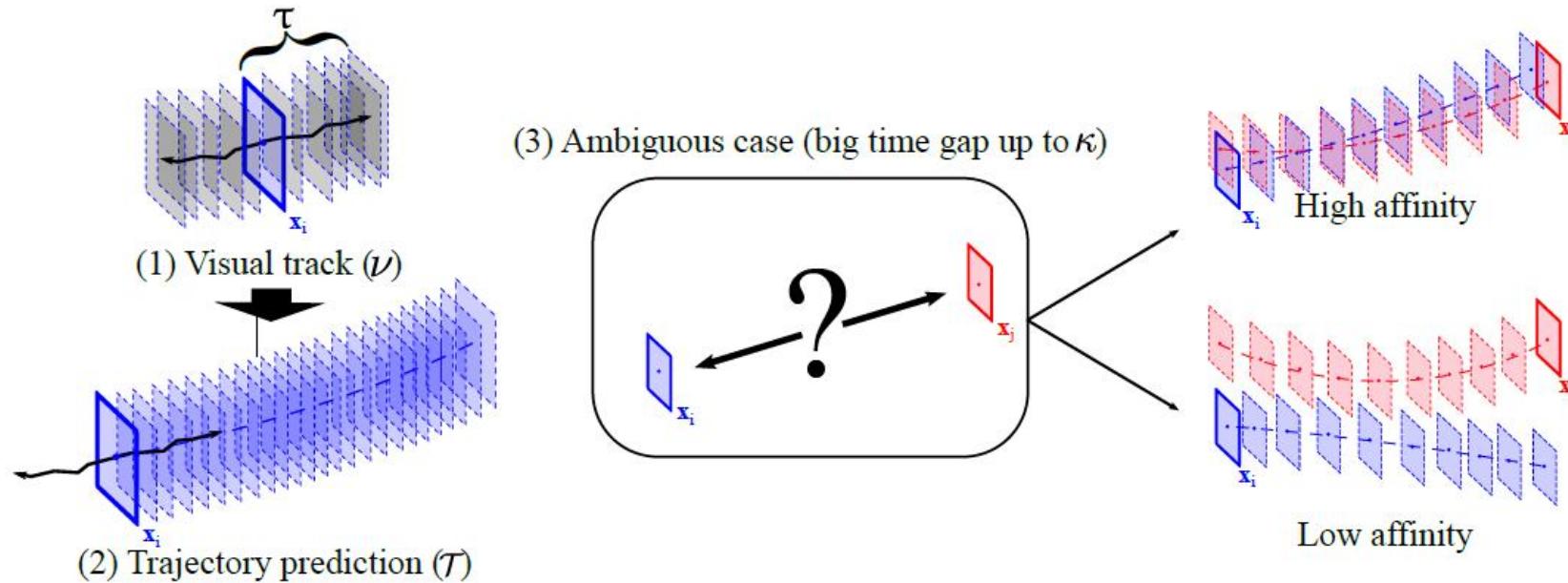


Figure 2: Illustration of the Trajectory Overlap (TO) affinity measure. For each detection we (1) run a visual tracker to obtain short trajectories ( $\nu$ ) that we use to (2) make trajectory predictions ( $\mathcal{T}_i$ ). (3) TO can handle ambiguous cases in which two detections are separated by a number of frames (gap), by measuring the overlap of the predicted trajectories.



# Резюме МОТ

---



- Основное подход – ассоциация обнаружений
- Варианты:
  - Online vs Offline
  - Обнаружения vs треклеты
  - Двухкадровая vs многокадровая ассоциация
  - Конкретные элементы
    - Детектор
    - Метод построения треклетов
    - Метод оптимизации