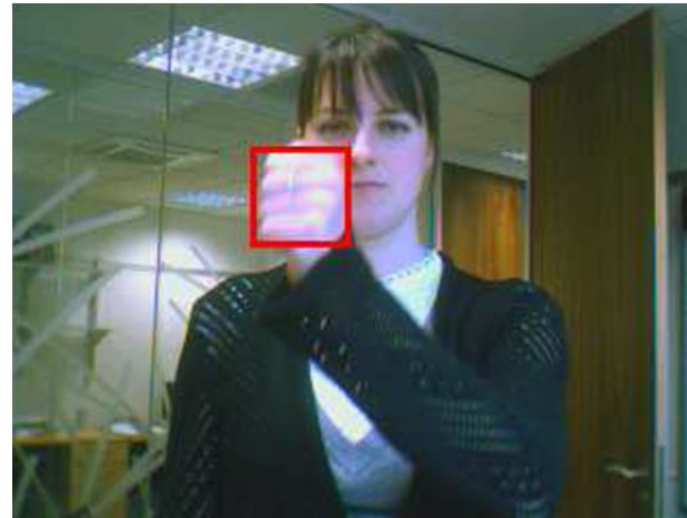
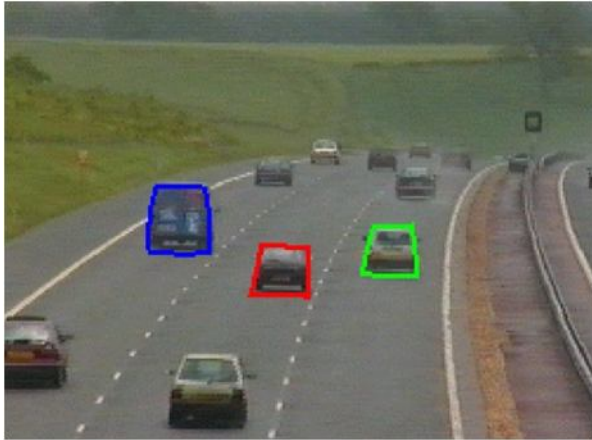




Основы видеонаблюдения





Общая информация

Microsoft
Research

Этот курс
подготовлен и
читается при
поддержке Microsoft
Research

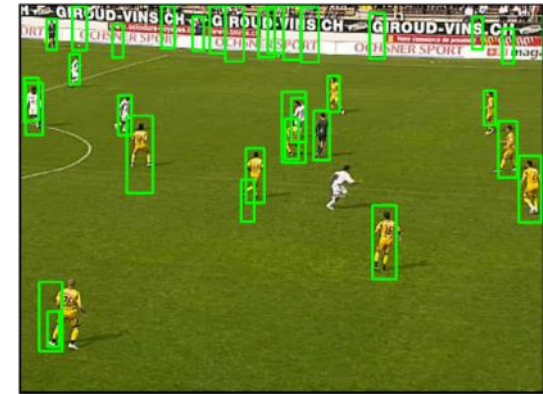
Microsoft
Research

- Страница курса
<http://courses.graphicon.ru/main/vision>



Задачи видеонаблюдения

- Выделение «объектов интереса» в видео и отслеживание их движения в последующих кадрах
- Выделение объектов интереса
 - Результат - ограничивающий прямоугольник
 - Или попиксельная маска
- Отслеживание (video tracking)
 - Вход – положение объекта на первом кадре
 - Результат – траектория движения объектов (след или “track”)





Проблемы

- Масштабируемость
 - Видео гораздо больше одного изображения, гораздо выше вычислительная нагрузка
- Изменение по времени
 - Вид объекта меняется от кадра к кадру из-за ракурса, изменения освещения, внутренних изменений (идуший человек)
- Несколько объектов
 - В сцене могут быть несколько объектов, которые могут быть похожи друг на друга, перекрывать друг друга и т.д.



Оценка качества

- Потребуется размеченные данные, в которых отмечено положение объекта в каждом (или в ключевых кадрах)

- Ошибка отслеживание – расстояние до ground truth данных

$$e_t^k = d(\hat{\mathbf{x}}_t^k, \mathbf{x}_t^{gt})$$

- Ожидаемая ошибка на видеопоследовательности

$$E[e^k] = \frac{1}{T} \sum_t e_t^k, k = 1, \dots, K.$$

- Точность (precision)

$$1 - E[e^k | e^k < \tau]$$

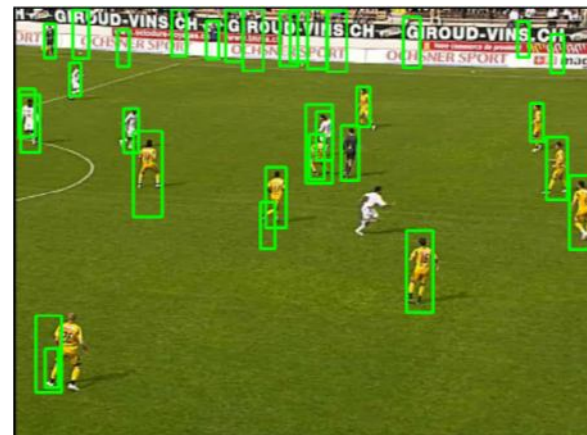
- Надежность (robustness)

$$p(e_t^k < \tau | e_{t-1}^k < \tau).$$



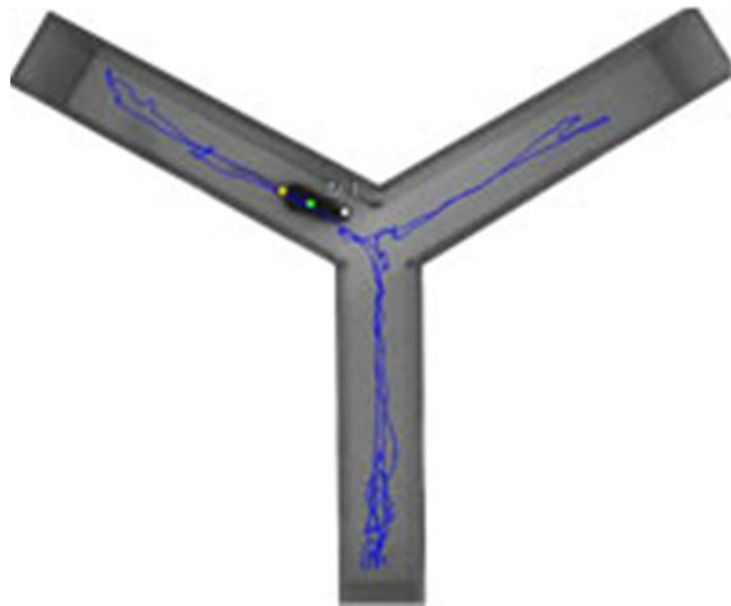
Выбор объекта интереса

- Инициализация вручную
- Детектор объектов
 - Детектор «пешехода»
 - Детектор «лиц»
- Сегментация видео
 - Выделение движущихся объектов





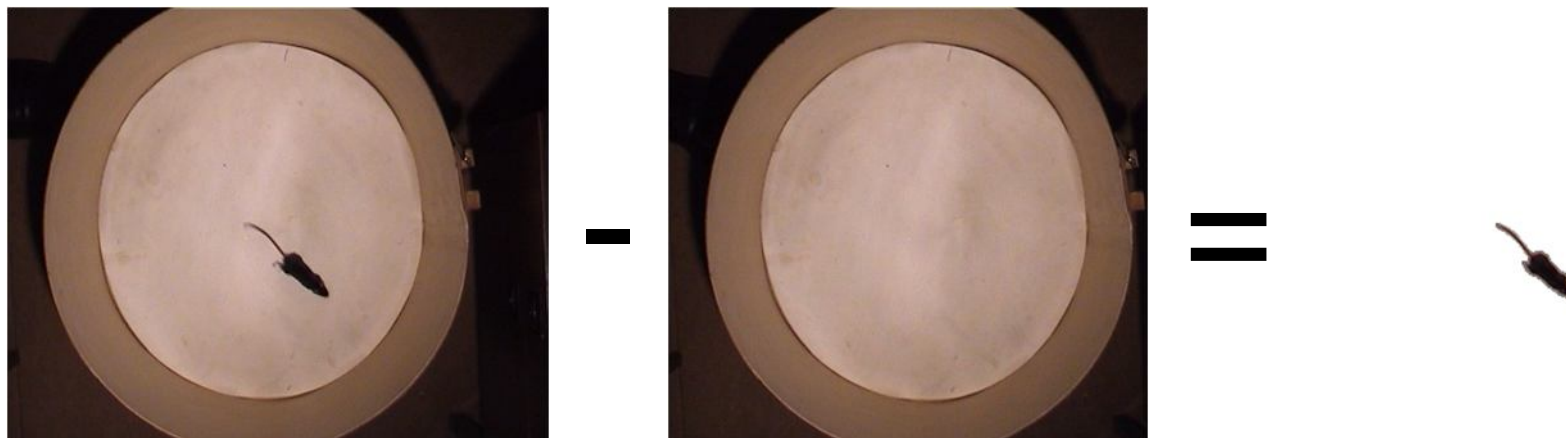
Система видеонаблюдения



- Построим стандартную, простую систему видеонаблюдения
- Упрощение 1: стационарная камера
- Упрощение 2: стабильный фон, мало изменяющийся между кадрами



Вычитание фона



- Background subtraction
 - Возьмем изображение без объектов (фон, *background*)
 - Вычтем фон из новых изображений с объектами
 - Сравним разницу для каждого пикселя с порогом
 - Порог – параметр алгоритма
 - Если разница больше порога - то пиксель принадлежит «переднему плану» (*foreground*)
 - Получаем маску «переднего плана»



Обработка маски переднего плана



- Выделим отдельные объекты как связанные компоненты
- «Блоб» («капля», *blob*) - связанная компоненты маски переднего плана
- Для подавления шума отфильтруем маску переднего плана
 - Медианная фильтрация, морфология
- Это простейший (и работоспособный) алгоритм выделения движущихся объектов
 - Для мышек часто применялся



Video capture





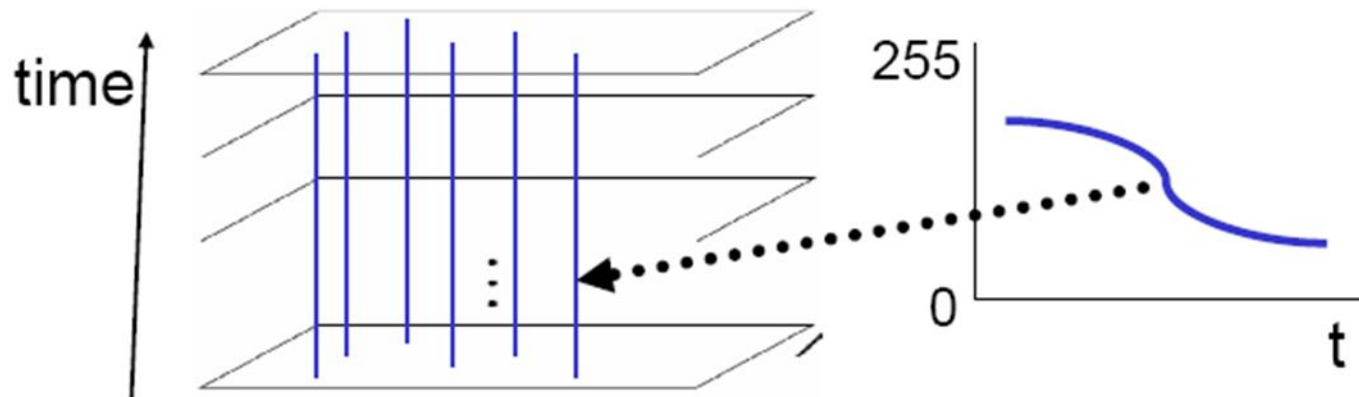
Реальная картина



Фон существенно меняется



Яркость по времени



- Будем рассматривать каждый пиксель изображения i независимо от остальных
- Яркость (цвет) x_i каждого пикселя фона изменяется по времени t
- Построим модель яркости(цвета) x_i пикселя i фона
- Если на новом кадре яркость (цвет) пикселя не удовлетворяет модели фона - значит это пиксель принадлежит движущемуся объекту



Общая схема вычитания фона

- *Initialize_background_model()*
- *For t = 1:N*
 - *Compute_frame_difference()*
 - *Threshold_frame_difference()*
 - *Noise_removal()*
 - *Update_background_model()*
- *end*



Усреднение кадров

- Простейшая модель фона – одно изображение чистого фона
- Возьмем N кадров и попиксельно усредним интенсивности (цвета)

$$I_0(\mathbf{x}, t) = \frac{1}{T} \sum_{t=1}^T I(\mathbf{x}, t)$$





Модель фона - медиана



- Когда не работает усреднение:
 - Всегда в кадре движущиеся объекты
 - Случайные и резкие изменения яркости (блики, засветка)
- Вместо усреднения берем медиану



Среднее vs Медиана



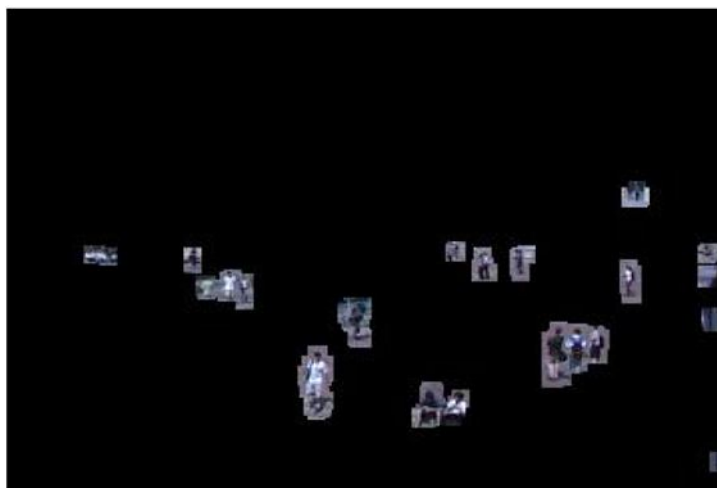
Средний цвет



Медиана



Вычитание медианного фона





Другие методы

- Во многих случаях медиана может не помогать
 - Пр.: когда мышь сидит в одном месте >50% кадров тестового видеоролика
- Предлагаются другие методы, на основе оптимизации целевой функции
 - Например, разобьём все изображения на сегменты
 - Будем выбирать сегменты из разных кадров, чтобы была наиболее плавная и стабильная картинка



Обновление модели фона

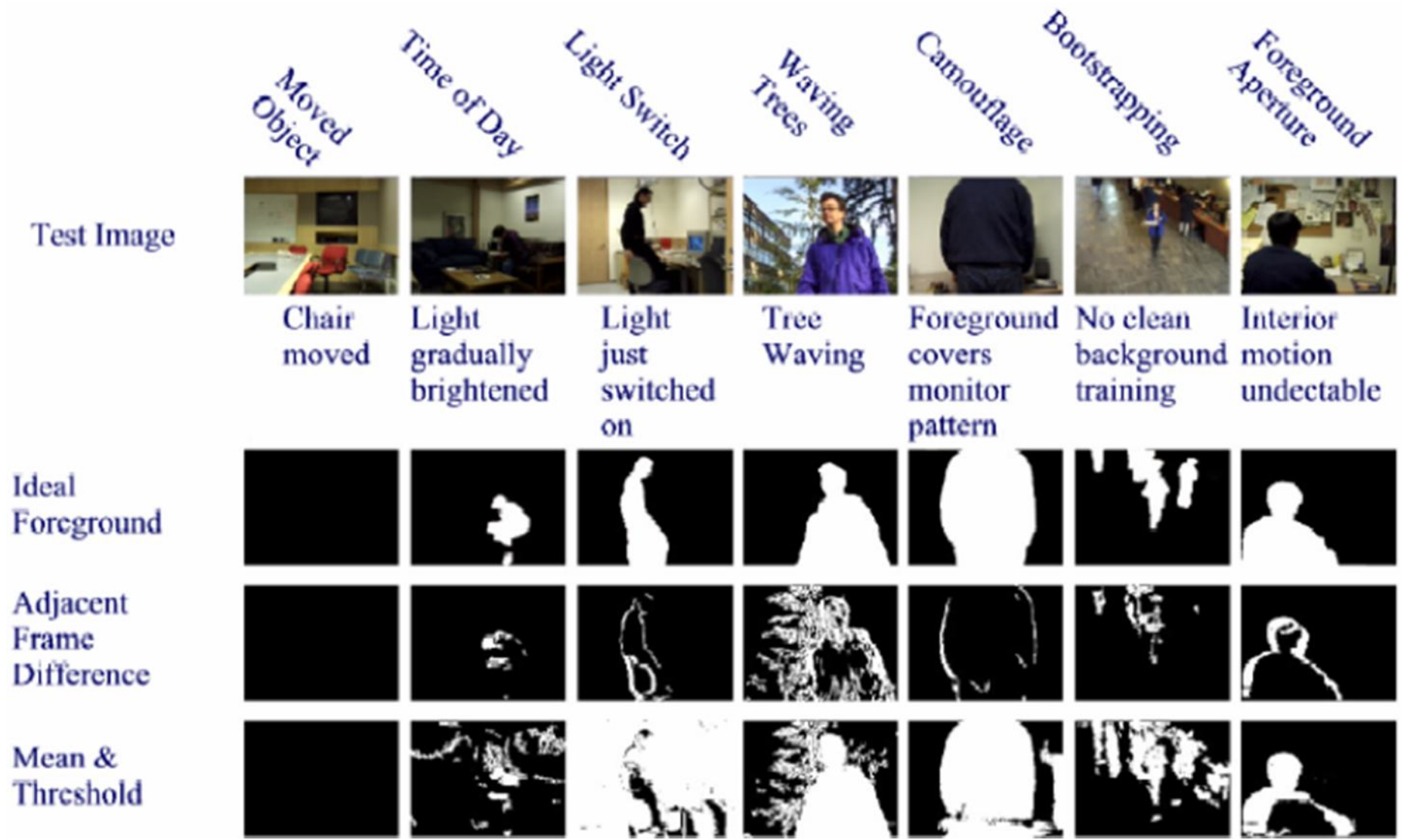
- Что делать, если освещенность постепенно меняется?
- Moving average – постоянное обновление фона
- Взвешенная сумма текущего и предыдущих N кадров

$$I_0(\mathbf{x}, t) = \frac{w_a I(\mathbf{x}, t) + \sum_{i=1}^N w_i I(\mathbf{x}, t - i)}{w_c}$$

- Ещё один (или несколько) параметр – вес кадра



Виды помехи

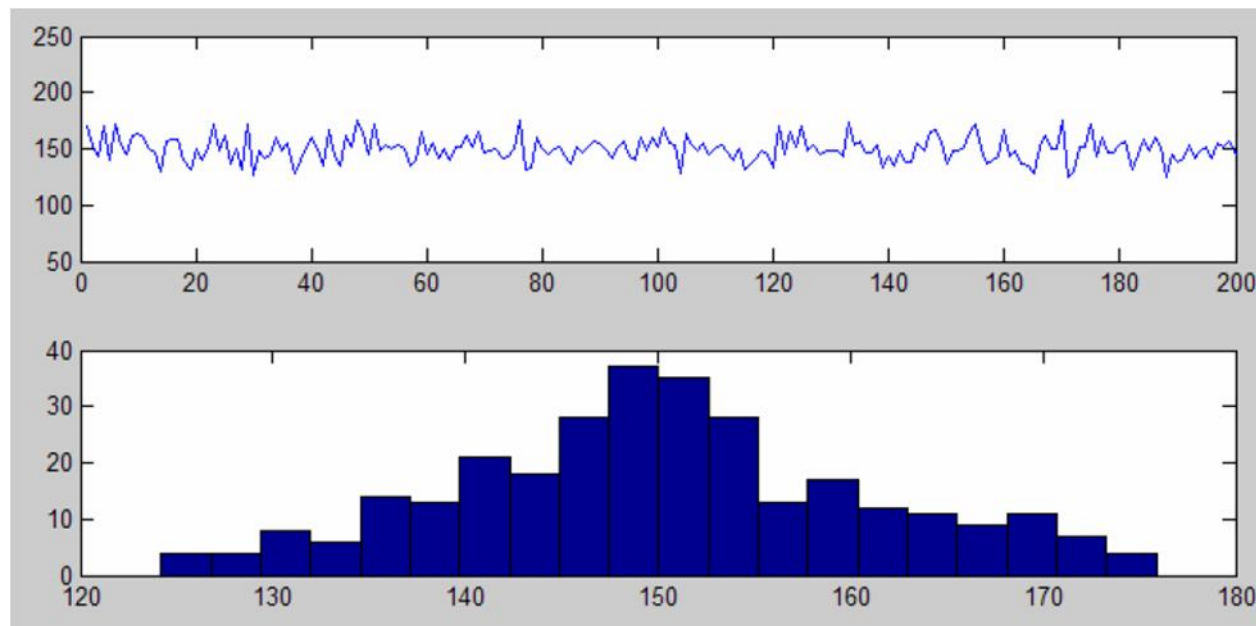


from K. Toyama et al.



