



Нижегородский государственный университет им. Н.И. Лобачевского

***Разработка мультимедийных приложений
с использованием библиотек OpenCV и IPP***

Лекция
Отслеживание движения и алгоритмы
сопровождения ключевых точек

При поддержке компании Intel

Кустикова В.Д.,
кафедра математического обеспечения ЭВМ

Содержание (1)

- Введение
- Определение областей движения
 - Постановка задачи
 - Вычитание фона
 - Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора
 - Параметрические модели движения
 - Многоуровневое движение
 - Вычисление оптического потока

Содержание (2)

- Сопровождение объектов
 - Постановка задачи
 - Классификация методов сопровождения
 - Методы сопоставления ключевых точек
 - Методы сопровождения объектов, основанные на вычислении оптического потока
 - Применение метода «сдвиг среднего» к решению задачи сопровождения
 - Применение фильтра Кальмана для оценки положения объекта
 - Применение фильтра частиц к задаче сопровождения

ВВЕДЕНИЕ



Движение

- **Движение** – смещение одних объектов относительно других.

- В задачах компьютерного зрения выделяется несколько принципиально разных случаев движения:
 - Неподвижная камера, постоянный фон (системы видеонаблюдения).
 - Движущаяся камера, относительно постоянный фон (создание панорамы из последовательности изображений).
 - Движущаяся камера, постоянно изменяющийся фон (системы автоматического управления автономными роботами).



Основные группы задач

- ❑ ***Определение движения или областей движения***
 - Решение не требует анализа полученной информации о движении
- ❑ ***Поиск движущихся объектов***
 - Необходимо не просто определить область, в которой происходит движение, но и выделить положение движущихся объектов внутри этой области
- ❑ ***Определение трехмерных свойств объекта из набора его последовательных двумерных изображений***
 - Восстановление трехмерной модели объекта
 - Реконструкции трехмерной сцены



ОПРЕДЕЛЕНИЕ ОБЛАСТЕЙ ДВИЖЕНИЯ



Постановка задачи

- **Вход** – последовательность кадров I_1, I_2, \dots, I_N некоторого видео V :

$$I_k = \{I_k(x, y), 0 \leq x < width, 0 \leq y < height\}, k = \overline{1, N}, \quad (1)$$

где $width$ – ширина кадра, $height$ – высота кадра, $I_k(x, y)$ в общем случае представляет собой вектор фиксированной размерности.

- **Выход** – совокупность областей изображения для каждого кадра видео, в которых происходит движение одного или нескольких объектов:

$$M_k(x, y) = \begin{cases} 255, & (x, y) - \text{пиксель объекта} \\ 0, & (x, y) - \text{пиксель фона} \end{cases}, k = \overline{1, N} \quad (2)$$

Методы определения областей движения

❑ Вычитание фона

- ❑ Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора
- ❑ Параметрические модели движения
- ❑ Многоуровневое движение
- ❑ Вычисление оптического потока

Вычитание фона

- Процедура вычитания предполагает, что для данного видео построена **модель фона** (3):

$$F = \{F(x, y), 0 \leq x < width, 0 \leq y < height\}. \quad (3)$$

- Допустим, что $I_k(x, y), F(x, y) \in \{0, \dots, 255\}, k = \overline{1, N}$.

- **Процедуру вычитания** можно разбить на два этапа:
 - Вычитание фонового изображения из текущего кадра видео.

$$D_k(x, y) = \text{abs}(I_k(x, y) - F(x, y)), k = \overline{1, N} \quad (4)$$

- Отбор пикселей, принадлежащих фону и объекту, – построение бинарного изображения (маски).

$$M_k(x, y) = \begin{cases} 255, & D_k(x, y) \geq \tau \\ 0, & D_k(x, y) < \tau \end{cases}, k = \overline{1, N} \quad (5)$$

Нерекурсивные методы вычитания фона (1)

□ *Нерекурсивные методы вычитания фона:*

- Метод вычитания текущего и предыдущего кадра

$$\begin{aligned} D_k(x, y) &= \text{abs}(I_k(x, y) - F_k(x, y)) = \\ &= \text{abs}(I_k(x, y) - I_{k-1}(x, y)), k = \overline{2, N} \end{aligned} \quad (6)$$

- Метод усреднения определенного количества предшествующих кадров

$$F_k(x, y) = \frac{1}{s} \sum_{j=0}^{s-1} I_{k-j}(x, y) \quad (7)$$

- Метод определения медианы фиксированного количества предшествующих кадров

$$F_k(x, y) = \text{med}_{j=\overline{0, s-1}} \{I_{k-j}(x, y)\} \quad (8)$$

Нерекурсивные методы вычитания фона (2)

□ *Преимущества:*

- простота реализации и скорость обновления моделей фона при переходе от кадра к кадру.

□ *Недостатки:*

- Качество работы зависит от скорости объектов. Медленно перемещающиеся объекты, обнаруживаются плохо.
- Не дают качественный результат при изменении света, либо при наличии динамического фона (листва деревьев, струящаяся вода и т.п.). Чтобы сгладить влияние таких эффектов, используется α -смешивание:

$$F_k(x, y) = \alpha I_k(x, y) + (1 - \alpha)F_{k-1}(x, y) \quad (9)$$



Рекурсивные методы вычитания фона

□ *Рекурсивные методы вычитания фона:*

– Гистограммный метод

- Цветовое пространство разбивается на отдельные бины.
- Для каждого изображения в последовательности выполняется построение гистограммы.
- Пиксели, составляющие некоторый бин, принадлежат фону, если величина данного бина меньше фиксированного порогового значения, в противном случае, считается, что они принадлежат объекту.