Гауссиана

- Предположение:
 - Интенсивность меняется в небольших пределах вокруг какого-то значения
 - «Шум камеры»
 - Нормальное распределение



Wren, Christopher R., Ali Azarbayejani, Trevor Darrell, and Alex Pentland. "Pfinder: Real-Time Tracking of the Human Body," IEEE PAMI, 1997



Вычисление параметров

- Для одного канала:

- Вычисляем среднее

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i$$

- Вычисляем дисперсию

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2$$

- Вероятность нового пикселя y :

$$\rho(x | \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Обновление параметров

- Обновление матожидания

$$\mu_{t+1} = \alpha \mu_t + (1 - \alpha) x_{t+1}$$

- Обновление дисперсии

$$\sigma_{t+1}^2 = \alpha (\sigma_{t+1}^2 + (\mu_{t+1} - \mu_t)^2) + (1 - \alpha) (x_{t+1} - \mu_{t+1})^2$$

α – скорость обновления (обучения)



Многоканальное изображение

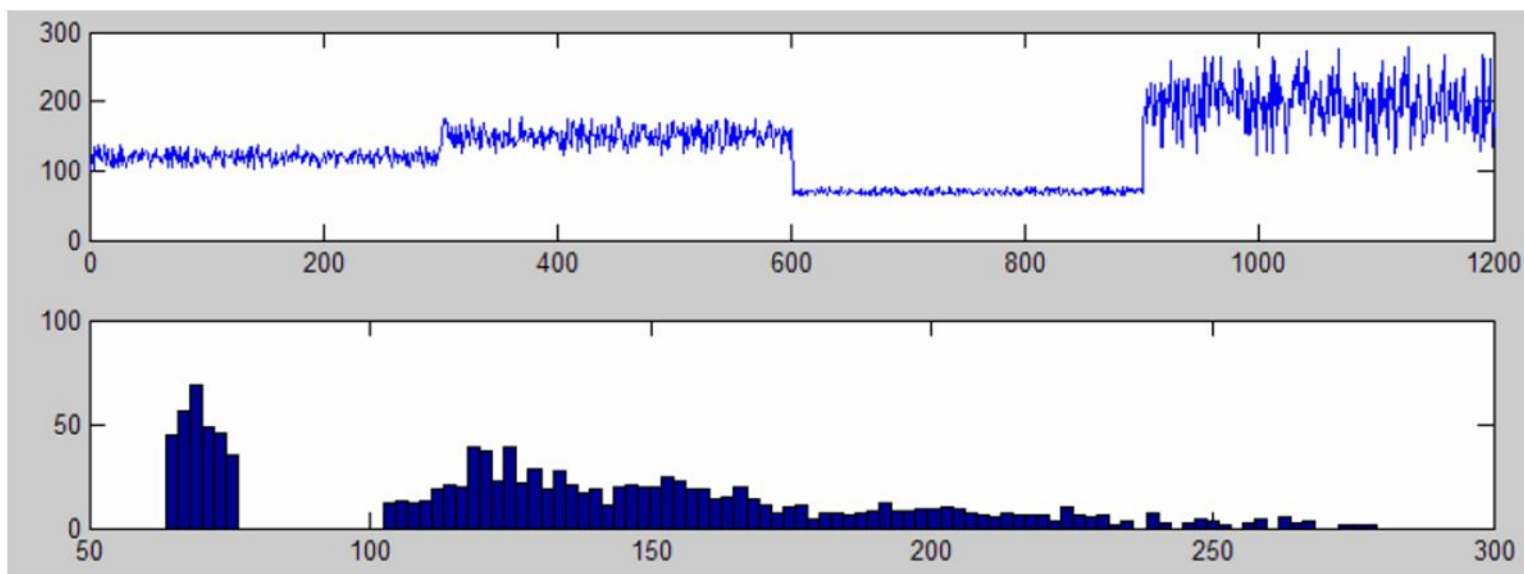
- Можно рассчитать полную матрицу ковариации K
- Обновлять полную матрицу сложно
- Обновление только среднего, с сохранением ковариации

$$\mu_{t+1} = \alpha \mu_t + (1 - \alpha) x_{t+1}$$



Смесь гауссиан

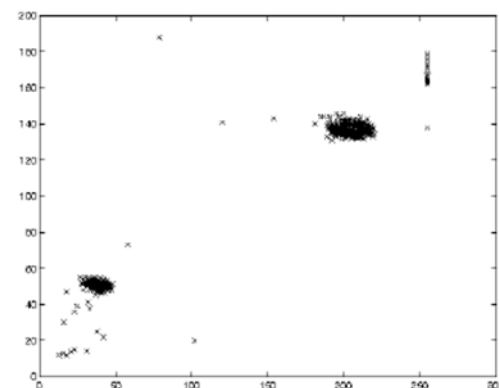
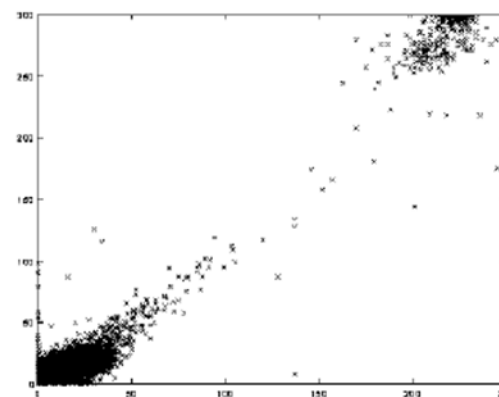
- Один гауссиан – один кластер
- Значения могут группироваться в несколько кластеров



W.E.L.Gimson, C.Stauffer. Adaptive background mixture models for real-time tracking. CVPR 1999



Смесь гауссиан



Распределение интенсивности красного
и зеленого каналов



Смесь гауссиан

- Вероятность интенсивности y при модели смеси K гауссиан

$$P(x_t) = \sum_{i=1}^K \omega_{i,t} N(x_t, \mu_{i,t}, \sigma_{i,t})$$

$\omega_{i,t}$ - вес компоненты i в момент t

- Вычисление смеси требует EM алгоритма (не в реальном времени), поэтому используем приближение



Обучение смеси на лету

- Пусть N – количество компонент в смеси
 - Инициализируем 1ую компоненту по первому изображению, вес = 1, вес остальных – 0
 - Сравниваем пиксель с каждой компонентой, пока не найдем совпадения
 - Обновляем матожидание и дисперсию совпавшей компоненты
 - Если совпадения не найдено, то заменяем компоненту с наименьшим весом
 - Обновляем веса



Обновление весов

Обновление весов:

$$\omega_{i,t} = (1 - a)\omega_{i,t-1} + aM_{i,t}$$

$M_{i,t} = 1$, если интенсивность пикселя
удовлетворяет i -ой компоненте

После обновления всех весов, они нормализуются



Моделирование фона

- Упорядочим все компоненты по критерию ω_i / σ
 - Чем «стабильнее» компонента, тем выше
 - Чем больше вес (чаще встречается, тем тоже выше)
- Определим порог T - какая доля выборки для каждого пикселя должна соответствовать фону
- Тогда фон для каждого пикселя:

$$B = \arg \min_b \left(\sum_{i=1}^b \omega_i > T \right)$$

- Все остальные компоненты - объекты



Пример работы



(1)

(2)

(3)

(4)

1. Исходный кадр
2. Модель фона (старшая гауссиана)
3. Мат.ожидания второй гауссианы
4. Маска переднего плана



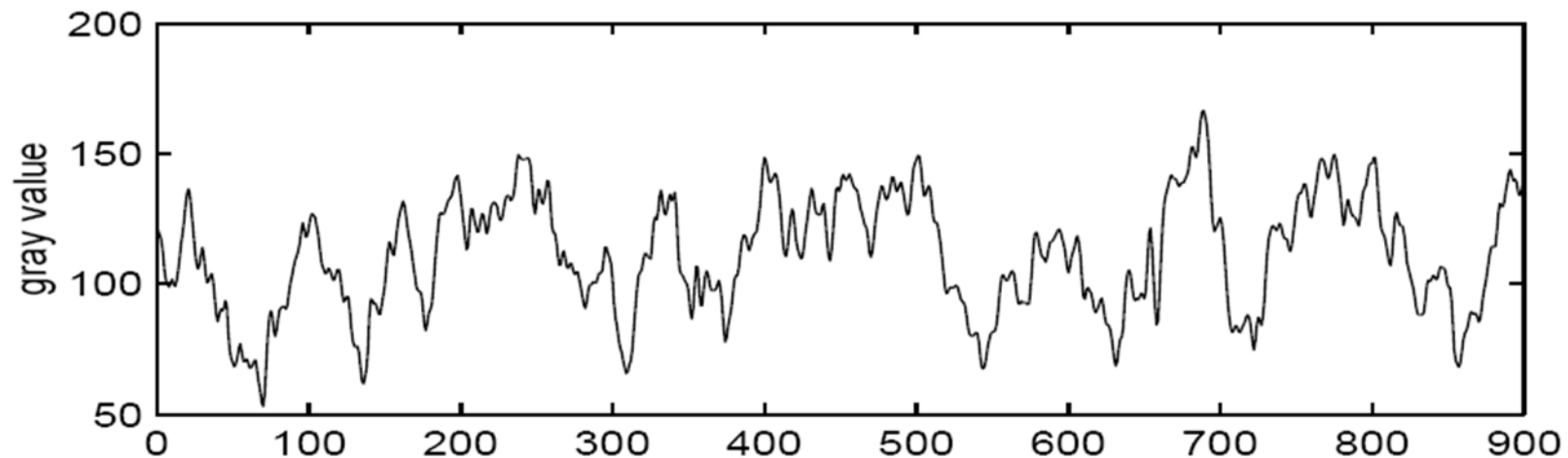
Как настраивать модель?

- Если есть размеченные данные:
 - Оценка precision/recall на наборе видеороликов
 - Выбор параметров, оптимизирующих желаемую точность
- Если нет размеченных данных
 - Можно взять видео без объектов
 - Настроить параметры таким образом, чтобы получить заданное количество ложных обнаружений
 - Минимальные пороги для достижения заданной «чувствительности»





Непараметрическая модель



A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction, " ICCV'99 Frame-rate workshop, 1999



Непараметрическая модель

Пусть x_1, \dots, x_N - N значений пикселя (выборка)

$$P(x) = \frac{1}{N} \sum_{j=1}^N K(x - x_j)$$

K – ядровая функция, обычно тоже гауссиан

$$Pr(x_t) = \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^d \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{1}{2} \frac{(x_{tj} - x_{ij})^2}{\sigma_j^2}} \quad \Sigma = \begin{pmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{pmatrix}$$

$P(x) > T$ - тогда фон



Непараметрическая модель

- К можем считать с помощью lookup-таблицы
- Можно суммировать до тех пор, пока не превысим порог (и пометим как объект)
- Оценка ширины ядра:

$$m = \mathit{median}|x_i - x_{i+1}|$$

$$\sigma = \frac{m}{0.68\sqrt{2}}$$



Двухэтапное обнаружение

- Объект может случайно сдвинуться дальше, чем в обученной модели
- Образуются ложные обнаружения
- Второй этап обнаружения:
 - Сравним пиксель со всеми из окрестности:

$$P_{\mathcal{N}}(x_t) = \max_{y \in \mathcal{N}(x)} Pr(x_t | B_y)$$

- Будем разрешать, только если вся компонента сдвинулась:

$$P_C = \prod_{x \in \mathcal{C}} P_{\mathcal{N}}(x)$$



Схема обновления фона

- Short-term model
 - N последних элементов
 - Выборочное обновление
- Long-term model
 - N элементов за долгий период времени
 - Слепое обновление
 - Больше false positives (т.к. модель не учитывает последних изменений)
 - Больше false negatives (т.к. в модель слепо можем добавить объекты)
- Комбинированная модель
 - Находим объекты на обеих моделях
 - Берём их пересечение
 - Добавляем объекты из «срочной» модели, если они граничат с найденным на «долгой» модели объектами



Пример работы





Поблочная обработка



- Вместо анализа каждого пикселя мы можем разбить всё изображение на блоки и анализировать блоки
- Точность сегментации получается хуже, зато считается статистика изображения в блоке, что повышает надежность

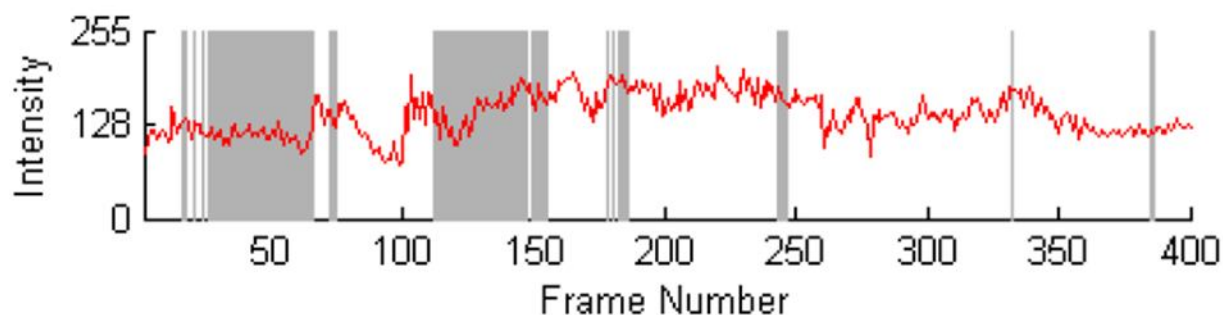


Статистики по окрестностям

Feeder Station Webcam data set



Intensity value of a pixel over time



[Ko, T.\[Teresa\]](#), [Soatto, S.\[Stefano\]](#), [Estrin, D.\[Deborah\]](#), Background Subtraction on Distributions, *ECCV 2008*



Схема алгоритма

- Посчитаем статистику (гистограмму) по 3D окрестности пикселя (фон):

$$p_{ij}(x) = \frac{1}{|S|} \sum_{s \in S} \delta(s - x)$$

$$S = \{x_t(a, b) \mid |a - i| < c, |b - j| < c, 0 \leq t < T\}$$

- Посчитаем статистику по 2д окрестности пикселя на текущем кадре:

$$q_{ij,\tau}(x) = \frac{1}{|S_\tau|} \sum_{s \in S_\tau} \delta(s - x)$$

$$S_\tau = \{x_\tau(a, b) \mid |a - i| < c, |b - j| < c\}$$

- Вычислим разницу:

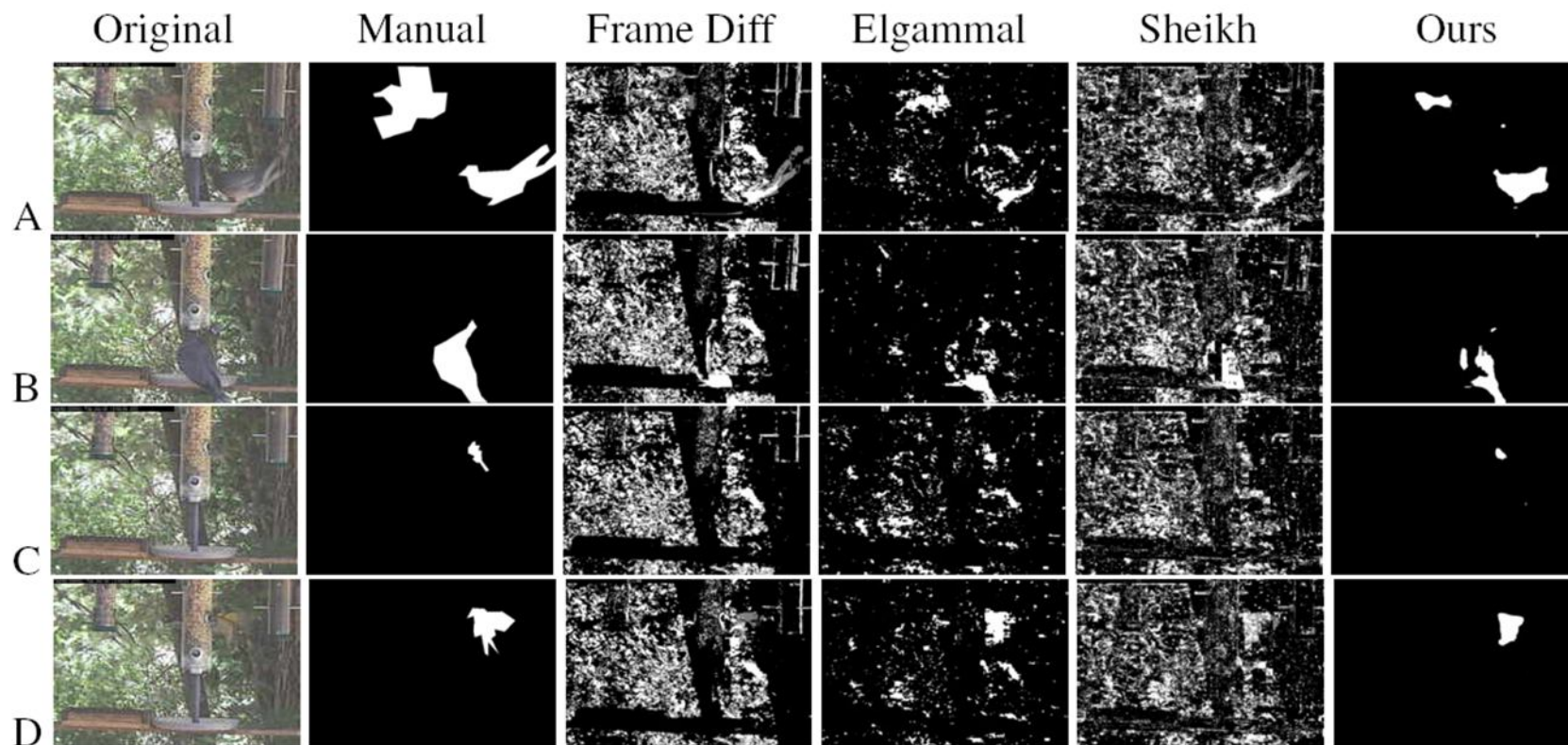
$$d = \int_X \sqrt{p_{ij,\tau-1}(x)q_{ij,\tau}(x)} dx$$

- Обновление модели:

$$p_{ij,t}(x) = (1 - \alpha)p_{ij,\tau-1}(x) + \alpha q_{ij,\tau}(x)$$



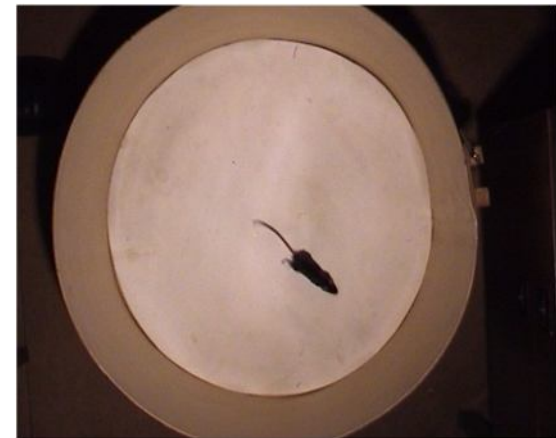
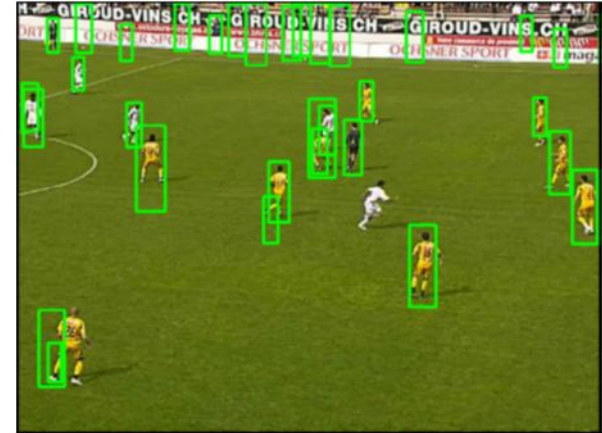
Результаты работы





Глобальные модели цветов

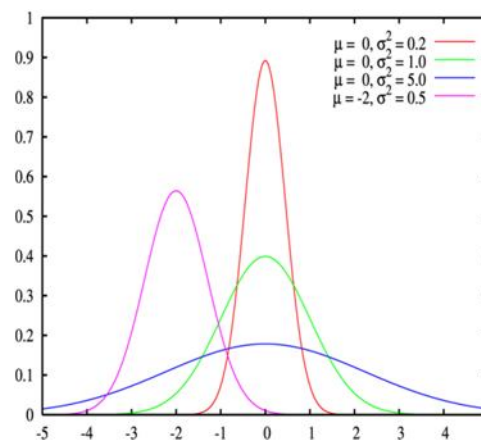
- Если у объекта и фона характерные цвета, то почему бы это не использовать?
- Построим цветовые модель объекта и модел фона
- Для каждого пикселя будем сравнивать вероятности принадлежности пикселя обоим моделям и выбирать ту, вероятность для которой будет выше





Модель фона

- Построим и глобальную, и локальную модели фона
- Глобальная: смесь гауссиан (GMM)



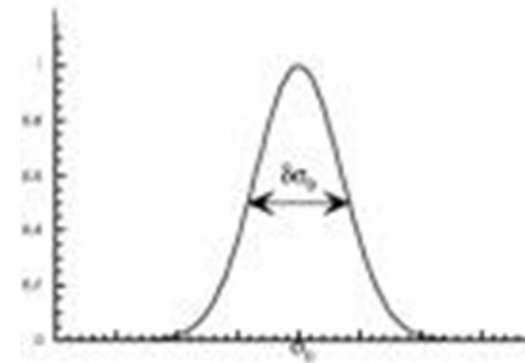
($K_b = 10-15$)

$$p(I_r | x_r = B) = \sum_{k=1}^{K_b} \omega_k^b N(I_r | \mu_k^b, \Sigma_k^b)$$



Модель фона

- Построим и глобальную, и локальную модели фона
- Глобальная: смесь гауссиан (GMM) ($K_b = 10 - 15$)
- Локальная: один гауссиан





Общая модель фона

- Глобальная: смесь гауссиан (GMM) ($K_b = 10 - 15$)

$$p_{global}(I_r | x_r = B) = \sum_{k=1}^{K_b} \omega_k^b N(I_r | \mu_k^b, \Sigma_k^b)$$

- Локальная: один гауссиан

$$p_{local}(I_r) = N(I_r | \mu_r^B, \Sigma_r^B)$$

- Объединенная:

$$p_{mix}(I_r) = \alpha \cdot p_{global}(I_r | x_r = B) + (1 - \alpha) \cdot p_{local}(x_r)$$

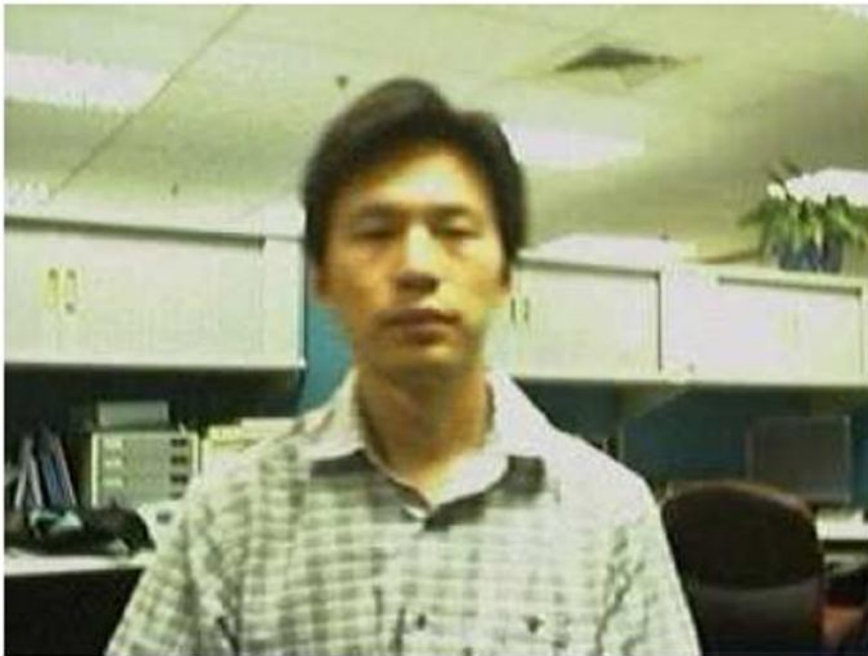


Модель переднего плана

- Возьмём те пиксели, для которых вероятность фона низкая

$$p_B(I_r) < t_f$$

- Обучим по ним смесь гауссиан (K=5)





Соотношение моделей?

$$p_{global}(I_r | x_r = F) \quad \text{if } x_r = F$$
$$p_{mix}(I_r) \quad \text{if } x_r = B$$

$$p_{mix}(I_r) = \alpha \cdot p_{global}(I_r | x_r = B) + (1 - \alpha) \cdot p_{local}(x_r)$$

$$\alpha = ?$$



Адаптивная смесь



$$p_{mix}(I_r) = \alpha \cdot p_{global}(I_r | x_r = B) + (1 - \alpha) \cdot p_{local}(x_r)$$



Адаптивная модель

Дивергенция Kullback-Liebler KL_{fb}

$$KL_{fb} = \sum_{k=0}^K w_k^f \min_i (KL(N_k^f \parallel N_i^b)) + \log \frac{w_k^f}{w_i^b}$$

дает разницу между двумя смесями гауссин

$$0 \leq KL_{fb} < \infty$$

$$KL_{fb} = 0 \iff \text{модели идентичные}$$



Адаптивная модель



$$\alpha = 1 - \frac{1}{2} e^{-\frac{KL_{fb}}{\sigma_{KL}}}$$



$$KL_{fb} \gg 1$$
$$\alpha \approx 1$$

$$KL_{fb} \ll 1$$

$$\alpha \approx 0.5$$

$$p_{mix}(I_r) \rightarrow p_{global}(I_r | x_r = B)$$

Только глобальная модель

$$p_{mix}(I_r) \rightarrow \frac{1}{2} \cdot p_{global}(I_r | x_r = B) + \frac{1}{2} \cdot p_{local}(x_r)$$

Поровну локальная и глобальные модели



Адаптивная модель

$$p_{global}(I_r | x_r = F) \quad \text{if } x_r = F$$
$$p_{mix}(I_r) \quad \text{if } x_r = B$$

$$p_{mix}(\alpha)(I_r) = \alpha \cdot p_{global}(I_r | x_r = B) + (1 - \alpha) \cdot p_{local}(x_r)$$

- Параметр смешения моделей можно уточнять на каждом кадре, в т.ч. с учётом других факторов
- Глобальная модель позволит нам обрабатывать сложные случаи изменения фона, когда локальная модель срабатывает плохо и наоборот



Поддержка модели фона

Изменение освещенности

- Автоподстройка камеры
- Лампы дневного света
- Включение света



Небольшое изменение освещенности:

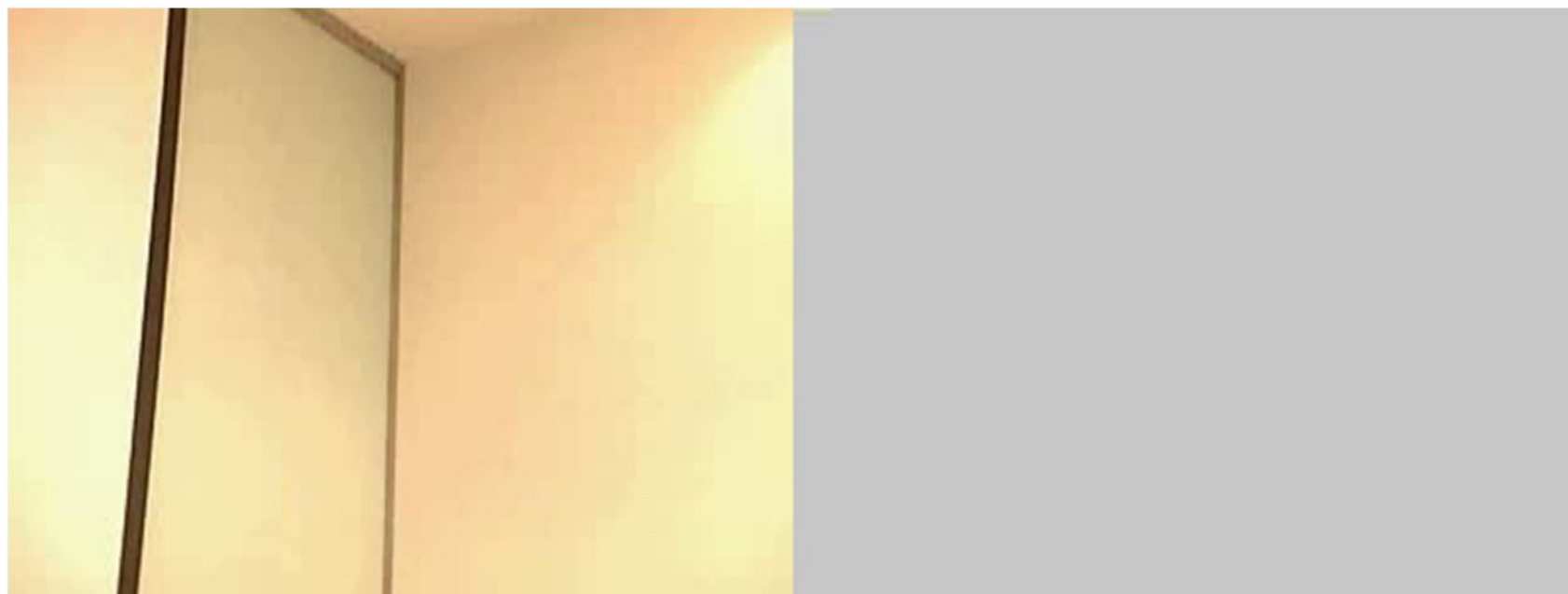
- Преобразование цветовой модели на основе преобразования гистограмм $\{I_r^B\} \leftrightarrow \{I_r \in B\}$

Резкие изменения

- Перестройка цветовой модели фона и переднего плана
- Использование карты достоверности краёв как подсказки



Поддержка модели фона





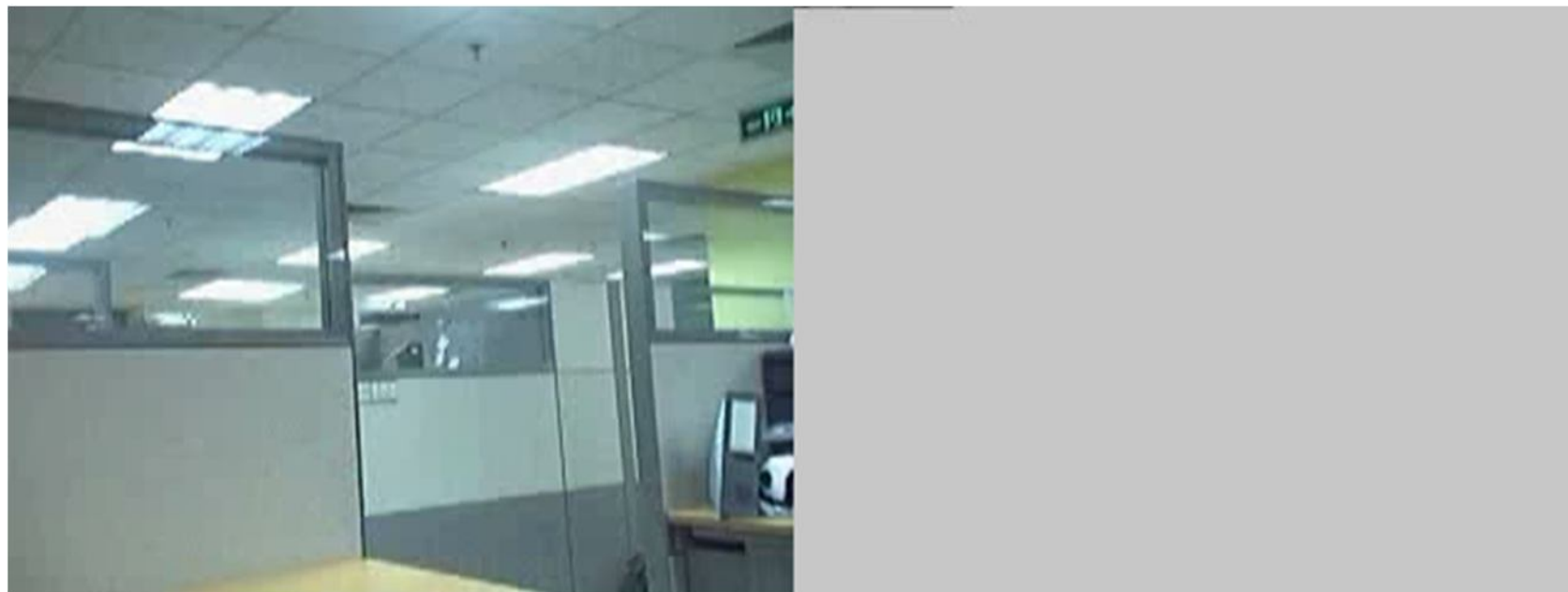
Поддержка модели фона

- Движение в фоне
 - Опираемся на глобальную модель
- Дрожание камеры
 - Применяем размытие по гауссу
 - Уменьшаем вес локальной цветовой модели
- «Заснувшие» и «Пешеход» на заднем плане
 - Оставляем только самую большую связанную компоненту





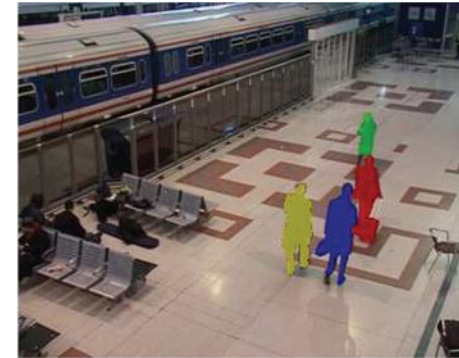
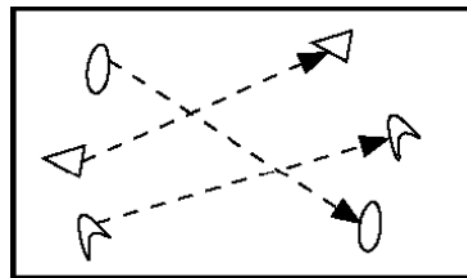
Поддержка модели фона





Отслеживание объектов

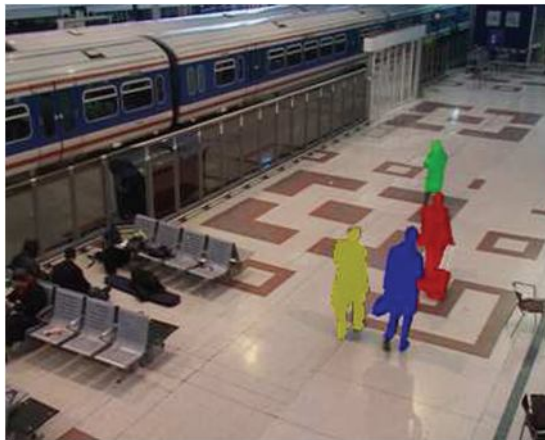
- Выделили «объекты интереса» на каждом кадре с помощью вычитания фона в виде набора блобов
- Какому объекту (следу) какой блоб соответствует?
- Фактически – задача сопоставления или ассоциации данных (Data association)
- «Tracking by detection»
- Можем каждый объект описать набором признаков





Детерминированное слежение

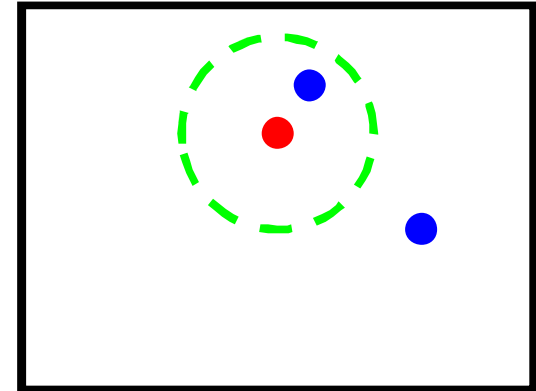
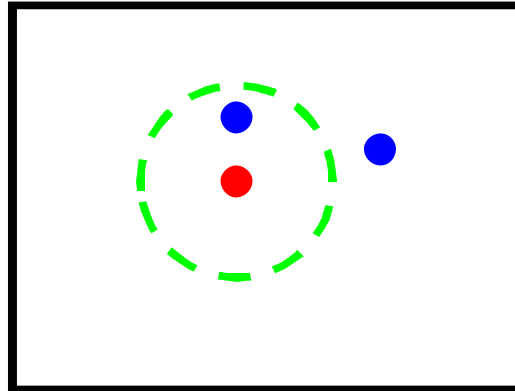
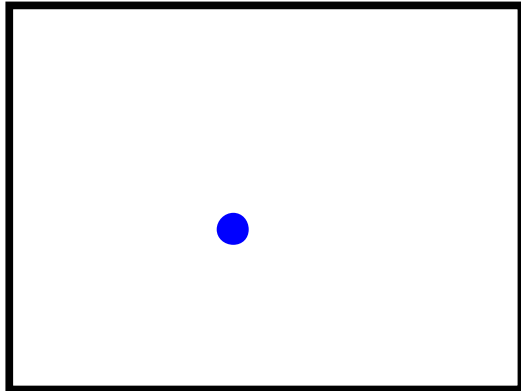
- Нахождение соответствий между объектами, найденными на соседних кадрах с учетом ограничений



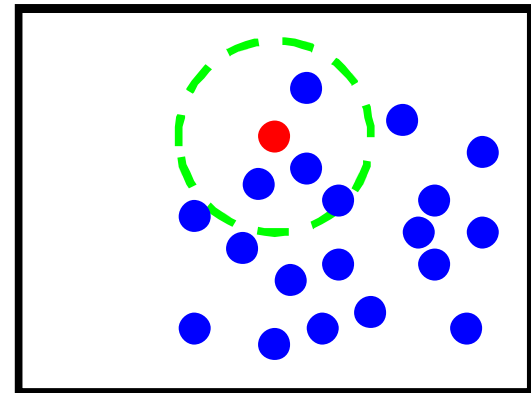
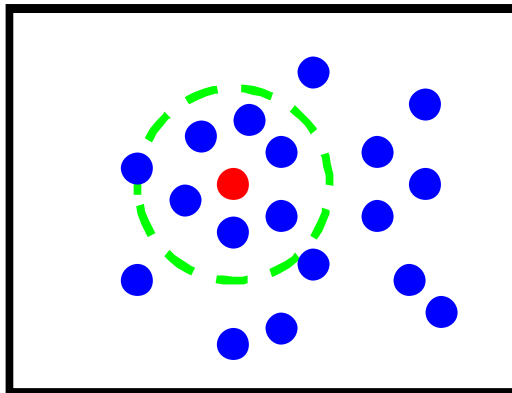
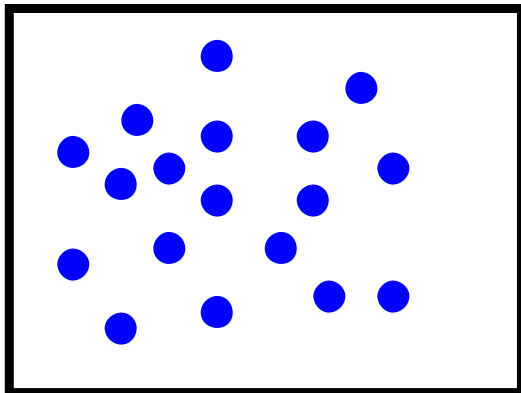
- Варианты:
 - Новые объект
 - Сопоставление старому следу
 - Пропадание объекта



Простейшая стратегия



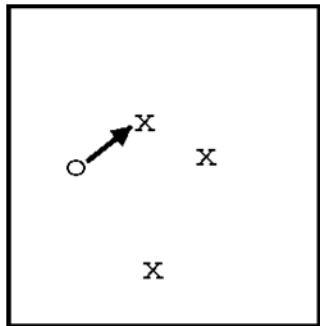
Сопоставим ближайшее наблюдение следу



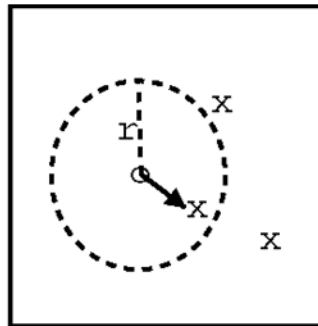
Простейшая стратегия в более сложных случаях не срабатывает, поэтому исследовались разные ограничения



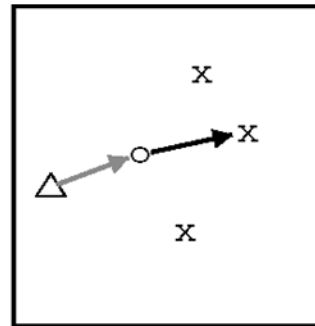
Виды ограничений



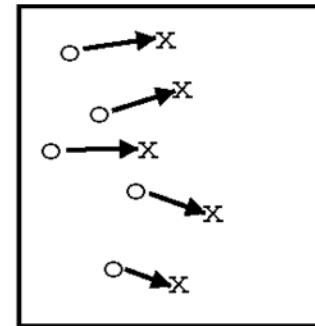
(a)



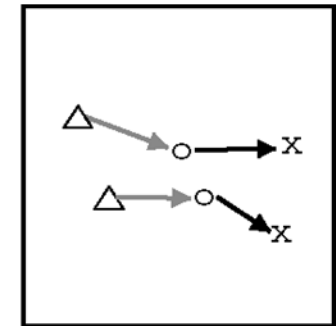
(b)