– Представление фона смесью Гауссовых распределений

– Метод извлечения визуального фона (Visual Background Extractor, ViBe)



Представление фона смесью Гауссовых распределений. Модель фона (1)

- Считается, что для любого пикселя (x_0, y_0) изображения I_k известна история изменения его интенсивности/цвета на всех предшествующих кадрах $\{X_1, X_2, \dots, X_k\} = \{I_j(x_0, y_0), j = \overline{1, k}\}$.
- Вероятность того, что наблюдается значение X_k , представляется смесью из s Гауссовых распределений:

$$P(X_k) = \sum_{j=1}^s \omega_j^k N(X_k | \mu_j^k, \Sigma_j^k), \quad (10)$$

где ω_j^k – вес j -ого распределения Гаусса для кадра с номером k , μ_j^k – математическое ожидание, Σ_j^k – среднеквадратичное отклонение, $N(X_k | \mu_j^k, \Sigma_j^k)$ – функция плотности нормального распределения (11).

Представление фона смесью Гауссовых распределений. Модель фона (2)

$$N(X_k | \mu_j^k, \Sigma_j^k) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_j^k|^{\frac{1}{2}}} e^{-\frac{1}{2} (X_k - \mu_j^k)^T (\Sigma_j^k)^{-1} (X_k - \mu_j^k)} \quad (11)$$

- Предполагается, что компоненты цвета независимы и имеют одинаковое среднеквадратичное отклонение:

$$\Sigma_j^k = (\sigma_j^k)^2 E$$

где E – единичная матрица.

- Новое значение будет представляться одной из основных компонент построенной смеси Гауссовых распределений и использоваться для обновления параметров модели.

Представление фона смесью Гауссовых распределений. Модель фона (3)

- ❑ Распределения сортируются в порядке уменьшения величины $r_j^k = \frac{\omega_j^k}{\sigma_j^k}$.
- ❑ Сортировка предполагает, что пиксель фона отвечает распределению с **большим** весом и **малой** дисперсией.
- ❑ **Модель фона:** первые B^k распределений, удовлетворяющих условию (12), соответствуют распределению цвета фоновых пикселей.

$$B^k = \operatorname{argmin}_b \{ \sum_{j=1}^b \omega_j^k > T \}, \quad (12)$$

где T – некоторое пороговое значение, параметр модели.

Представление фона смесью Гауссовых распределений. Классификация пикселя (4)

- Когда приходит очередной кадр I_{k+1} , для каждого пикселя изображения выполняется тест.
- Для расстояния Махаланобиса проверяется условие:

$$\sqrt{(X_{k+1} - \mu_j^k)^T (\Sigma_j^k)^{-1} (X_{k+1} - \mu_j^k)} < 2,5\sigma_j^k \quad (13)$$

- Если нашлось соответствующее распределение Гаусса, то в зависимости от того, входит ли оно в группу из B^k распределений или нет, пиксель классифицируется как фоновый, либо как принадлежащий объекту.
- Если не обнаружилось ни одного распределения, удовлетворяющего условию (13), то считается, что пиксель принадлежит объекту.

Представление фона смесью Гауссовых распределений. Обновление модели фона (5)

- В зависимости от результата поиска соответствия, обновление выполняется по-разному.
- **Соответствие обнаружено.**
 - Пересчитываются весовые коэффициенты, составляющие смесь распределений, которым соответствует X_{k+1} , и параметры распределений.

$$\omega_j^{k+1} = (1 - \alpha)\omega_j^k + \alpha, \quad (14)$$

$$\mu_j^{k+1} = (1 - \rho)\mu_j^k + \rho X_{k+1}, \quad (15)$$

$$(\sigma_j^{k+1})^2 = (1 - \rho)(\sigma_j^k)^2 + \rho(X_{k+1} - \mu_j^{k+1})(X_{k+1} - \mu_j^{k+1})^T, \quad (16)$$

где α – заданная константа, $\rho = \alpha N(X_{k+1} | \mu_j^k, \Sigma_j^k)$.

Представление фона смесью Гауссовых распределений. Обновление модели фона (6)

- Для всех распределений, которым X_{k+1} не соответствует, параметры не изменяются, пересчитываются только коэффициенты ω_j^{k+1} .

$$\omega_j^{k+1} = (1 - \alpha)\omega_j^k \quad (17)$$

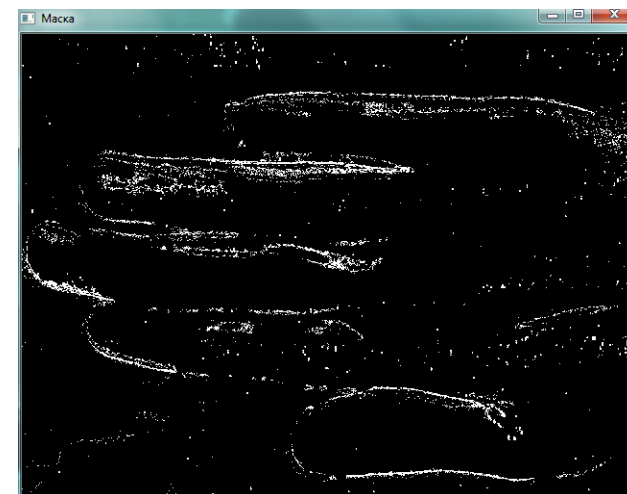
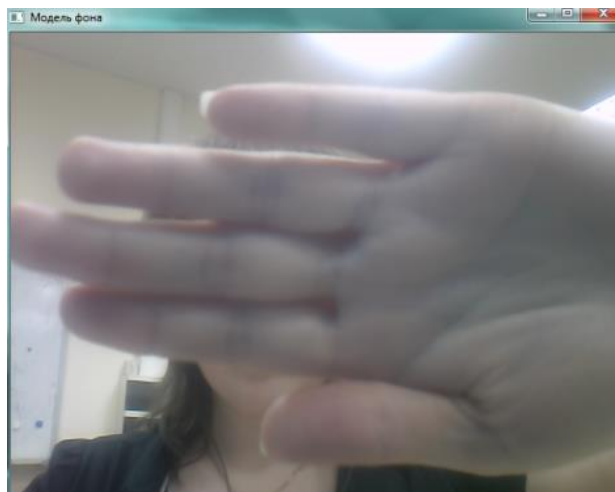
□ **Соответствие не найдено.**