(c)



(d)

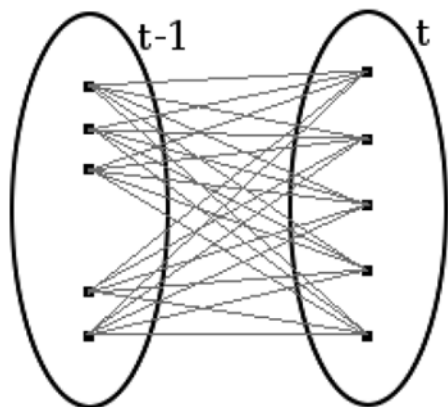


(e)

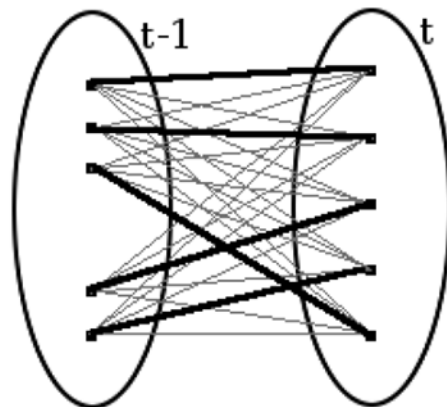
- (a) близость
- (b) максимальная скорость
- (c) малое изменение вектора скорости
- (d) общее движение
- (e) «жесткость»



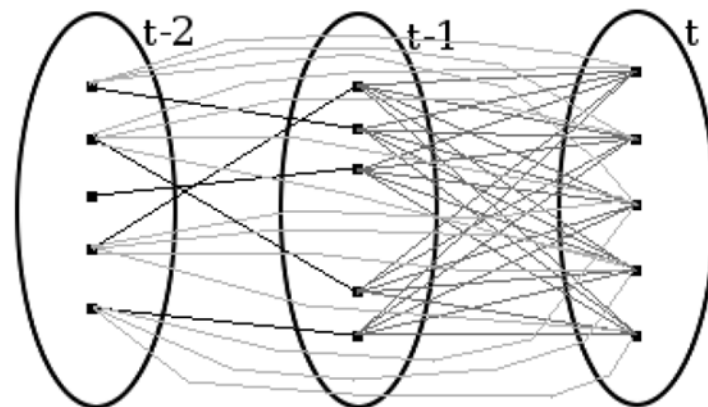
Детерминированные алгоритмы



(a)



(b)



(c)

Двухкадровые

Многокадровые

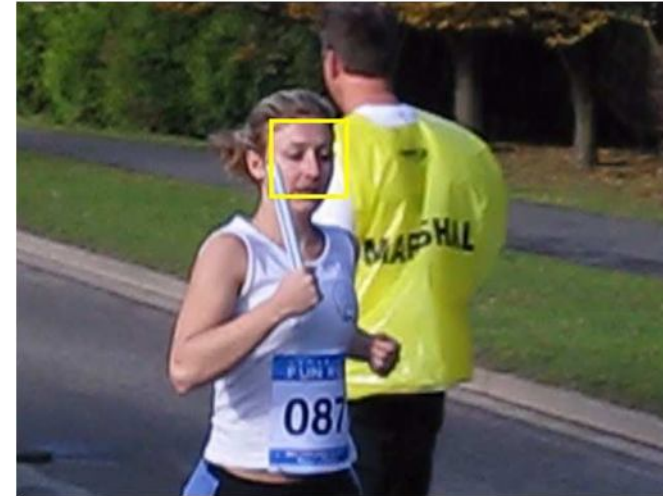
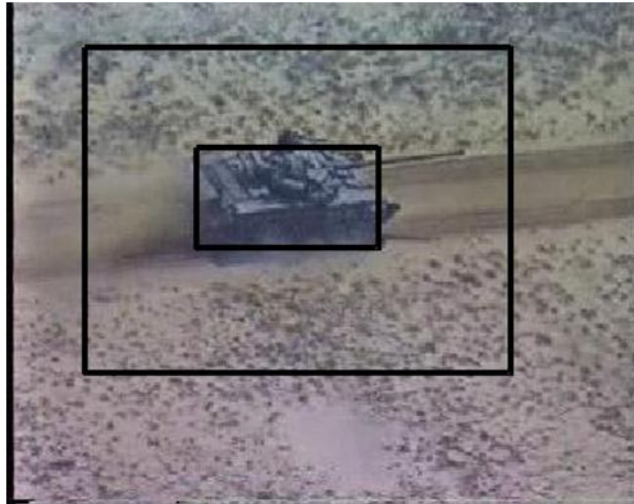


Базовый метод видеонаблюдения

- Обучение модели фона
- Для каждого кадра
 - Вычитание фона
 - Обработка маски (фильтрация, морфологии)
 - Выделение связанных компонент
 - Ассоциация объектов в треки
 - Инициализация новых треков
 - Обновление фона



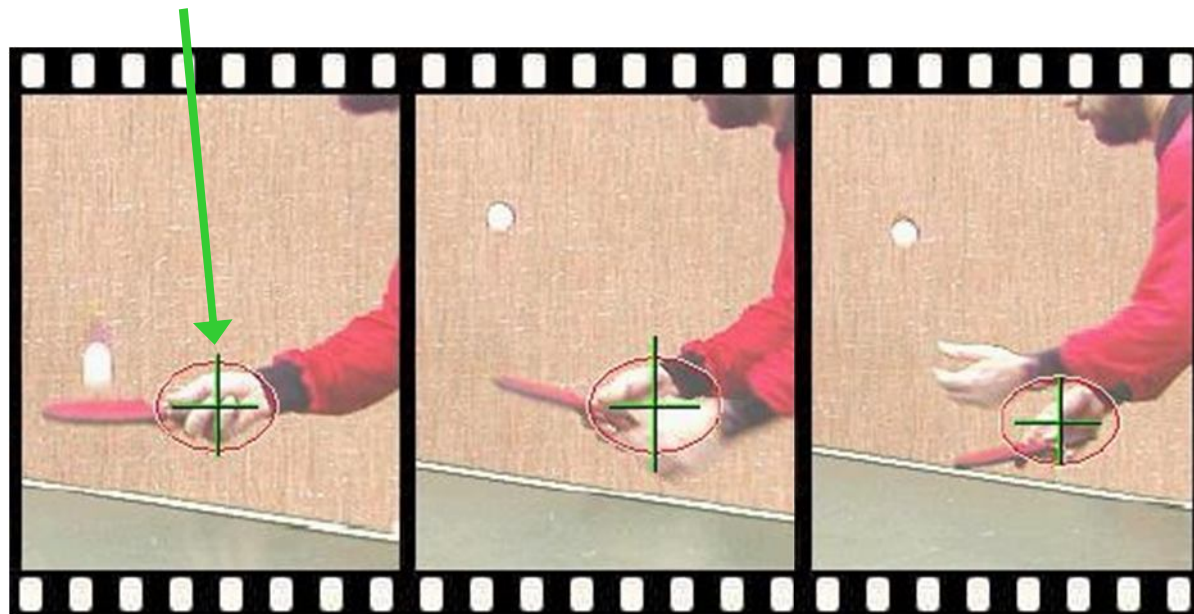
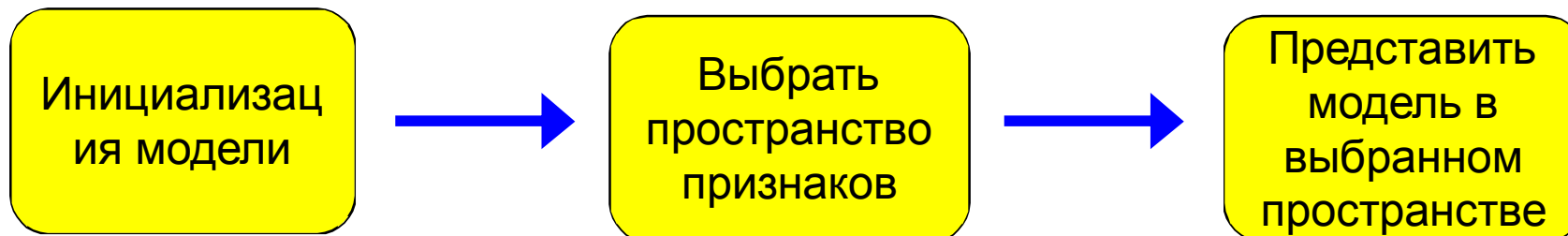
Отслеживание объектов



- Что делать, если фон не стационарный?
- Придётся строить какую-то модель объекта, и искать на следующем кадре похожий объект



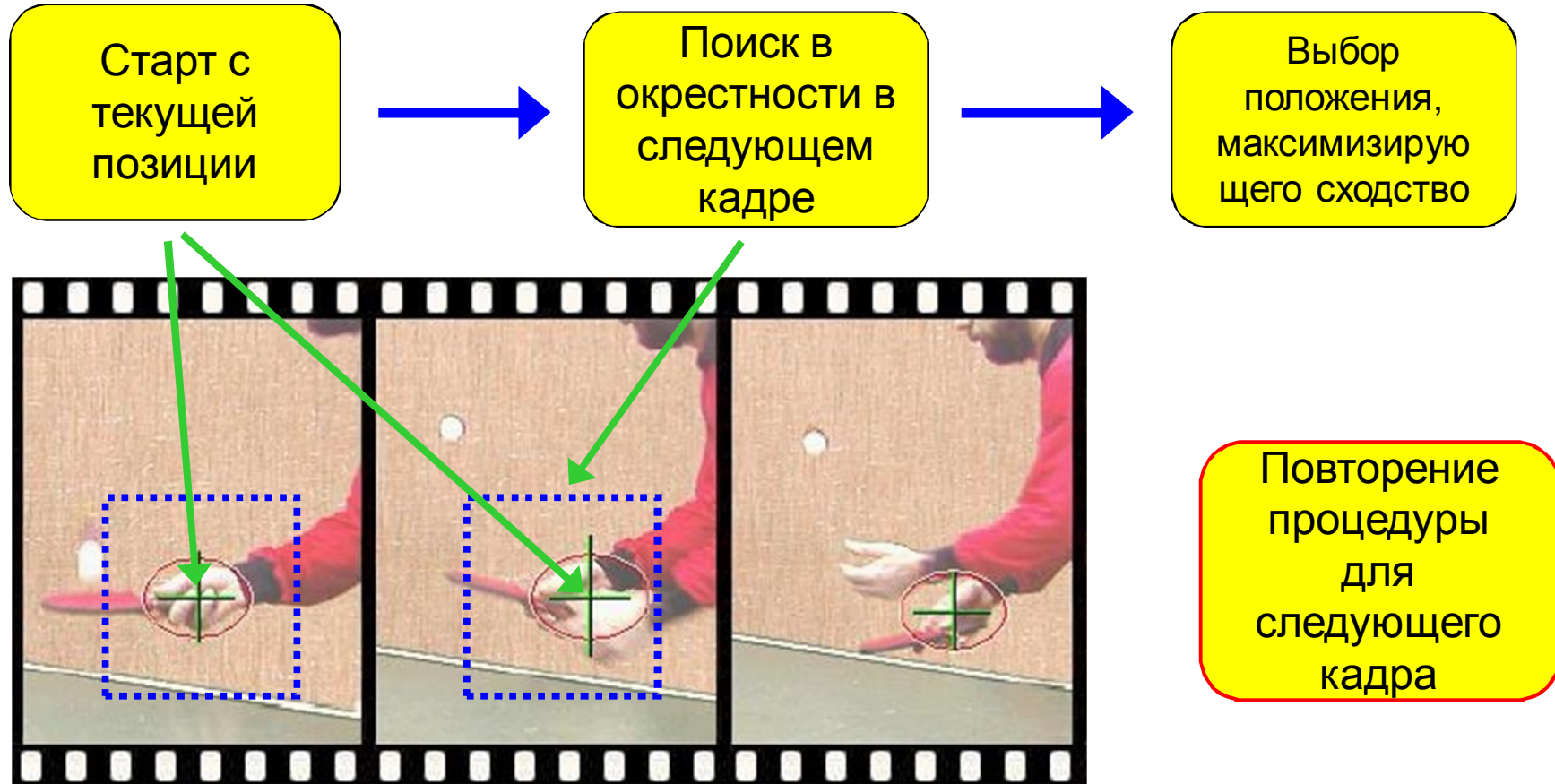
Схема слежения



... Current frame → ...



Схема слежения





Представление объектов



- Как будем описывать модель объекта и фона?
- Некоторое приближение – адаптивные модели для вычитания фона («глобальная цветовая модель фона» и «модель цвета объекта»)



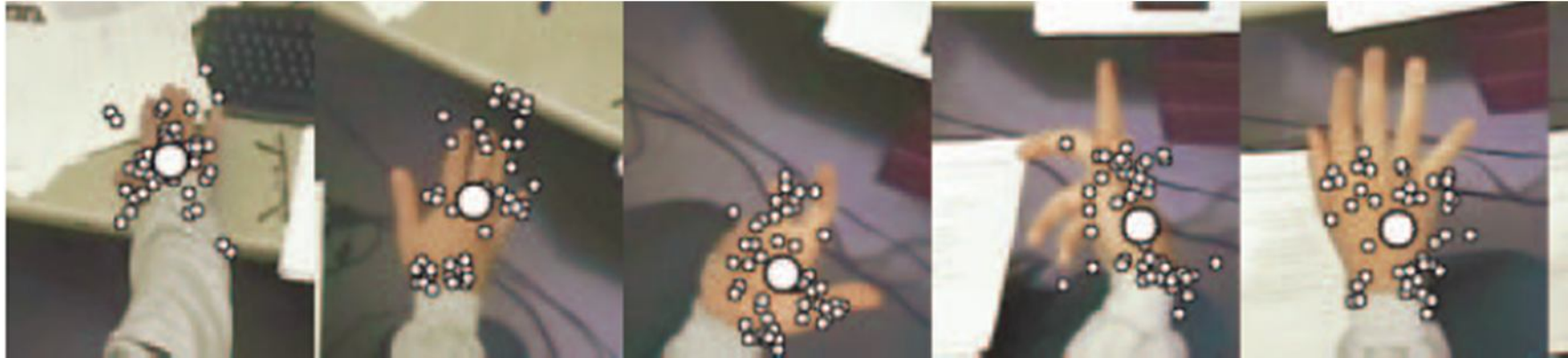
Множество точек

- Простейший kernel метод – отслеживание ключевых точек
- Общая проблема – быстрая деградация точек
- Решение – использование «стаи точек» (flock of features)





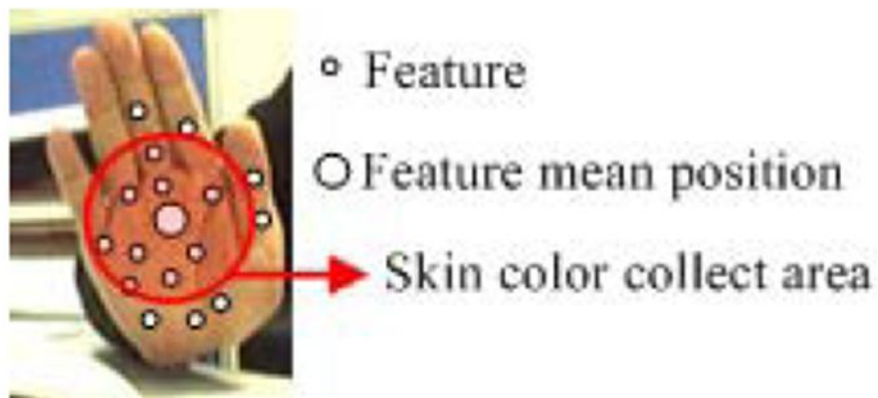
Пример: отслеживание руки



- Условия «стаи»
 - Никакие две особенности не совпадают (порог на близость)
 - Никакая особенность не уходит далеко от медианного центра (порог на удаление)



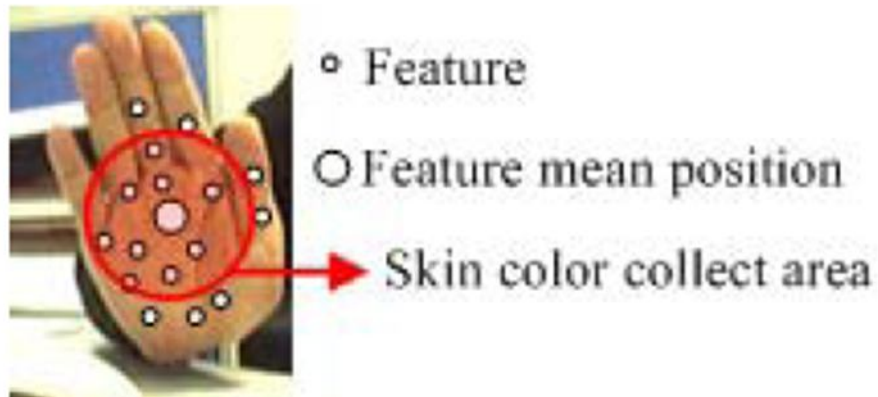
Пример: отслеживание руки



- Шаг 1: инициализация
 - Находим 100 особенностей в рамке руки
 - Вычисляем медиану
 - Вычисляем цветовую статистику в окрестности центра
 - одна гауссиана (или гистограмма нормализованная)
 - Можем вычислить карту цветовой сегментации



Пример: отслеживание руки



- Шаг 2: слежение
 - Отслеживаем особенности
 - Если особенность нарушает условия стаи, её удаляем
- Шаг 3: инициализация новых особенностей
 - Ищем особенности, используем карту вероятности цвета кожи
 - Если вероятность $<$ порога, отбрасываем особенность



Template matching

- Сопоставление шаблонов – поиск положения (bbox, эллипс)
- Ключевой вопрос – моделирование внешности:
 - Само изображение как шаблон (NCC)
 - Ищем на следующем кадре похожие по кросс-корреляции фрагменты
 - Карты краёв
 - Вероятностное моделирование
 - Цвет, текстура, и т.д. (гистограммы обычно)
- Общая проблема – негладкость функции качества, из-за чего градиентные (итеративные) методы часто ошибаются





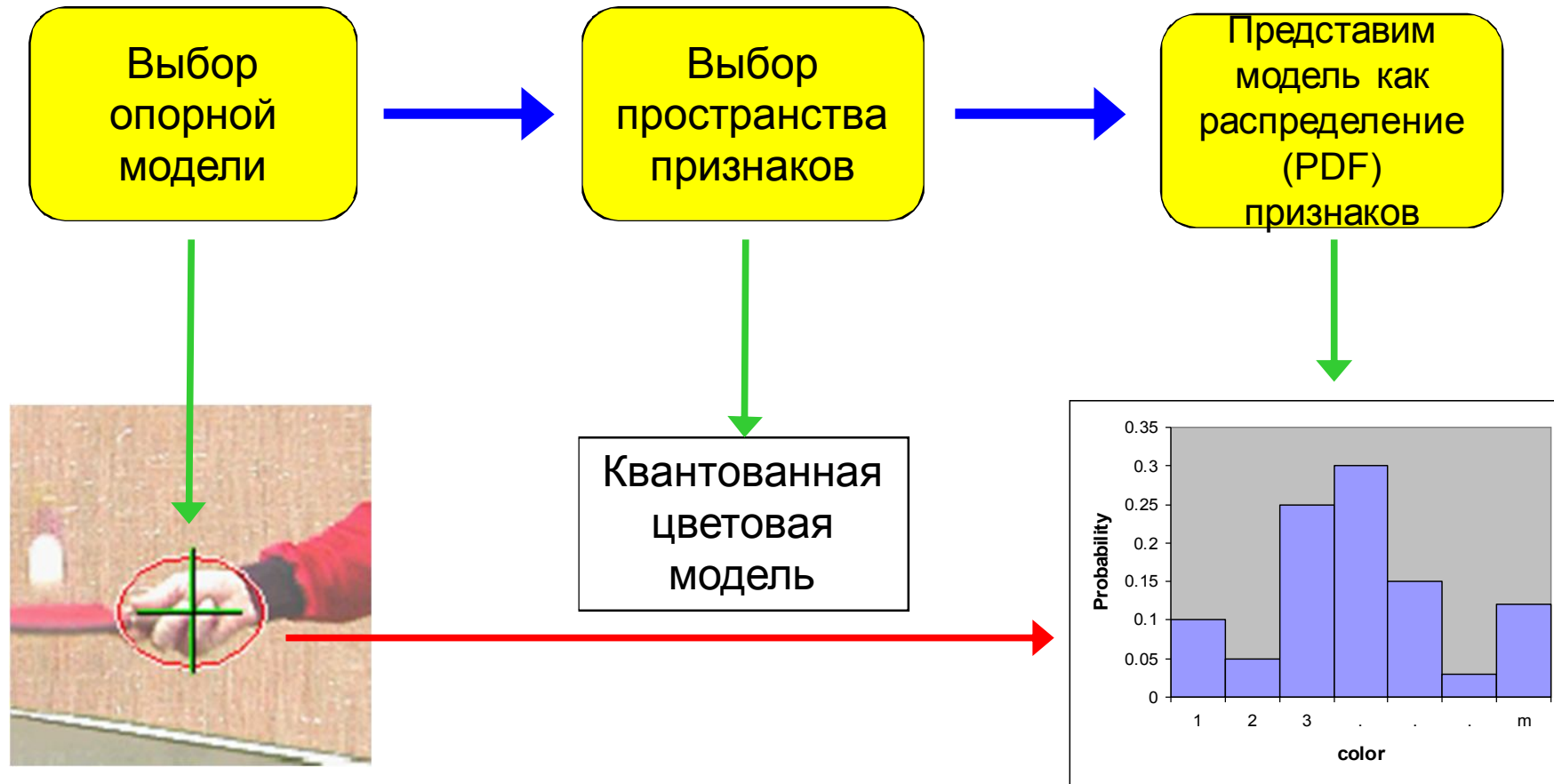
Карта краёв



- Карта краёв выбирается как шаблон для дальнейшего сравнения
 - Вычисляется метрика Хаусдорфа на основании distance transform
 - 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



Слежение сдвигом среднего

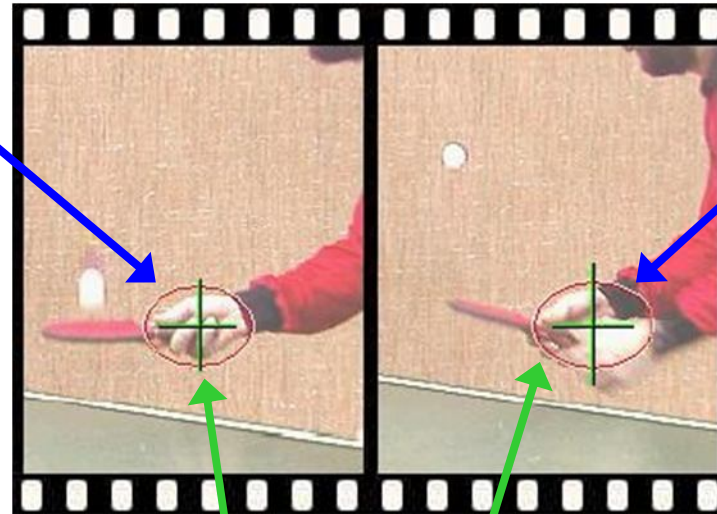
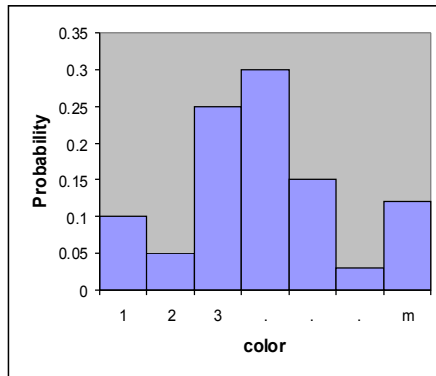


- Метод на основе вероятностного представления модели объекта
- Решение проблемы гладкости целевой функции

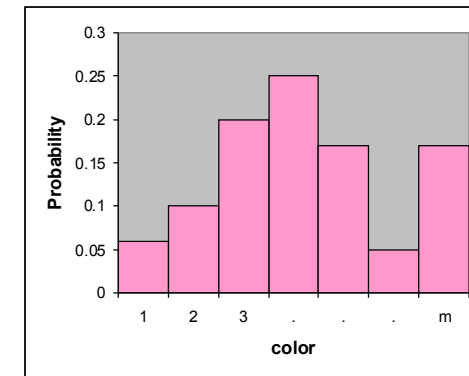


Слежение сдвигом среднего

Модель
(центр в 0)



Кандидат
(центр в y)



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

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

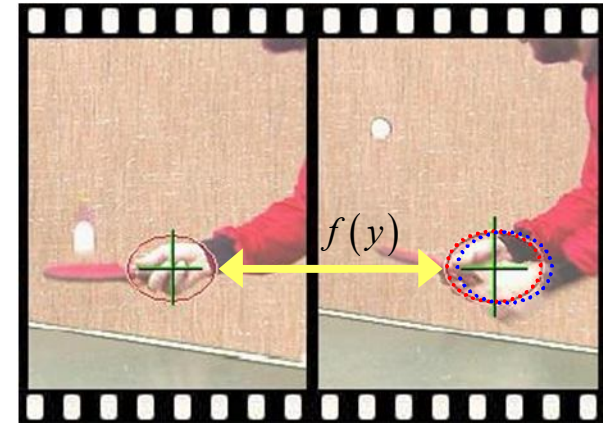
СХОДСТВО: $f(y) = f[\vec{q}, \vec{p}(y)]$



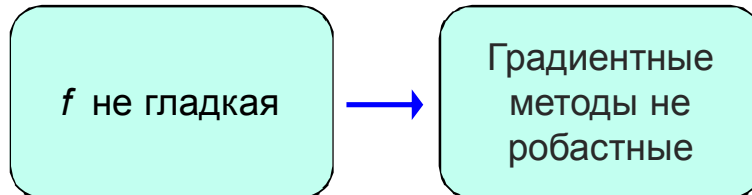
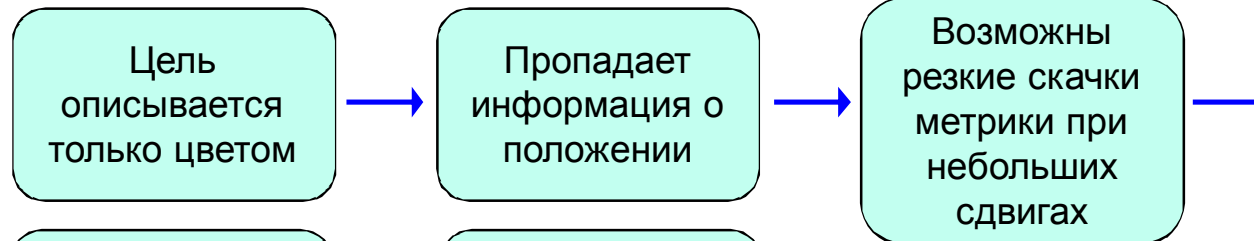
Слежение сдвигом среднего

Сходство:

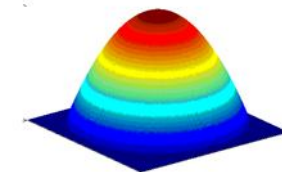
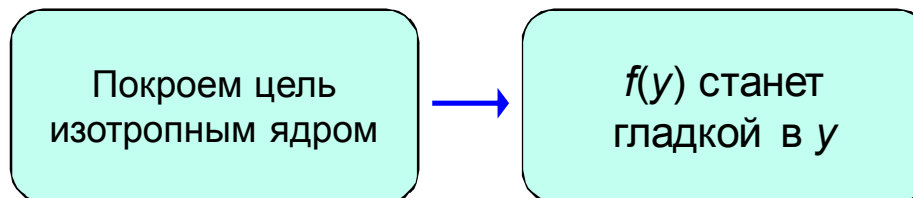
$$f(y) = f[\vec{p}(y), \vec{q}]$$



Проблема:



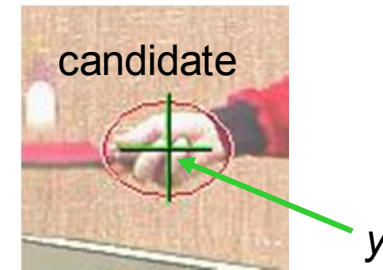
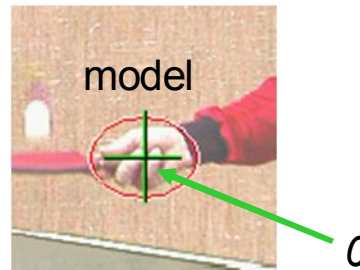
Решение:





Слежение сдвигом среднего

$\{x_i\}_{i=1..n}$ Пиксели цели



$k(x)$ Изотропное, выпуклое, дифференцируемое, монотонно убывающее ядро (Пиксели по краям могут перекрываться, взаимодействовать с фоном и т.д.)

$b(x)$ Индекс ячейки гистограммы (1..m) для пиксела x

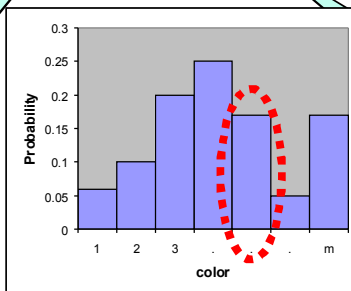
Вероятность фичи u в модели

Вероятность фичи u в кандидате

$$q_u = C \sum_{b(x_i)=u} k(\|x_i\|^2)$$

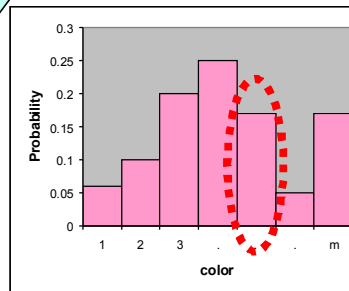
$$p_u(y) = C_h \sum_{b(x_i)=u} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right)$$

Нормализация



Вес пиксела

Нормализация



Вес пиксела



Слежение сдвигом среднего

Модель цели: $\vec{q} = (q_1, \dots, q_m)$

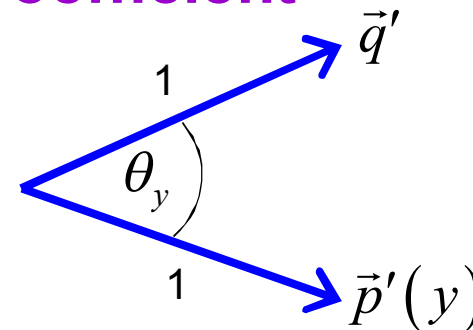
Кандидат: $\vec{p}(y) = (p_1(y), \dots, p_m(y))$

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

The Bhattacharyya Coefficient

$$\vec{q}' = (\sqrt{q_1}, \dots, \sqrt{q_m})$$

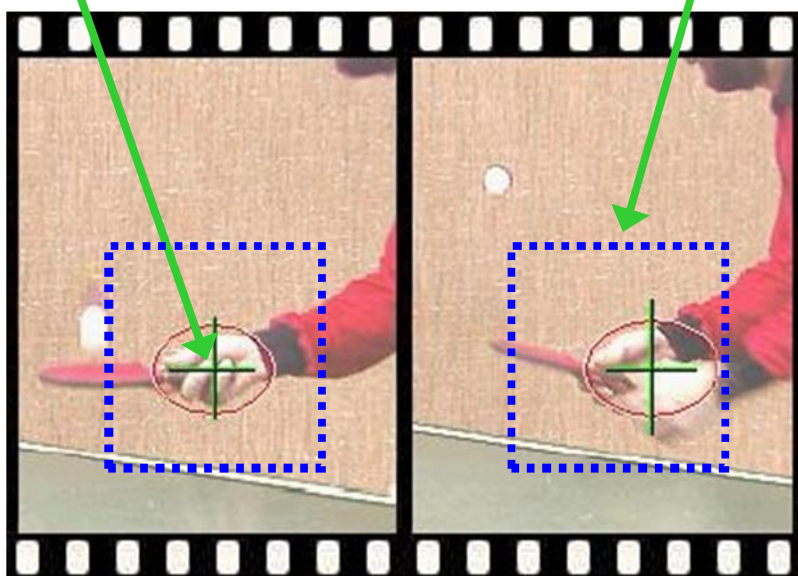
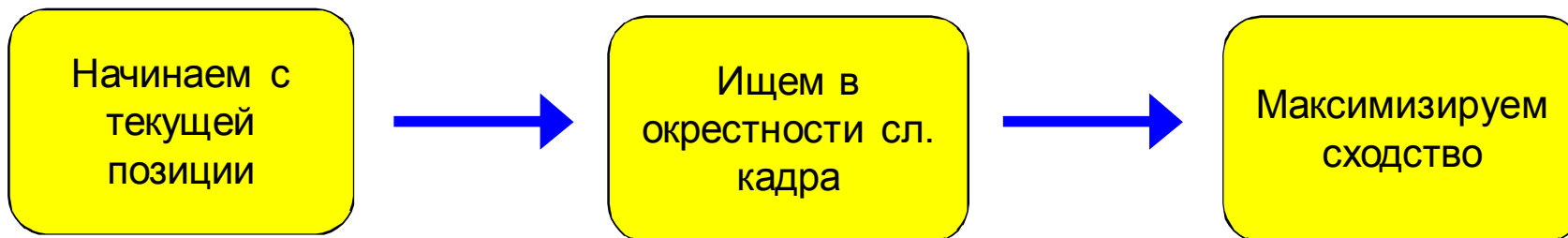
$$\vec{p}'(y) = (\sqrt{p_1(y)}, \dots, \sqrt{p_m(y)})$$



$$f(y) = \cos \theta_y = \frac{p'(y)^T q'}{\|p'(y)\| \cdot \|q'\|} = \sum_{u=1}^m \sqrt{p_u(y) q_u}$$

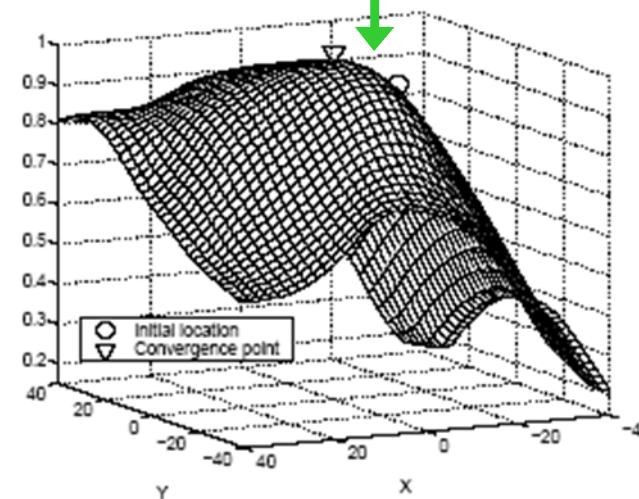


Слежение сдвигом среднего



$$\vec{q}$$

$$\vec{p}(y)$$



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



Слежение сдвигом среднего

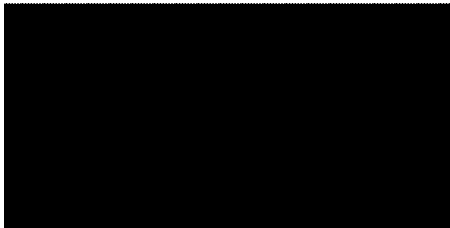
$$f(y) = \sum_{u=1}^m \sqrt{p_u(y) q_u}$$

Положение модели: y_0

Положение кандидата: y

Линейное приближение.
(вокруг y_0)

$$f(y) \approx$$



$$\frac{1}{2} \sum_{u=1}^m p_u(y) \sqrt{\frac{q_u}{p_u(y_0)}}$$

Независимо от y

$$p_u(y) = C_h \sum_{b(x_i)=u} k\left(\left\|\frac{y-x_i}{h}\right\|^2\right)$$

$$\frac{C_h}{2} \sum_{i=1}^n w_i k\left(\left\|\frac{y-x_i}{h}\right\|^2\right)$$

Оценка плотности!
(как функции от y)

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

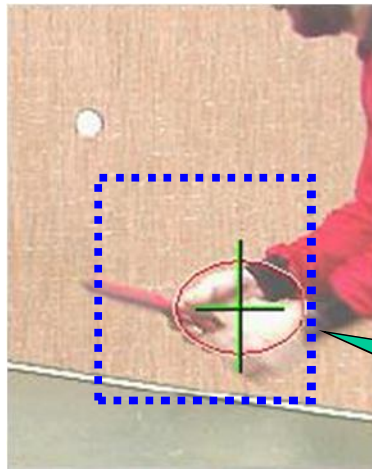


Слежение сдвигом среднего

Мода

$$\frac{C_h}{2} \sum_{i=1}^n w_i k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right) = \text{целевой максимум}$$

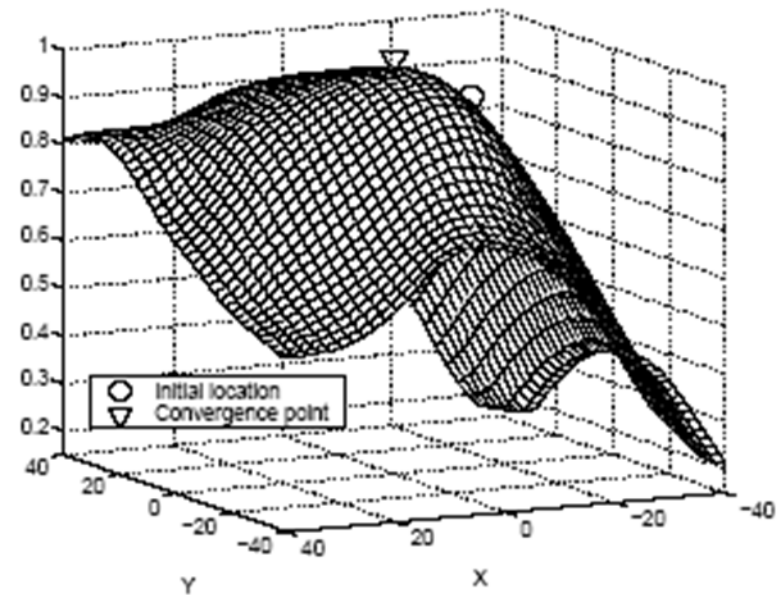
Предположения:



Представление
цели достаточно
для
распознавания



Одна мода в
окрестности





Слежение сдвигом среднего

Мода $\frac{C_h}{2} \sum_{i=1}^n w_i k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right) =$ целевой максимум

$$K(x) = ck \left(\|x\|^2 \right) \quad k'(x) = -g(x)$$

Original
Mean-Shift:

Find mode of $c \sum_{i=1}^n k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right)$ using $y_1 = \frac{\sum_{i=1}^n x_i g \left(\left\| \frac{y_0 - x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n g \left(\left\| \frac{y_0 - x_i}{h} \right\|^2 \right)}$

Extended
Mean-Shift:

Find mode of $c \sum_{i=1}^n w_i k \left(\left\| \frac{y - x_i}{h} \right\|^2 \right)$ using $y_1 = \frac{\sum_{i=1}^n x_i w_i g \left(\left\| \frac{y_0 - x_i}{h} \right\|^2 \right)}{\sum_{i=1}^n w_i g \left(\left\| \frac{y_0 - x_i}{h} \right\|^2 \right)}$



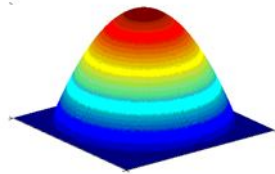
Слежение сдвигом среднего

Особый класс
радиально-
симметричных ядер

$$K(x) = ck(\|x\|^2)$$

$$k'(x) = -g(x)$$

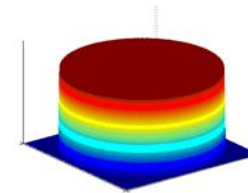
Epanechnikov kernel:



$$k(x) = \begin{cases} 1 - \|x\|^2 & \text{if } \|x\| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$



Uniform kernel:

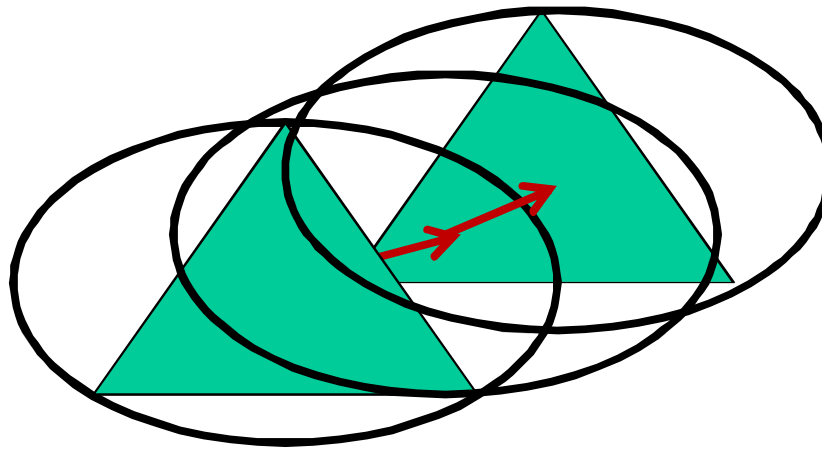


$$g(x) = -k(x) = \begin{cases} 1 & \text{if } \|x\| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$y_1 = \frac{\sum_{i=1}^n x_i w_i g\left(\left\|\frac{y_0 - x_i}{h}\right\|^2\right)}{\sum_{i=1}^n w_i g\left(\left\|\frac{y_0 - x_i}{h}\right\|^2\right)} \longrightarrow y_1 = \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i}$$