- Крайнее в смысле введенного отношения порядка распределение Гаусса замещается распределением с новыми параметрами:
 - Математическое ожидание выбирается равным текущему значению цвета пикселя $\mu_s^{k+1} = X_{k+1}$.
 - Дисперсия $(\sigma_s^{k+1})^2$ устанавливается максимально возможной, а вес ω_s^{k+1} минимально допустимым.

Представление фона смесью Гауссовых распределений. Замечания (7)

- ❑ Количество распределений определяется сложностью фона и имеющимися вычислительными мощностями (в первоисточнике предлагается использовать значение в пределах от 3 до 5).
- ❑ Начальная инициализация параметров распределений может выполняться с использованием метода k-средних, либо EM-алгоритма (Expectation Maximization).

Представление фона смесью Гауссовых распределений. Пример применения (8)



Метод извлечения визуального фона (ViVe).

Модель фона (1)

- **Модель фона** на кадре с номером k представляется набором множеств

$$M^k(p) = \{v_1, v_2, \dots, v_N\}$$

для всех пикселей $p = (x, y)$, где v_i – интенсивность/цвет пикселя (в общем случае вектор).

Метод извлечения визуального фона (ViVe).

Классификация пикселя (2)

- Для классификации пикселя p в цветовом пространстве строится сфера $S_R(v(x))$ радиуса R и определяется количество векторов множества $M(p)$, которые попадают вовнутрь построенной сферы, $K = |S_R(v(p)) \cap M(p)|$.
- Если $K > T_{min}$, где T_{min} – фиксированное пороговое значение, то принимается, что пиксель p принадлежит фону, в противном случае, объекту.

Метод извлечения визуального фона (ViVe).

Инициализация (3)

- На начальном этапе необходимо выполнить инициализацию множеств $M^0(p)$ для всех пикселей p следующим образом:

$$M^0(p) = \{v^0(y), y \in N_G(p)\}, \quad (18)$$

где $N_G(p)$ – окрестность пикселя p размера 3×3 (9 клеток, включая текущий пиксель), y выбирается N раз случайным образом.

Метод извлечения визуального фона (ViVe).

Обновление модели фона (3)

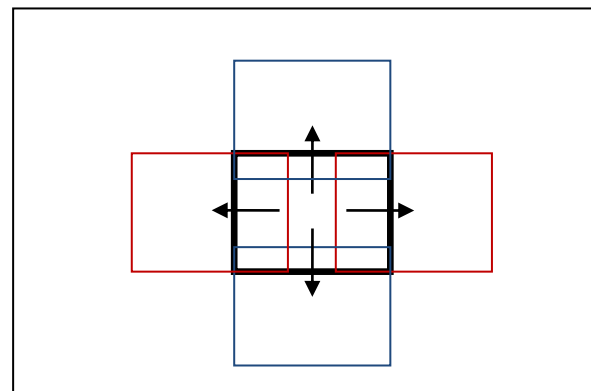
- Обновление модели фона для кадра I_k выполняется в два шага:
 - Если p проклассифицирован как пиксель фона, то из множества $M^k(p)$ случайно выбирается компонента, которая заменяется значением $v(p)$.
 - Случайным образом выбирается один соседний пиксель из окрестности $N_G(p)$, для которого выполняется предыдущий шаг.

Методы определения областей движения

- ❑ Вычитание фона
- ❑ **Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора**
- ❑ Параметрические модели движения
- ❑ Многоуровневое движение
- ❑ Вычисление оптического потока

Полный перебор

- ❑ Перебор всех возможных вариантов смещения изображений или их отдельных фрагментов (translational alignment).
- ❑ Необходимо выбрать метрику для оценки степени сходства фрагментов.
- ❑ Исходная задача определения движения может быть сведена к **минимизации функции ошибки** по всем возможным направлениям смещения.



Функции ошибки. Одноканальное изображение (1)

□ **Квадратичная функция ошибки.**

$$E(u, v) = \sum_i (I_k(x_i + u, y_i + v) - I_{k-1}(x_i, y_i))^2, \quad (19)$$

где (u, v) – вектор смещения изображения или фрагмента.