Иллюстрация





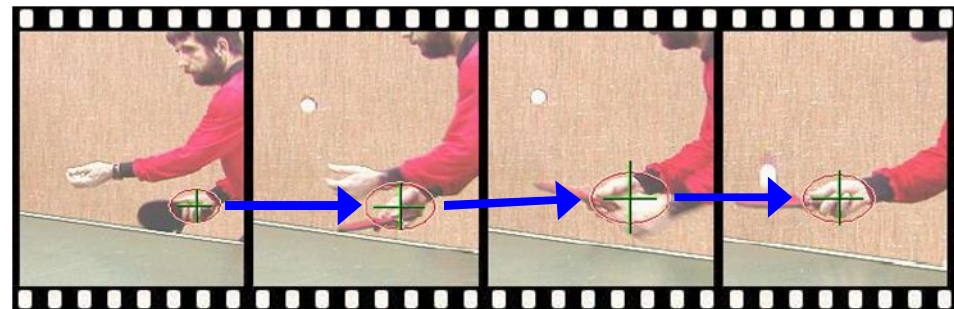
Слежение сдвигом среднего

Проблема :

Масштаб
цели
меняется



Масштаб
(h) ядра
должен
меняться

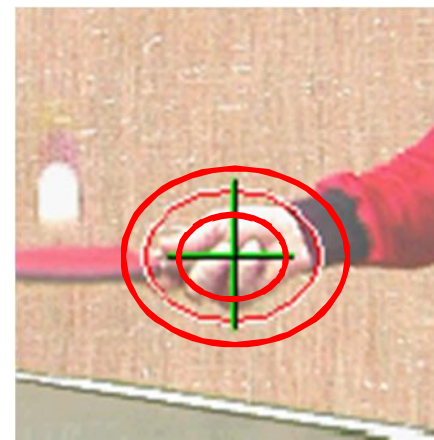


Решение:

Прогоняем
поиск 3 раза с
разными h

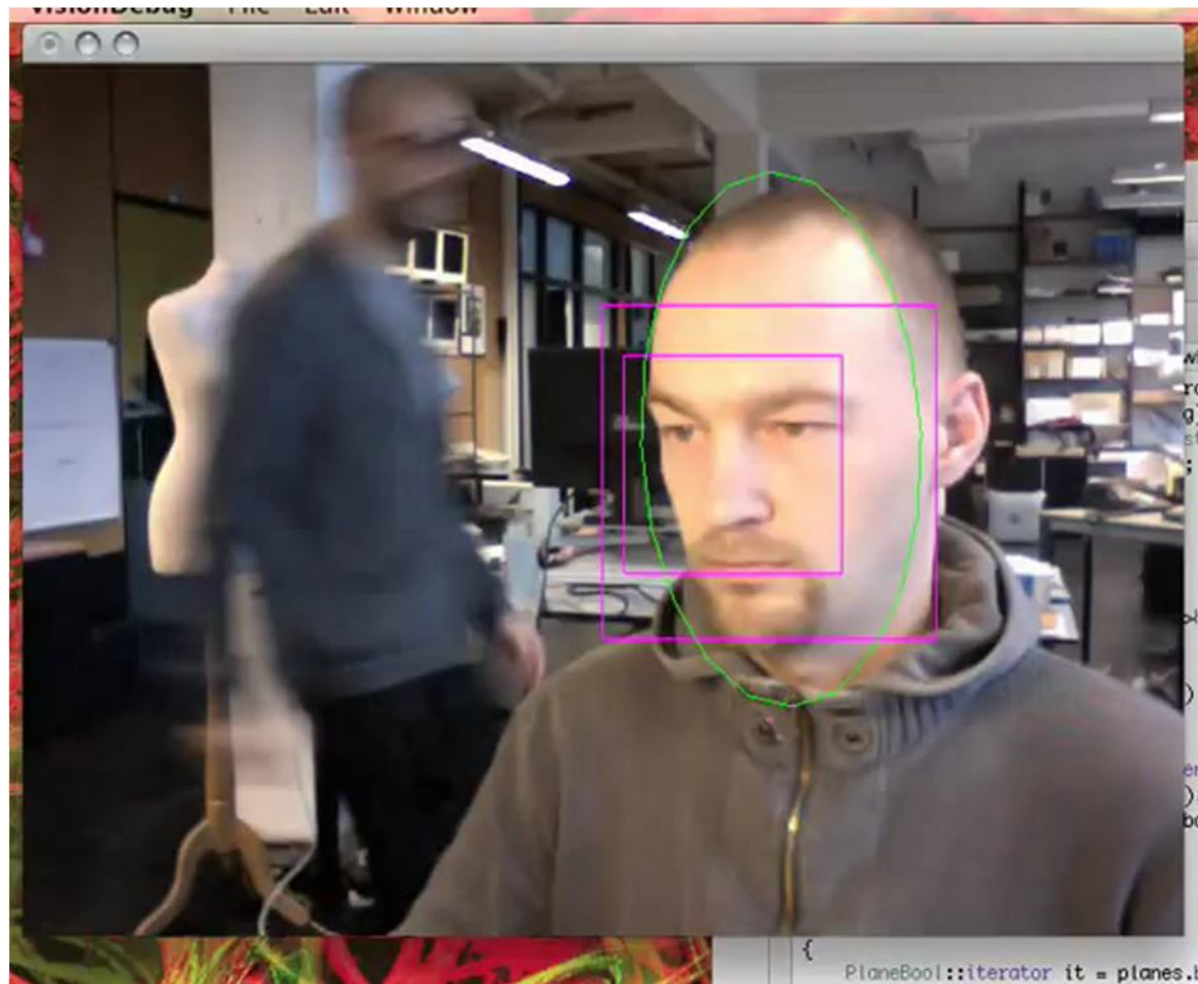


Выбираем h ,
на котором
достигается
максимум





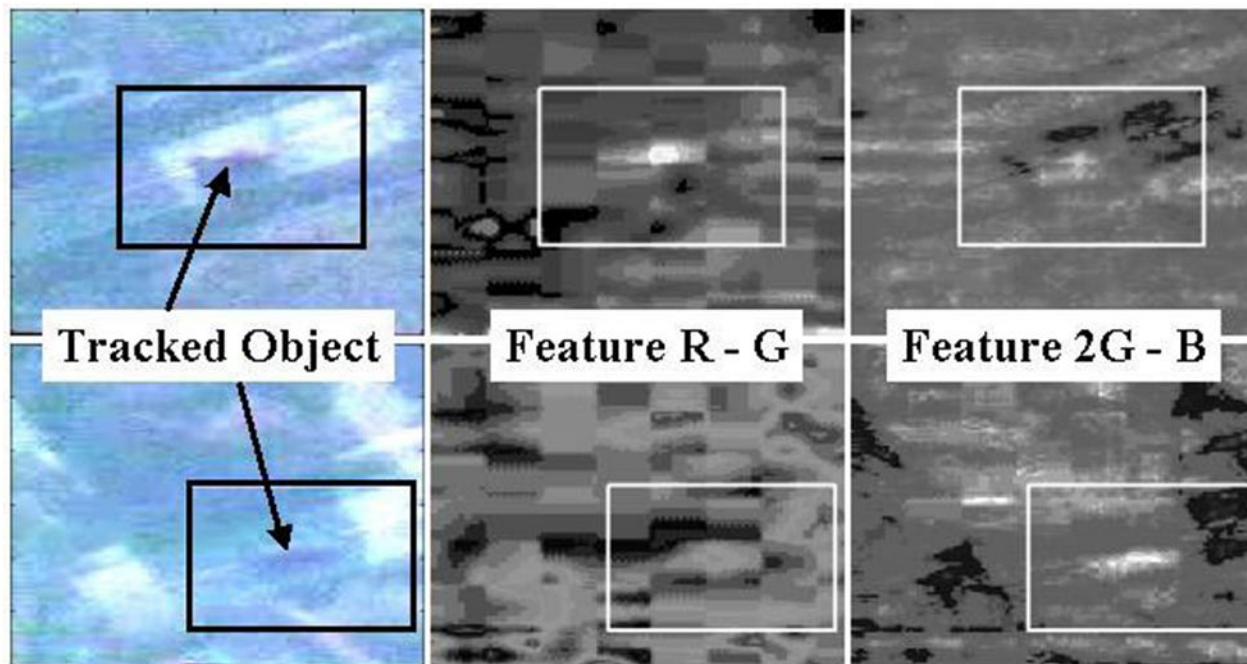
Слежение сдвигом среднего



Пространство признаков: $16 \times 16 \times 16$ квантованное RGB



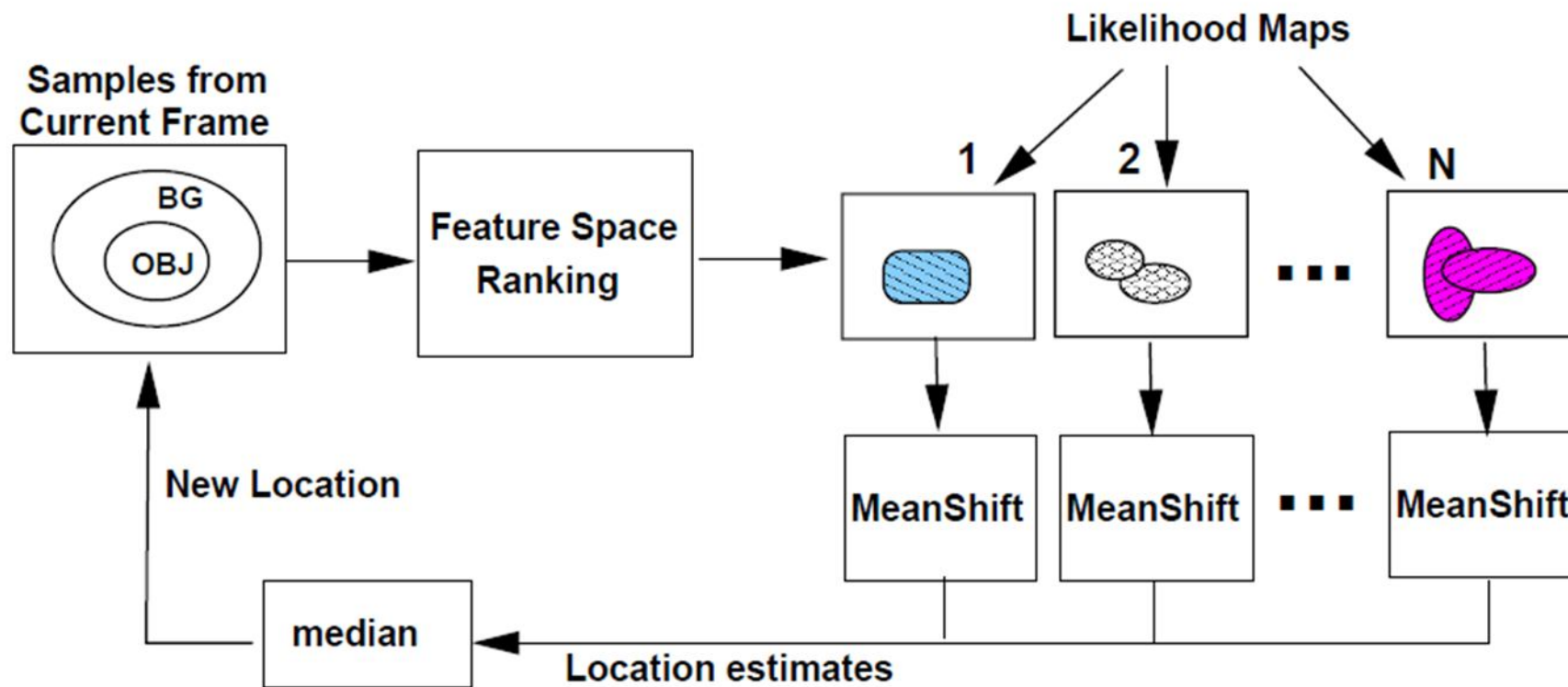
Выбор признаков



- В зависимости от разных факторов, объект может быть различим в разных каналах, по разным признакам

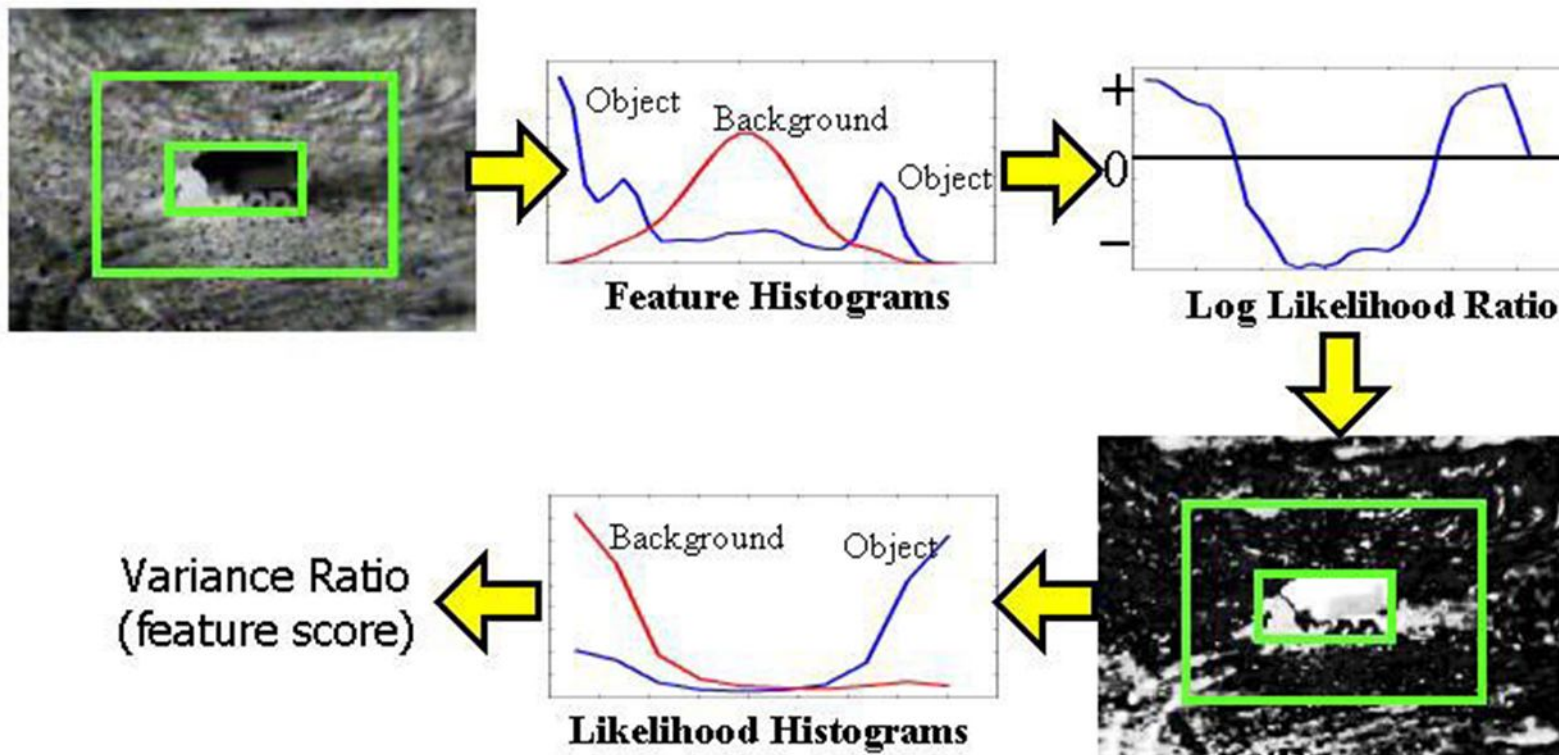


Схема метода





Оценка качества канала



$$L(i) = \log \frac{\max \{p(i), \delta\}}{\max \{q(i), \delta\}}$$

$$\text{var}(x) = Ex^2 - (Ex)^2$$

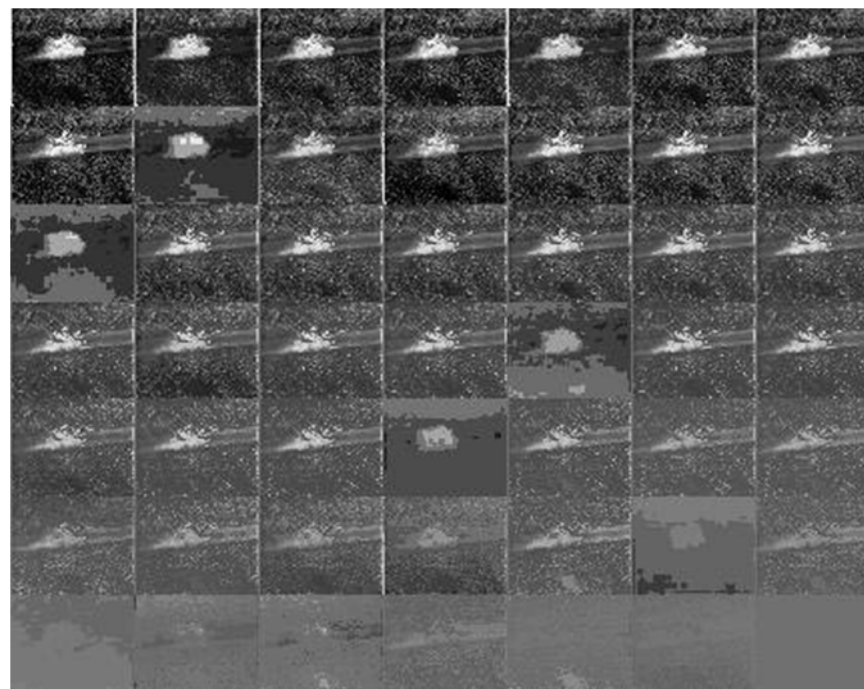
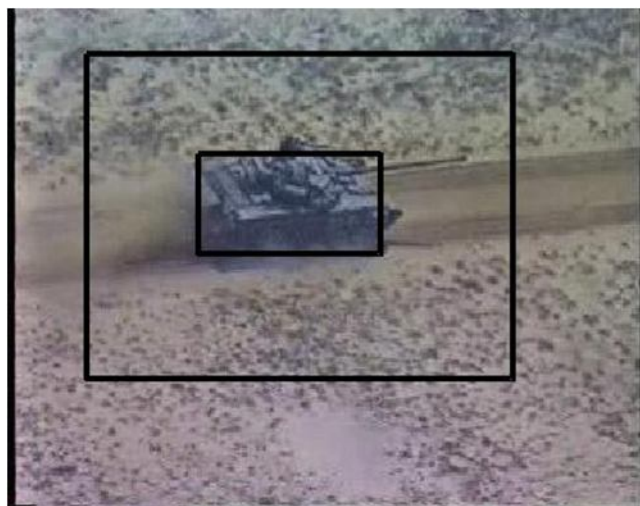
$$\text{var}(L; a) = \sum a(i)L^2(i) - [\sum a(i)L(i)]^2$$

$$\text{VR}(L; p, q) \equiv \frac{\text{var}(L; (p + q)/2)}{[\text{var}(L; p) + \text{var}(L, q)]}$$

Рассмотрим отдельный канал



Оценка каналов

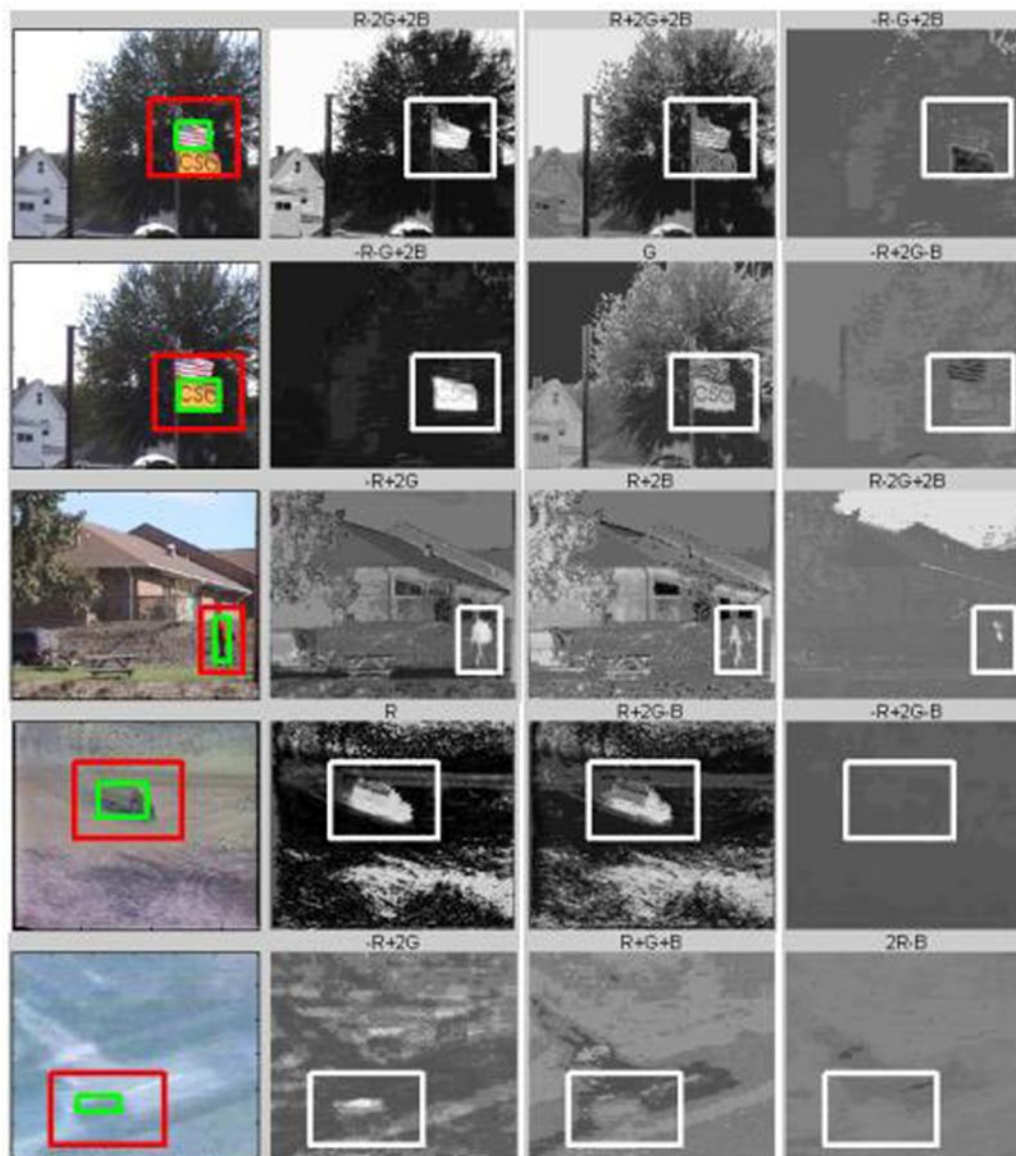


- Всего 49 каналов, как линейные комбинации цветовых каналов:

$$F_1 \equiv \{w_1R + w_2G + w_3B \mid w_* \in [-2, -1, 0, 1, 2]\}$$



Примеры оценки каналов

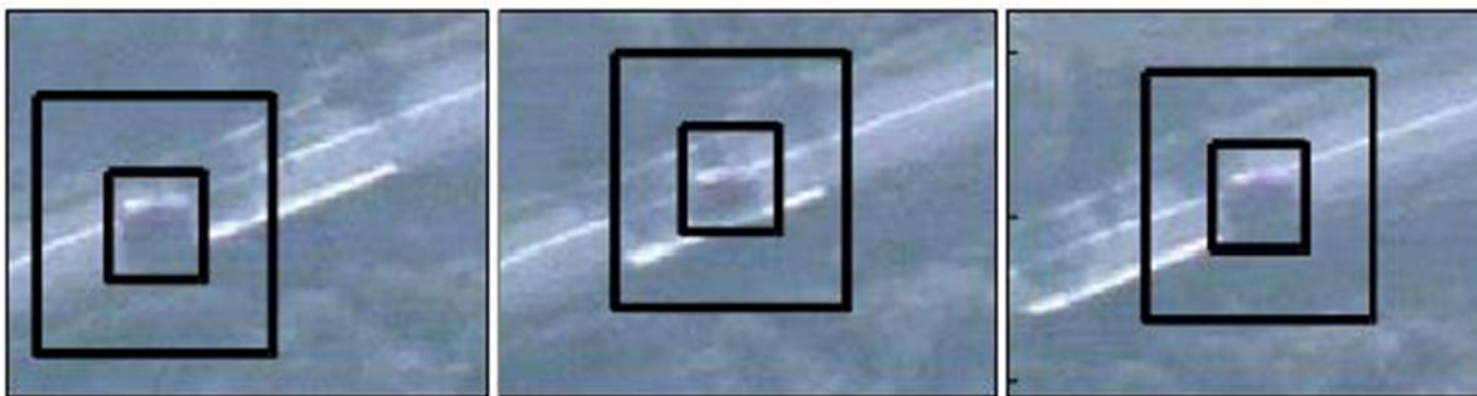




Сравнение методов



Один фиксированный канал (MeanShift)



Выбор каналов на лету

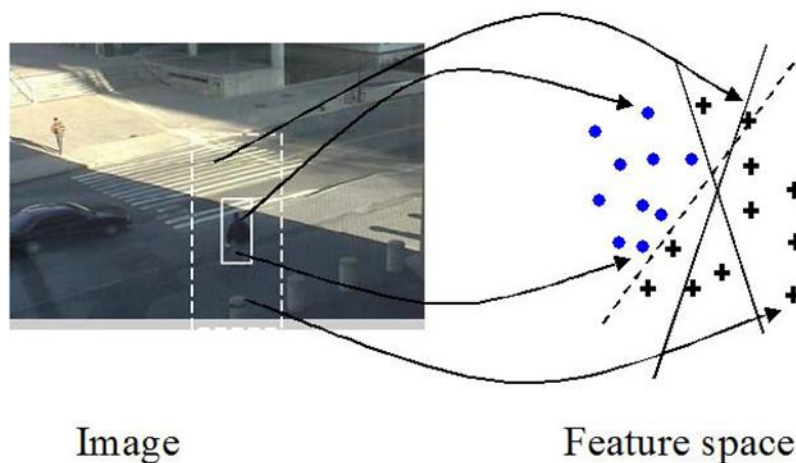
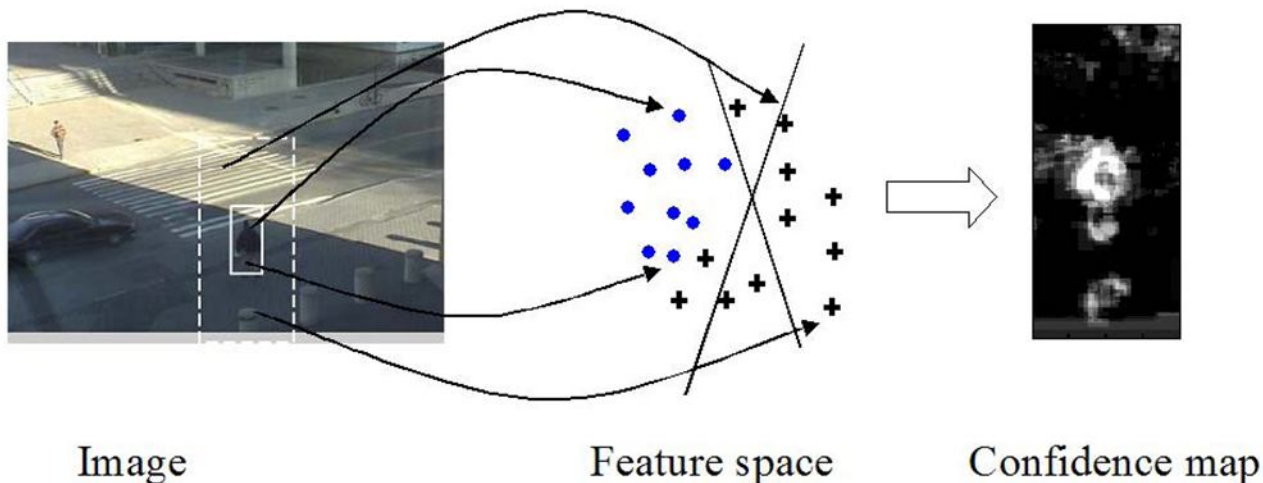


Комитетный трекер

- Для слежения будем классифицировать пиксели на объект/фон сильным классификатором (бустинг)
- На каждом кадре убирается K самых старых слабых классификаторов
- Строится K новых классификаторов для текущего кадра



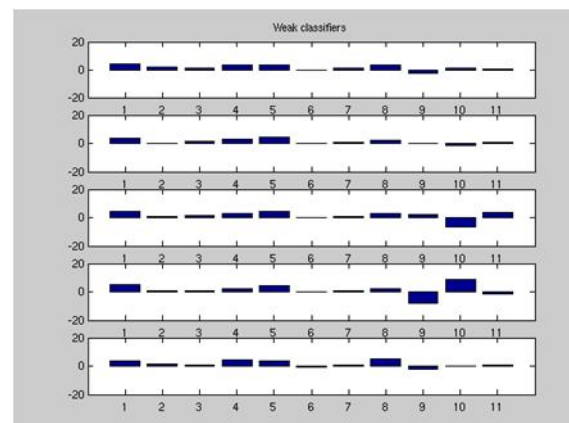
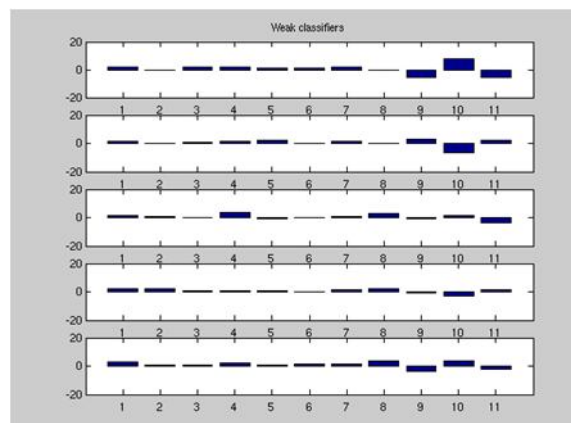
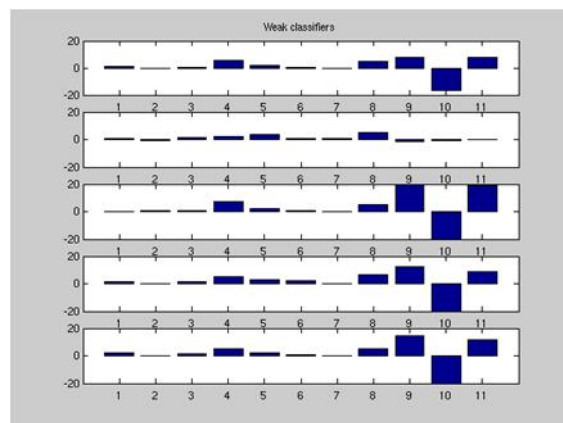
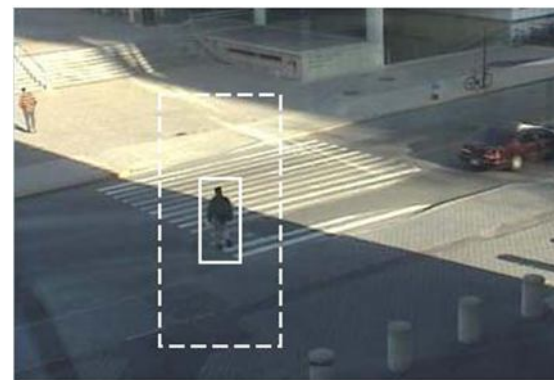
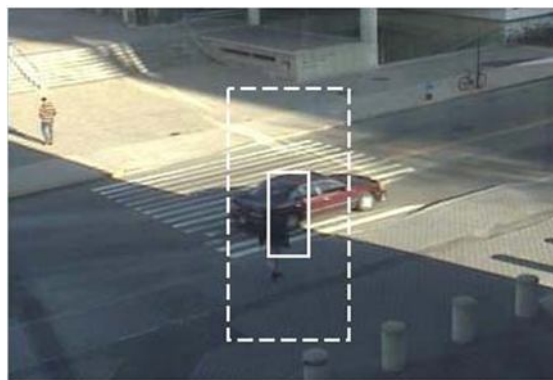
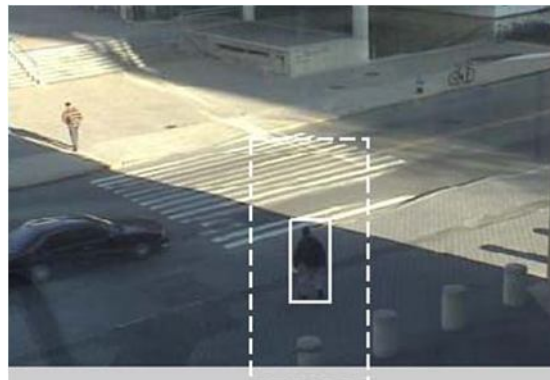
Слабые классификаторы



- 11D признаки для каждого пикселя (HOG + RGB)
- Линейный классификатор по МНК



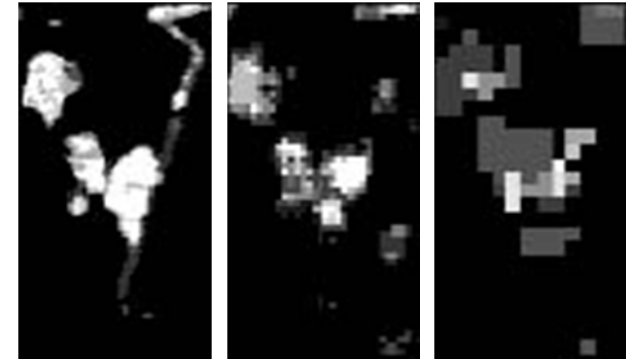
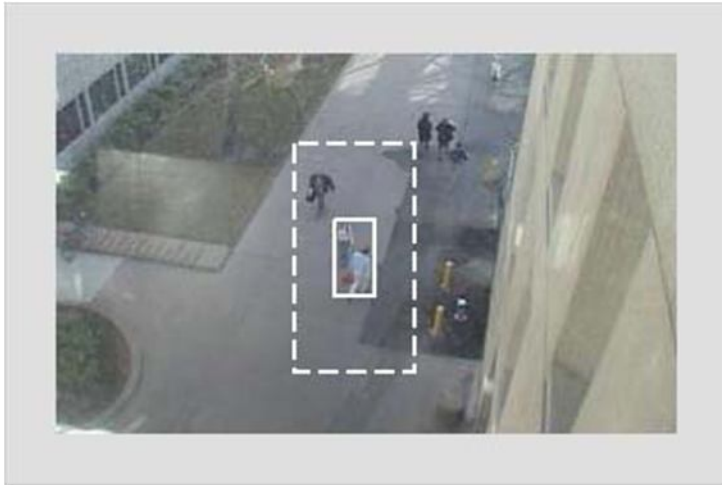
Адаптация классификаторов



- Демонстрация изменения слабых классификаторов
- 5 слабых классификаторов, вверху – самый свежий
- Уровни задают веса признаков, первые 8 –HOG, далее RGB



Многомасштабное слежение



- Для 3х уровней масштаба строится свой сильный классификатор
- На каждом уровне строится своя карта вероятности объекта
- Карты приводятся к одному уровню и смешиваются
- Находим пик карты с помощью MeanShift



Результаты



(a)



(b)



(c)



(d)



(e)



(f)



(g)



(h)

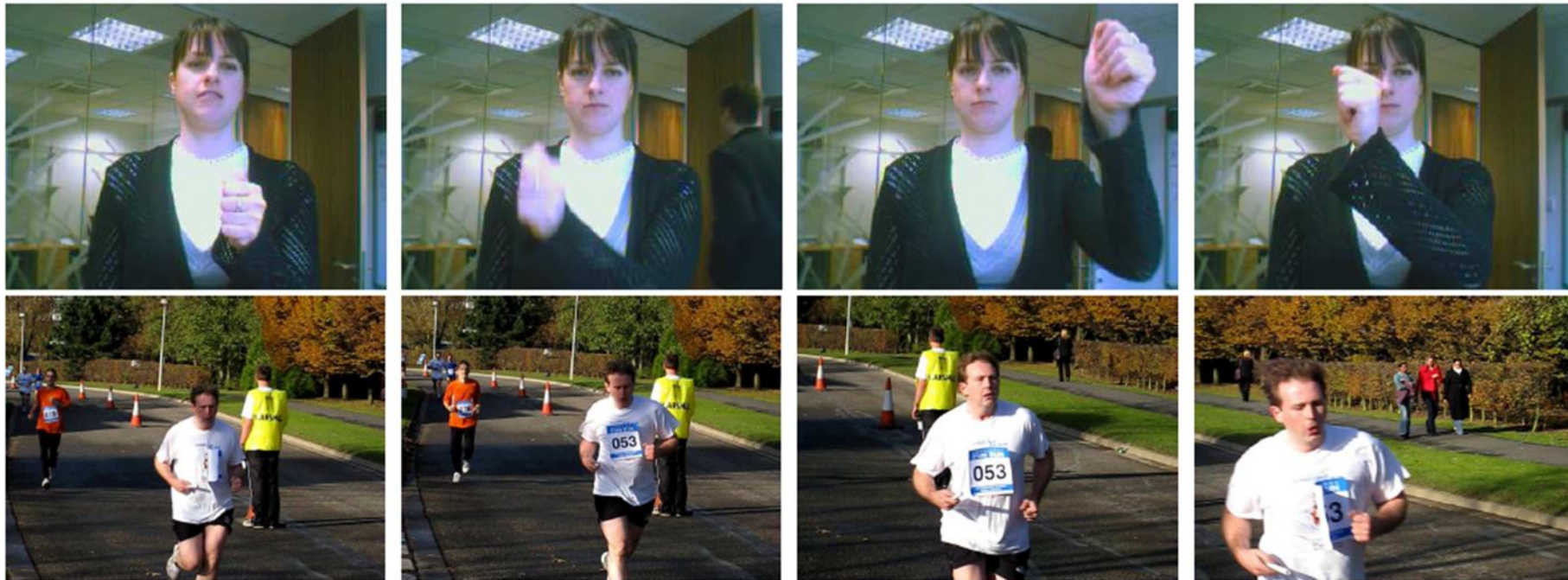


Объединение трекеров

- Идея:
 - Протестировать ряд базовых трекеров на наборе выборок, оценить качество и надежность
 - Найти зависимость качества трекера от уверенности (confidence value)
 - Найти хорошо дополняющие друг друга группы методов
 - При падении доверия к первому методу в группе мы переключаемся на второй метод



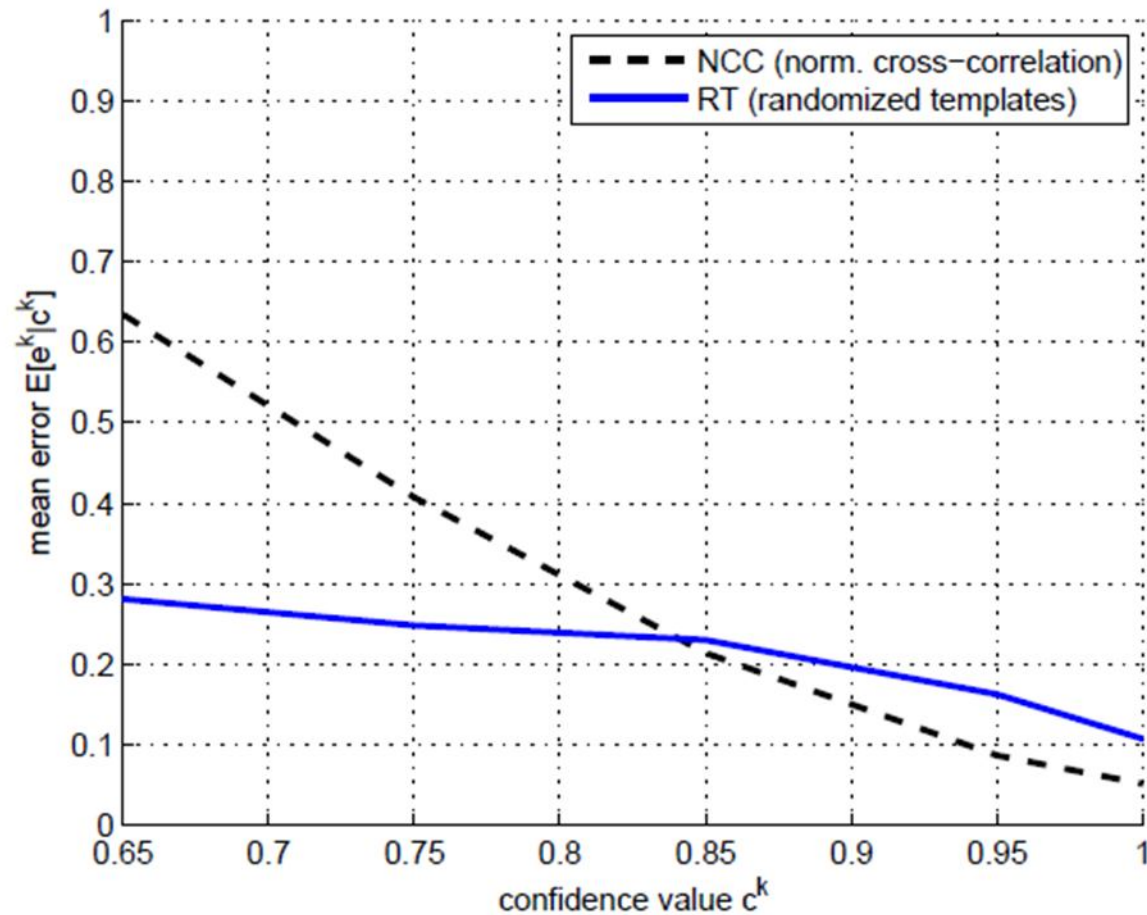
Тестовые данные



- Indoor и Outdoor данные
- Во всех присутствует изменение позы, размера, размытие из движения (motion blur)