□ **Грубые оценочные функции ошибки.** $\rho(e)$ – функция, которая имеет меньшую степень роста по сравнению квадратичной функцией.

$$E(u, v) = \sum_i \rho(I_k(x_i + u, y_i + v) - I_{k-1}(x_i, y_i)) \quad (20)$$

Примером грубой функции ошибки может служить сумма абсолютных разностей (21).

$$E(u, v) = \sum_i |I_k(x_i + u, y_i + v) - I_{k-1}(x_i, y_i)| \quad (21)$$

Функции ошибки. Одноканальное изображение (2)

- **Взвешенная квадратичная функция ошибки.** Весовые коэффициенты – дискретные функции, которые принимают нулевые значения за пределами изображения.

$$E(u, v) = \sum_i \omega_{k-1}(x_i, y_i) \omega_k(x_i + u, y_i + v) \cdot (I_k(x_i + u, y_i + v) - I_{k-1}(x_i, y_i))^2 \quad (22)$$

- **Кросс-корреляция пары выровненных изображений.**

$$E(u, v) = \sum_i I_{k-1}(x_i, y_i) \cdot I_k(x_i + u, y_i + v) \quad (23)$$

Заметим, что при выборе кросс-корреляции в качестве функции ошибки выполняется не минимизация, а максимизация по всем возможным направлениям смещения.



Функции ошибки. Цветное изображение

- ❑ **1 вариант:** выполнить их конвертирование в полутоновые, либо ввести дополнительную сумму по числу каналов
- ❑ **2 вариант:** использовать одну из предложенных функций ошибки.

Ускорение перебора. Иерархическая схема (1)

- ❑ Конструируется пара пирамид для последовательно идущих изображений посредством масштабирования исходных изображений.
- ❑ Поиск выполняется от мелких изображений к более крупным:
 - Постепенно отсекаются направления смещения, в которых заведомо не происходит движение.
 - Если на некотором уровне обнаружился вектор смещения для определенного фрагмента, то на следующем – этот вектор используется для предсказания расположения соответствующего фрагмента.

Ускорение перебора. Применение БПФ (2)

- Метод опирается на тот факт, что модули образов Фурье исходного и смещенного сигналов связаны уравнением:

$$\begin{aligned} F(I_k(x_j + u, y_j + v)) &= F(I_k(x_i, y_i))e^{-i\left(\begin{bmatrix} u \\ v \end{bmatrix}, \bar{\omega}\right)} = \\ &= I_k(\bar{\omega})e^{-i\left(\begin{bmatrix} u \\ v \end{bmatrix}, \bar{\omega}\right)}, \end{aligned} \quad (24)$$

где $\bar{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix}$ – вектор частот двумерного БПФ, $I_k(\bar{\omega}) = F(I_k(x_i, y_i))$ – образ Фурье сигнала $I_k(x_i, y_i)$.

- Покажем применение на примере кросс-корреляции.

Ускорение перебора. Применение БПФ (3)

$$\begin{aligned} F(E(u, v)) &= F\left(\sum_i I_{k-1}(x_i, y_i) \cdot I_k(x_i + u, y_i + v)\right) = \\ &= F(I_{k-1}(u, v) * I_k(u, v)) = I_{k-1}(\overline{\omega}) I_k^*(\overline{\omega}) \end{aligned} \quad (25)$$

- ❑ Вычислить образы Фурье для исходной пары изображений с помощью БПФ (сложность составляет $O(NM \log NM)$ для изображения размера $N \times M$).
- ❑ Перемножить полученные образы (за время, не превышающее $O(N^2 M^2)$).
- ❑ Вычислить обратное преобразование Фурье.

Субпиксельная точность определения движения

- ❑ **Субпиксельная точность** определения смещения – смещение в дробных координатах.
- ❑ Один из наиболее известных подходов состоит в том, чтобы использовать **процедуру уточнения направления смещения**.
- ❑ Впервые была введена при разработке алгоритма сопровождения Лукаса-Канаде.
- ❑ Процедура основана на применении метода градиентного спуска для минимизации квадратичной функции ошибки в смещенной точке – функции энергии, которая приближается разложением в ряд Тейлора.

Уточнение направления движения (1)

$$\begin{aligned} E(u + \Delta u, v + \Delta v) &= \\ &= \sum_i (I_k(x_i + u + \Delta u, y_i + v + \Delta v) - I_{k-1}(x_i, y_i))^2 \approx \quad (26) \\ &\approx \sum_i \left(I_k(x_i + u, y_i + v) + \nabla I_k(x_i + u, y_i + v) \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} - I_{k-1}(x_i, y_i) \right)^2 \\ &= \sum_i \left(\nabla I_k(x_i + u, y_i + v) \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} + e_i \right)^2 = \end{aligned}$$

$$= \sum_i \left(J_k(x_i + u, y_i + v) \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} + e_i \right)^2 \rightarrow \min_{(u,v)} , \quad (27)$$

где $e_i = I_k(x_i + u, y_i + v) - I_{k-1}(x_i, y_i)$,

$$J_k(x_i + u, y_i + v) = \nabla I_k(x_i + u, y_i + v) = \left[\frac{\partial I_k}{\partial x}, \frac{\partial I_k}{\partial y} \right] \Big|_{(x_i+u, y_i+v)} .$$

Уточнение направления движения (2)

- Полученная задача минимизации (27) сводится к решению системы линейных алгебраических уравнений вида (28).

$$A \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = b, \quad (28)$$

где $A = \sum_i J_k^T(x_i + u, y_i + v) J_k(x_i + u, y_i + v)$,
 $b = -\sum_i e_i J_k^T(x_i + u, y_i + v)$.

- Эффективность решения задачи достигается за счет того, что для текущего изображения градиенты $J_k(x_i + u, y_i + v)$ в смещенной точке можно приближенно заменить градиентами предыдущего изображения $J_{k-1}(x_i, y_i)$ в исходной точке, т.е. $J_k(x_i + u, y_i + v) \approx J_{k-1}(x_i, y_i)$.