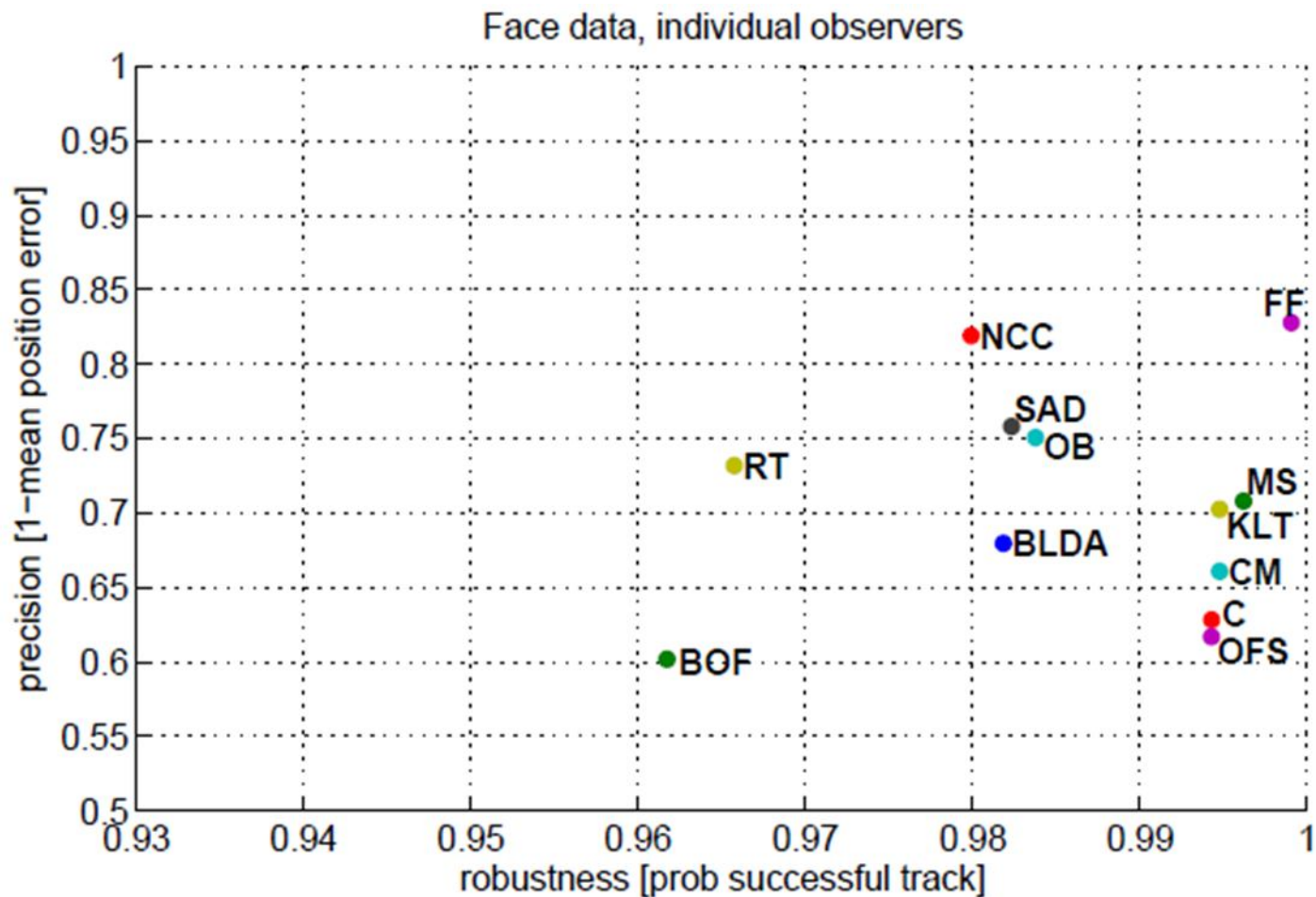
Обучение метрики доверия



- По тестовым данным обучаем $p(e|c)$, где c – параметр доверия, специфичный для каждого метода

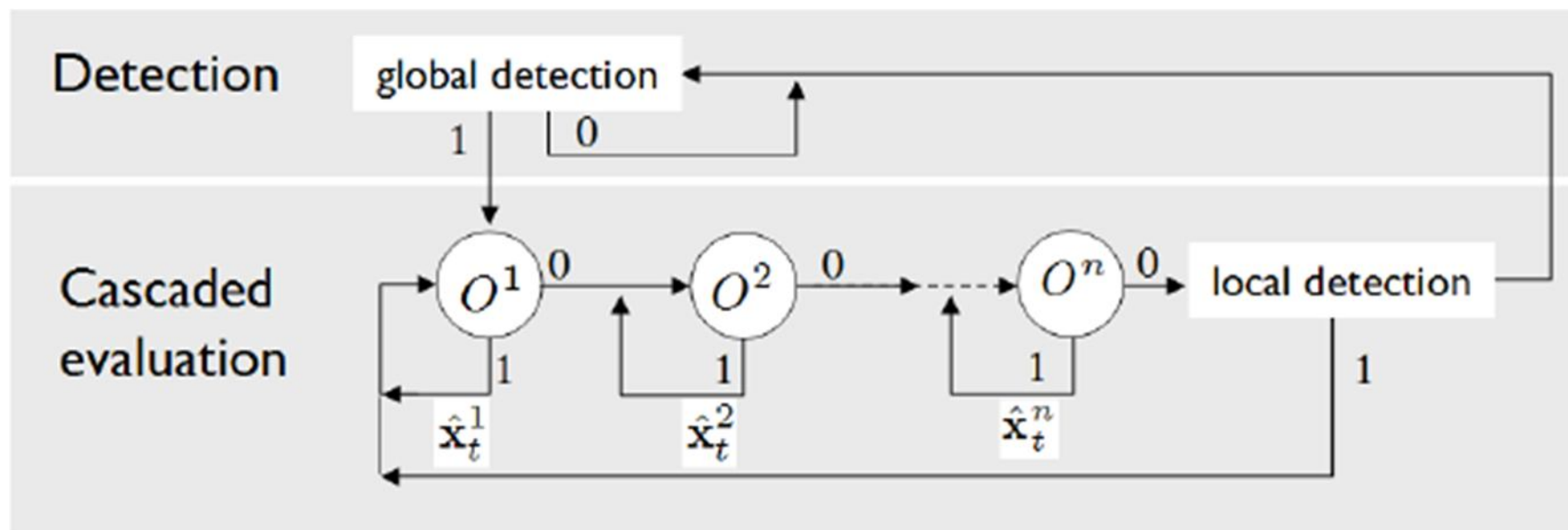


Результаты оценки





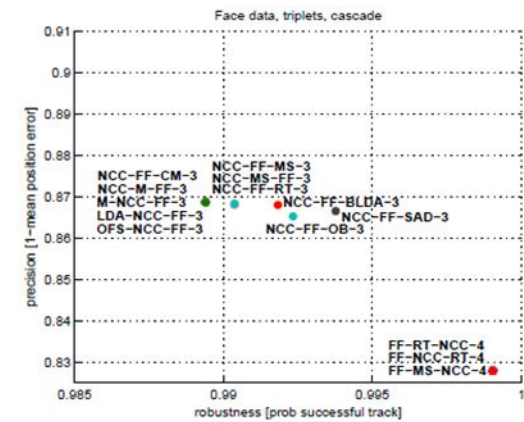
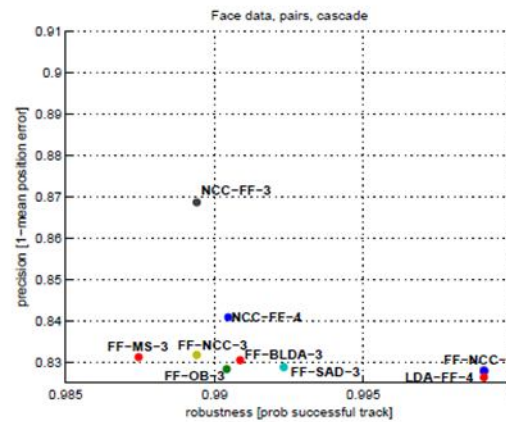
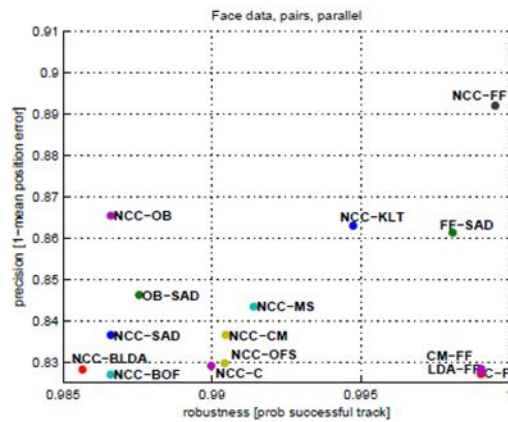
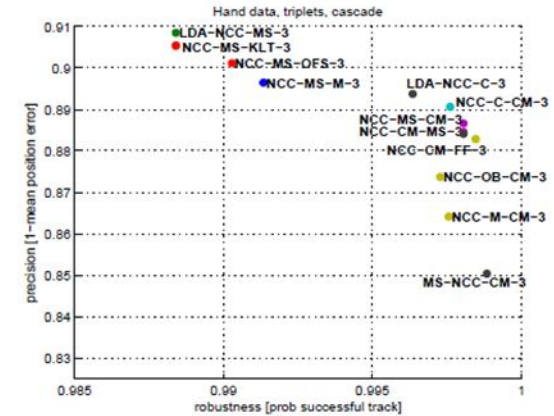
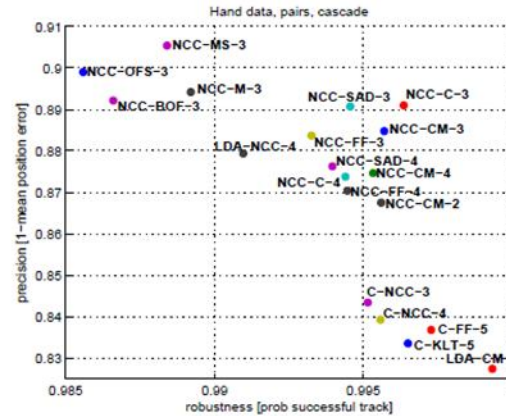
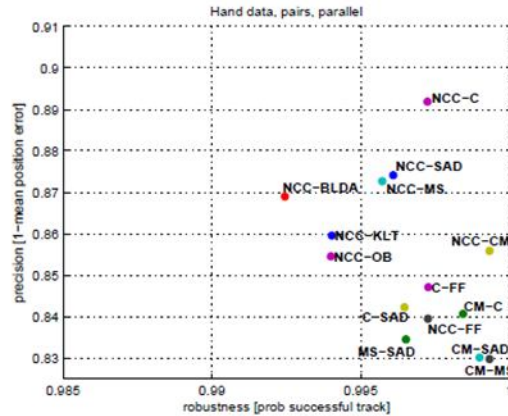
Каскадное объединение



- Если оценка качества по доверию для метода 1 падает ниже порога, переключаемся на метод 2
- Когда все методы дают низкое качество, происходит переинициализация метода

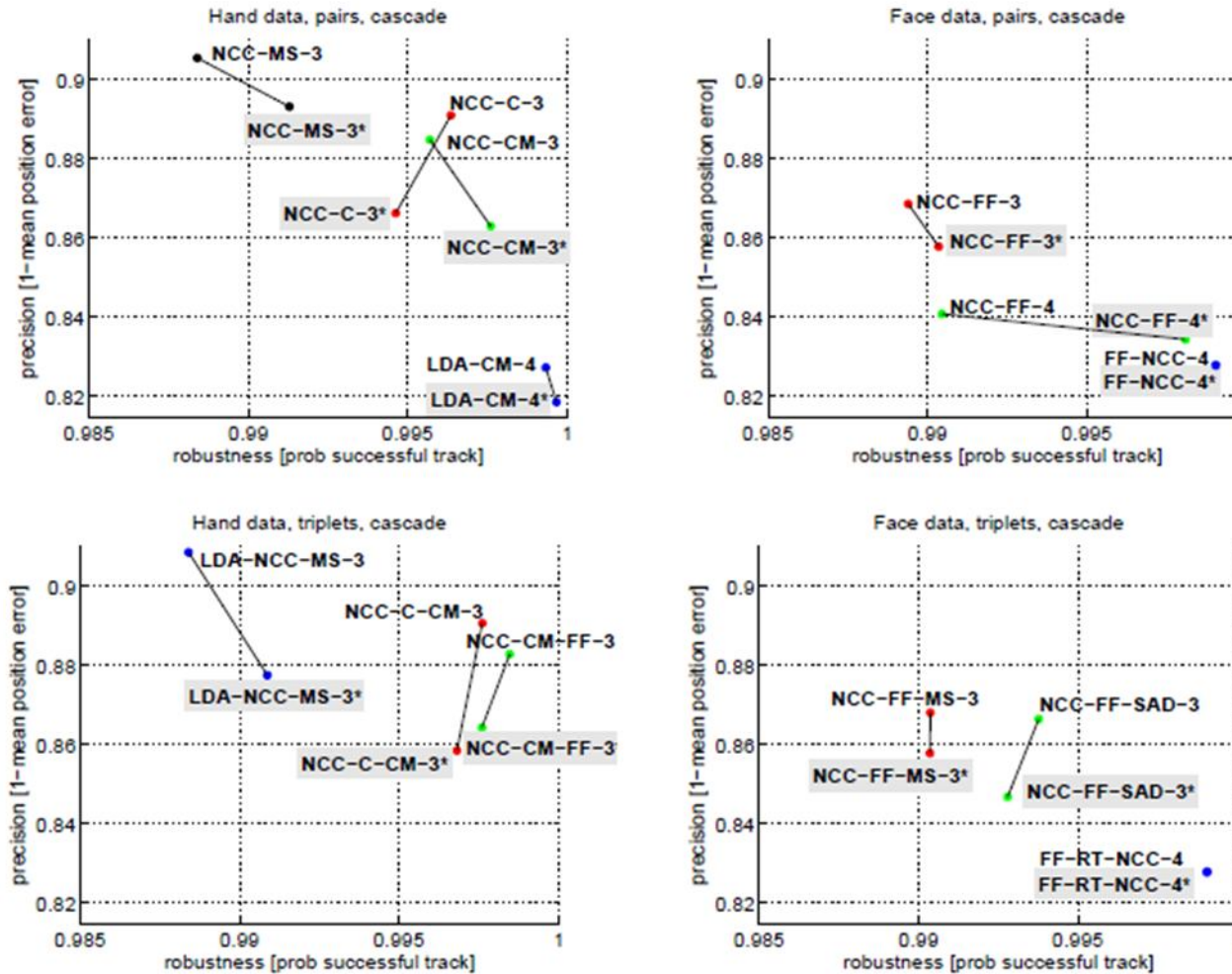


Результаты объединения





Сравнение



- Сравнение результатов на тестовых данных и на других реальных данных



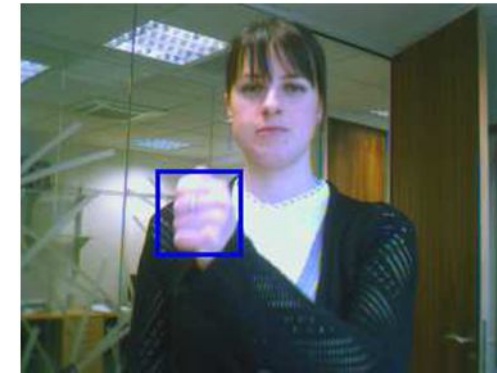
Примеры работы



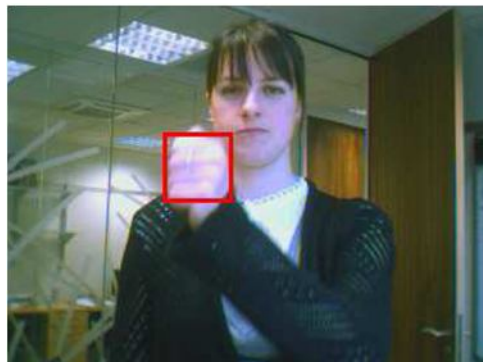
NCC



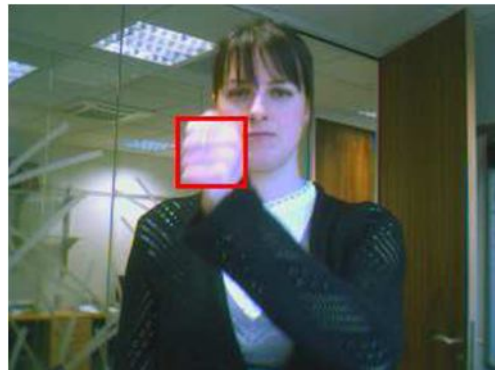
NCC



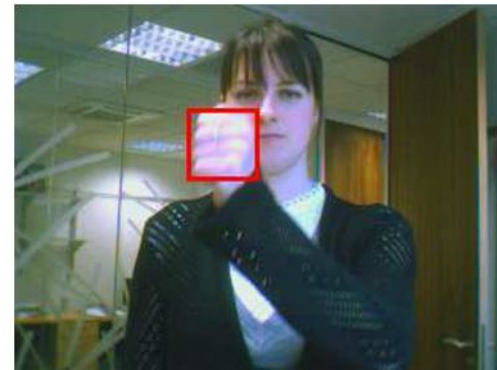
NCC



CM



CM



CM

- Работает NCC, при возникновении motion blur система переключается на CM (color + motion map)



Примеры работы



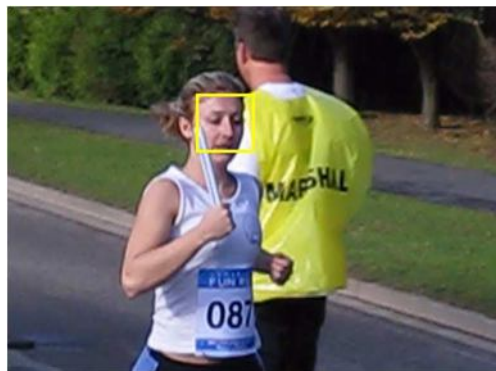
NCC



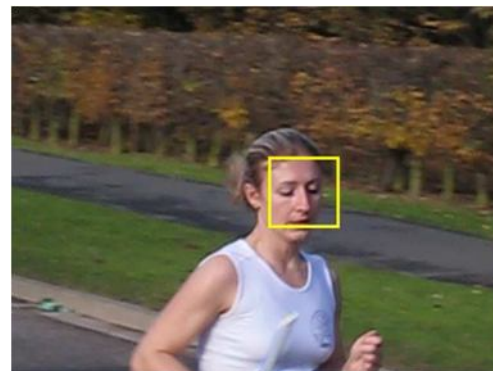
NCC



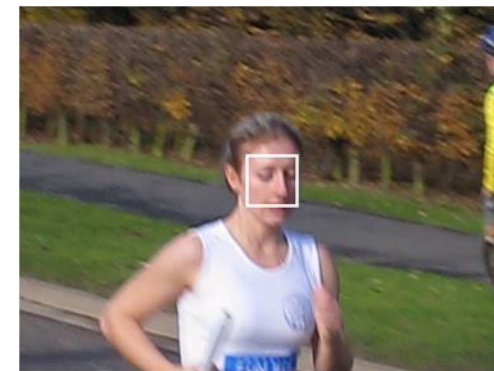
NCC



FF



FF



MS

- Работает NCC, потом переключение на FF (flock of features), потом на MS (mean shift)



Примеры работы



CVPR
2009
MIAMI

Learning to Track with Multiple Observers

Björn Stenger¹
Thomas Woodley²
Roberto Cipolla²

¹ Toshiba Research Europe
² University of Cambridge

The slide features a dark background with a white title and author information. At the top left, there is a logo for CVPR 2009 Miami, which includes the text 'CVPR 2009 MIAMI' and a small image of a building. The main title 'Learning to Track with Multiple Observers' is centered in a large, white, sans-serif font. Below the title, the authors' names are listed in two columns, with superscripted numbers indicating their affiliations. The affiliations are listed to the right of the authors' names.



Резюме лекции

- Вычитание фона – основа систем видеонаблюдения со стационарным фоном
- Если фон динамический, то нужно отдельно выделять объекты и затем применять методы отслеживания объектов
- Зато есть ряд базовых методов, каждый в своём случае хорошо работает
 - NCC
 - Flocks of features
 - MeanShift
- Выход в их комбинации, но это иногда существенно замедляет скорость работы