Методы определения областей движения

- ❑ Вычитание фона
- ❑ Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора
- ❑ **Параметрические модели движения**
- ❑ Многоуровневое движение
- ❑ Вычисление оптического потока

Параметрические модели движения

- ❑ Движение не описывается только линейным сдвигом!
- ❑ Поэтому рассматриваются пространственные поля смещений, и выполняется построение **параметрических моделей движения** (parametric motion).
- ❑ В параметрических моделях вместо постоянного вектора смещения $[u, v]^T$ рассматривается параметризованное поле смещений $x'(x, y; p)$, где p -вектор параметров.

Примеры параметрических моделей движения (1)

□ **Линейный сдвиг** (29).

$$x'(x, y; p) = \begin{bmatrix} E & p \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (29)$$

где $p = [p_1 \ p_2]$ – вектор сдвига.

□ **Поворот со сдвигом** (30).

$$x'(x, y; p) = \begin{bmatrix} \cos \varphi & -\sin \varphi & p_1 \\ \sin \varphi & \cos \varphi & p_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (30)$$

где $R = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix}$ – матрица поворота, $p = (p_1, p_2, \varphi)$ – набор параметров модели движения.



Примеры параметрических моделей движения (2)

□ *Преобразование подобия* (31).

$$x'(x, y; p) = \begin{bmatrix} a & -b & p_1 \\ b & a & p_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (31)$$

где $sR = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ – матрица поворота, умноженная на коэффициент масштабирования s , $p = (p_1, p_2, a, b)$ – набор параметров модели движения.

□ *Аффинное преобразование* (32).

$$x'(x, y; p) = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (32)$$

Примеры параметрических моделей движения (3)

- *Перспективная проекция или гомография* (33).

$$x'(x, y; p) = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H\bar{x} \quad (33)$$

- Аналогично процедуре уточнения направления, можно выполнить определение направления движения для любой параметрической модели, расписав квадратичную функцию ошибки с помощью разложения в ряд Тейлора по вектору параметров и выполнив ее минимизацию.

Методы определения областей движения

- ❑ Вычитание фона
- ❑ Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора
- ❑ Параметрические модели движения
- ❑ **Многоуровневое движение**
- ❑ Вычисление оптического потока

Многоуровневое движение

- ❑ Во многих случаях визуальное движение вызвано смещением небольшого количества объектов, находящихся на разной глубине изображения.
- ❑ Поэтому движение пикселей можно описать более эффективно, если сгруппировать их в слои, и отслеживать **многоуровневое движение** (layered motion) построенных слоев, например, с помощью параметрических моделей.

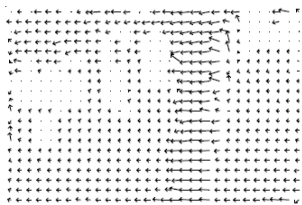
Как представить слой?

- Слой определяется набором из трех карт (матриц):
 - матрица интенсивности слоя (в компьютерной графике текстурная карта);
 - α -карта, которая определяет прозрачность слоя в каждой точке изображения;
 - карта скоростей описывает изменение положения точек с течением времени.

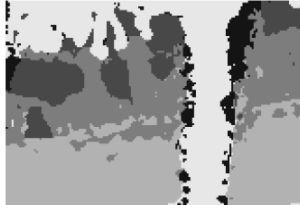
Как разбить множество пикселей на слои и определить движение каждого слоя?

- ❑ Оценивание движения с использованием оптического потока и аффинной модели для набора неперекрывающихся блоков (a), полученных в результате разбиения исходного изображения.
- ❑ Кластеризация полученных оценок с помощью метода k -средних (b). Каждый кластер определяет сегмент, отвечающий некоторому слою изображения.
- ❑ Применение медианной фильтрации для получения смешанных слоев, устойчивых к незначительным изменениям интенсивности, а также для выявления перекрытий между слоями (c).

Пример выделения слоев



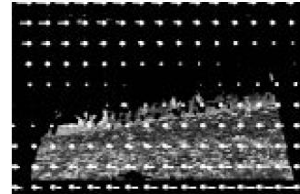
а) Оптический поток



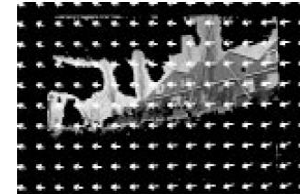
б) Начальное множество слоев



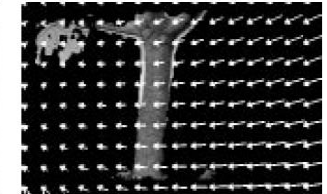
с) Результирующее множество слоев



д) Слой, отвечающий клумбе



е) Слой, отвечающий строению



ф) Слой, отвечающий дереву



г) Кадр 0



h) Кадр 15



и) Кадр 30



j) Кадр 0



к) Кадр 15



л) Кадр 30

Кадры, восстановленные на основании послойного представления

Кадры, восстановленные на основании послойного представления, исключая слой, содержащий дерево

Wang, J. Y. A., Adelson E. H. Representing moving images with layers // IEEE Transactions on Image Processing. 1994. 3(5). P. 625–638. [http://persci.mit.edu/pub_pdfs/wang_tr279.pdf].

Методы определения областей движения

- ❑ Вычитание фона
- ❑ Полный перебор допустимых вариантов смещения изображений или их фрагментов. Возможные способы ускорения перебора
- ❑ Параметрические модели движения
- ❑ Многоуровневое движение
- ❑ **Вычисление оптического потока**



Вычисление оптического потока (1)

- Другой распространенный подход к решению задачи детектирования областей движения – **вычисление оптического потока** (optical flow).
- Условия применимости:
 - яркость каждой точки объекта не изменяется с течением времени;
 - ближайшие точки, принадлежащие одному объекту, в плоскости изображения двигаются с похожей скоростью.

Вычисление оптического потока (2)

- Пусть имеется непрерывно изменяющееся во времени изображение.
- $I_k(x, y)$ – яркость пикселя с координатами (x, y) в момент времени k .
- Считая динамическое изображение функцией положения и времени, яркость пикселя в следующий момент времени можно выразить с помощью разложения в ряд Тейлора:

$$I_{k+\Delta k}(x + \Delta x, y + \Delta y) = I_k(x, y) + \\ + \frac{\partial I_k}{\partial x} \Delta x + \frac{\partial I_k}{\partial y} \Delta y + \frac{\partial I_k}{\partial k} \Delta k + O(\partial^2), \quad (38)$$

где $\frac{\partial I_k}{\partial x}, \frac{\partial I_k}{\partial y}, \frac{\partial I_k}{\partial k}$ – частные производные функции I_k .

Вычисление оптического потока (3)

- Предполагается, что за небольшой промежуток времени пиксель смещается незначительно, т.е. $(\Delta x, \Delta y) \rightarrow 0$ при $\Delta k \rightarrow 0$, значит можно считать, что яркость почти не изменяется, поэтому остаточным членом ряда можно пренебречь:

$$I_{k+\Delta k}(x + \Delta x, y + \Delta y) = I_k(x, y)$$

- **Уравнение оптического потока:**

$$-\frac{\partial I_k}{\partial k} = \frac{\partial I_k}{\partial x} \frac{dx}{dk} + \frac{\partial I_k}{\partial y} \frac{dy}{dk} \quad (39)$$

- **Замечание:** под частными производными в силу дискретности представления изображения понимается центральный разностный оператор.



Вычисление оптического потока (4)

□ Цель – построить вектор скорости $c = \left(\frac{dx}{dk}, \frac{dy}{dk} \right) = (u, v)$.

□ Для полноты постановки задачи вводится условие гладкости изменения скорости:

$$u_x^2 + u_y^2 = 0 \text{ и } v_x^2 + v_y^2 = 0$$

□ Исходная задача сводится к задаче минимизации квадратичной ошибки $\left(\frac{\partial I_k}{\partial x} u + \frac{\partial I_k}{\partial y} v + \frac{\partial I_k}{\partial k} \right)^2$ при наличии ограничений в виде равенств (задача математического программирования).

□ Процедура минимизации применяется к каждому пикселю текущего изображения => поля векторов смещения.

СОПРОВОЖДЕНИЕ ОБЪЕКТОВ



Постановка задачи (1)

- **Сопровождение** (трекинг) движущихся объектов – это один из составляющих компонентов многих систем реального времени таких, как системы слежения, анализа видео и других.
- **Вход** – последовательность изображений I_1, I_2, \dots, I_N .
- **Выход** – набор траекторий движения целевых объектов на входной последовательности кадров.

Постановка задачи (2)

- ❑ **Положение объекта** на изображении с номером k обозначается P_k .
- ❑ **Траекторией** движения объекта называется последовательность его положений $P_s, P_{s+1}, \dots, P_{s+l-1}$, где s – номер первого кадра, на котором был обнаружен объект, l – количество кадров последовательности, где наблюдается объект.
- ❑ **Замечание:** в зависимости от метода сопровождения положение объекта может определяться по-разному (координаты и размер сторон окаймляющего прямоугольника, координаты центра масс контура и т.п.).

Сопровождение объектов

❑ Классификация методов сопровождения

- ❑ Методы сопоставления ключевых точек
- ❑ Методы сопровождения объектов, основанные на вычислении оптического потока
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения
- ❑ Применение фильтра Кальмана для оценки положения объекта
- ❑ Применение фильтра частиц к задаче сопровождения

Классификация методов сопровождения (1)

□ **Методы сопровождения особых точек** (point tracking).

– Детерминистские методы

- Используют качественные эвристики движения (небольшое изменение скорости, неизменность расстояния в трехмерном пространстве между парой точек, принадлежащих объекту).
- Примеры: методы, основанные на вычислении плотного и разреженного оптического потока, методы сопоставления (matching) дескрипторов ключевых точек.

– Вероятностные методы

- Используют подход, основанный на понятии пространства состояний.
- Примеры: методы на базе фильтра Кальмана и фильтра частиц (particle filter).

Классификация методов сопровождения (2)

- **Методы сопровождения компонент** (kernel tracking).
 - *Компонента* – форма объекта (шаблон прямоугольной или овальной формы, трехмерная модель объекта, спроецированной на плоскость изображения).
 - Методы данной группы применяются, если движение определяется обычным смещением, поворотом или аффинным преобразованием.
 - Трекинг компонент – итеративная процедура локализации, основанная на максимизации некоторого критерия подобия. На практике реализуется с использованием сдвига среднего (mean shift) и его непрерывной модификации (CAM Shift).

Классификация методов сопровождения (3)

- **Методы сопровождения силуэта** (silhouette tracking).
Силуэт может быть задан контуром, либо набором связанных простых геометрических примитивов.
 - Методы сопоставления и сопровождения фрагментов изображения, содержащих объект
 - При сопоставлении естественным образом должна быть введена мера сходства пары областей, ограниченных контуром.
 - Определение положения фрагмента выполняется посредством вычисления оптического потока для внутренних точек области.
 - Методы сопровождения контура
 - Использование моделей пространства состояний (по типу фильтра Кальмана).
 - Минимизации функции энергии контура.

Сопровождение объектов

- ❑ Классификация методов сопровождения
- ❑ **Методы сопоставления ключевых точек**
- ❑ Методы сопровождения объектов, основанные на вычислении оптического потока
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения
- ❑ Применение фильтра Кальмана для оценки положения объекта
- ❑ Применение фильтра частиц к задаче сопровождения

Методы сопоставления ключевых точек.

Общая схема

- Схема сопровождения особых точек может быть представлена в виде последовательности действий:
 - Поиск особых точек на предыдущем и текущем кадрах посредством выбранного детектора (LoG, DoG, SURF, SIFT, ORB и др.).
 - Вычисление дескрипторов для полученного набора точек (SURF, SIFT, GLOH, DAISY, BRIEF и т.д.) – n -мерных векторов-описателей.
 - **Сопоставление** (matching) дескрипторов, полученных на текущем и предыдущем кадре. Необходимо построить преобразование точек n -мерного пространства дескрипторов.
 - Отсечение ложных срабатываний.

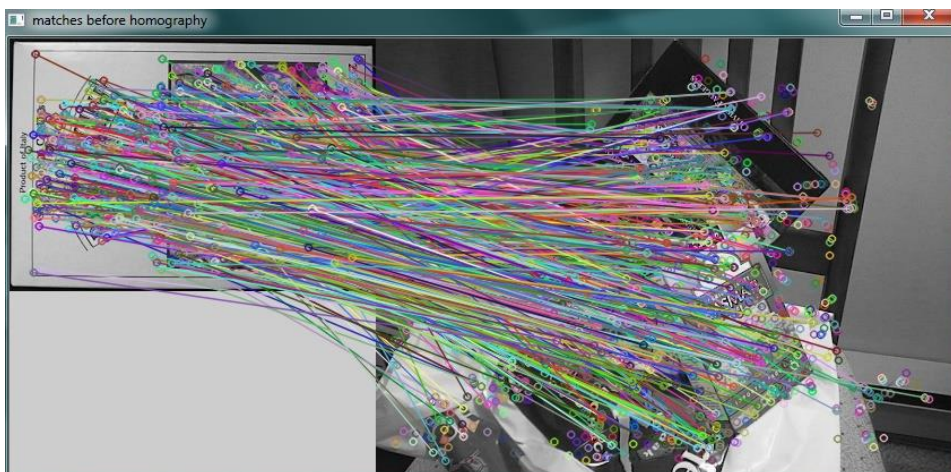


Методы сопоставления ключевых точек.

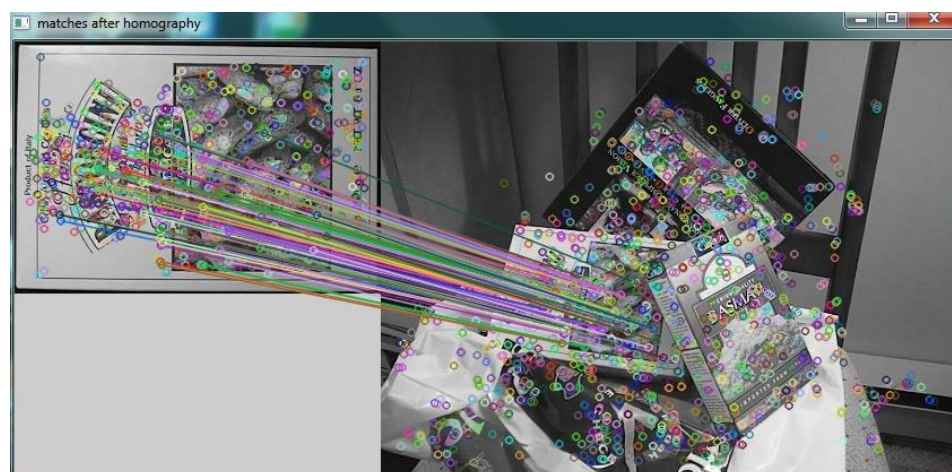
Пример



Пример исходного изображения объекта (слева)
и объекта внутри некоторой сцены (справа)



Набор соответствий до применения
алгоритма RANSAC



Набор соответствий после
применения алгоритма RANSAC

Алгоритм RANSAC

- ❑ **RANSAC** – это общий метод, который используется для оценки параметров модели на основании случайных выборок.
- ❑ При сопоставлении модель представляет собой матрицу преобразования (гомография).
- ❑ **Вход** – два множества дескрипторов, полученных на предыдущем и текущем изображении.
- ❑ **Выход** – наилучшая матрица гомографии.

Схема работы алгоритма RANSAC (1)

□ ***Выбор точек и построение параметров модели:***

- Случайный выбор без повторений наборов фиксированного размера из дескрипторов.
- На основании полученных наборов строится матрица преобразования.

□ ***Проверка построенной модели:***

- Для каждого дескриптора предыдущего кадра находится проекция на текущем кадре и выполняется поиск наиболее близкого дескриптора из множества дескрипторов текущего кадра.
- Дескриптор помечается как выброс, если расстояние между проекцией и соответствующим дескриптором текущего изображения больше некоторого порога.

Схема работы алгоритма RANSAC (2)

□ *Замещение модели:*

- После проверки всех точек проверяется, является ли построенная модель лучшей среди набора предшествующих моделей.

Сопровождение объектов

- ❑ Классификация методов сопровождения
- ❑ Методы сопоставления ключевых точек
- ❑ **Методы сопровождения объектов, основанные на вычислении оптического потока**
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения
- ❑ Применение фильтра Кальмана для оценки положения объекта
- ❑ Применение фильтра частиц к задаче сопровождения

Методы сопровождения, основанные на вычислении оптического потока (1)

□ *Метод Хорна*

- Вычисление плотного оптического потока.
- Решение системы дифференциальных уравнений, полученных в задаче математического программирования с помощью итерационных методов для построения компонент вектора скорости в каждой точке изображения.

□ *Алгоритмы сопоставления блоков*

- Предыдущее и текущее изображения разбиваются на блоки (квадратные и перекрывающиеся).
- Определяется движение этих блоков посредством сопоставления. Изображение поля скоростей обычно имеет меньшее разрешение по сравнению с исходным.



Методы сопровождения, основанные на вычислении оптического потока (2)

□ *Алгоритм Лукаса-Канаде*

- Основан на вычислении *разреженного оптического потока* – построение векторного поля скоростей для выделенного набора точек.
- Задача слежения без учета аффинных искажений с помощью данного алгоритма сводится к поиску оптического потока в особых точках.
- Замечание: впоследствии появились модификации данного алгоритма Томаши-Канаде (Tomasi-Kanade) и Ши-Томаши-Канаде (Shi-Tomasi-Kanade). Трекер Ши-Томаши-Канаде впервые учитывает аффинные искажения окрестных точек.

Сопровождение объектов

- ❑ Классификация методов сопровождения
- ❑ Методы сопоставления ключевых точек
- ❑ Методы сопровождения объектов, основанные на вычислении оптического потока
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения**
- ❑ Применение фильтра Кальмана для оценки положения объекта
- ❑ Применение фильтра частиц к задаче сопровождения

Применение сдвига среднего

- ❑ Идея методов состоит в том, что для каждой особой точки (в общем случае, для каждого объекта) выбирается окно поиска, вычисляется центр масс распределения интенсивностей (гистограммы).
- ❑ Соответственно центр окна смещается в центр масс, который представляет собой положение точки на текущем кадре.
- ❑ Определение положения точки на последующих кадрах сводится к применению очередного шага метода «сдвига среднего». Метод останавливается, когда центр масс перестает смещаться.

Сопровождение объектов

- ❑ Классификация методов сопровождения
- ❑ Методы сопоставления ключевых точек
- ❑ Методы сопровождения объектов, основанные на вычислении оптического потока
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения
- ❑ Применение фильтра Кальмана для оценки положения объекта**
- ❑ Применение фильтра частиц к задаче сопровождения

Задача сопровождения (1)

- ❑ Задачу сопровождения можно рассматривать как хорошо изученную проблему теории управления.
- ❑ Необходимо оценить состояние системы на основании последовательности зашумленных измерений.
- ❑ Формально имеется модель объекта, которая наблюдается в зашумленном пространстве сцены:
 x – **модель** объекта, z – **наблюдение** объекта.
- ❑ В общем случае x, z – вектора признаков, которые могут иметь разную размерность.

Задача сопровождения (2)

- В процессе сопровождения объекта модели и наблюдения могут использоваться двумя способами:
 - Последовательность наблюдений z_1, z_2, \dots применяется для уточнения базовой модели x . Заметим, что модель с течением времени также может изменяться, поэтому каждое новое наблюдение z_k может давать новую оценку модели x_k .
 - Построенная оценка модели x_k используется для предсказания модели x_{k+1} и наблюдения z_{k+1} .

Механизм обратной связи

- В результате получаем механизм обратной связи:
наблюдаем $z_k \rightarrow$
оцениваем $x_k \rightarrow$
предсказываем x_{k+1} и $z_{k+1} \rightarrow$
обновляем x_{k+1} на основании z_{k+1} .
- Такой механизм лежит в основе методов сопровождения, основанных на фильтрах Кальмана и фильтрах частиц.

Фильтр Кальмана. Предположения (1)

□ Предположения:

- Система является линейной (наблюдение являются линейной функции состояния).
- Шумы описываются Гауссовым распределением с математическим ожиданием, равным нулю.

Фильтр Кальмана. Модель обратной связи (2)

$$x_{k+1} = F_k x_k + w_k, \quad (40)$$

$$z_k = H_k x_k + v_k, \quad (41)$$

где F_k – матрица преобразования состояния системы;

w_k – “белый” шум с нормальным распределением $N(0, Q_k)$ с математическим ожиданием, равным 0, и матрицей ковариации Q_k ;

H_k – матрица связи модели и наблюдения;

v_k – “белый” шум с нормальным распределением $N(0, R_k)$.

Фильтр Кальмана. Пример модели для случая равномерного распределения (1)

- Состояние системы определяется положением и скоростью точки:

$$x_k = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix}_k \quad (42)$$

- Матрица преобразования состояния $F_k = F$, и, исходя из физических соображений.

$$F = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (43)$$

Фильтр Кальмана. Пример модели для случая равномерного распределения (2)

- Поскольку при сопровождении камера регистрирует только координаты положения объекта, то $z_k = \begin{bmatrix} z_x \\ z_y \end{bmatrix}_k$.

Поэтому матрицу связи наблюдения и предсказания $H_k = H$ можно записать в соответствии с (44).

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (44)$$

Обобщенные уравнения для обновления состояния и модели (1)

- Идея состоит в том, что сначала строится априорная оценка \tilde{x}_k^- состояния согласно (40):

$$\tilde{x}_k^- = F_{k-1}x_{k-1} + w_{k-1}. \quad (45)$$

- Затем с использованием построенной оценки вычисляется наблюдение

$$z_k = H_k \tilde{x}_k^- + v_k. \quad (46)$$

- Также можно определить апостериорную оценку \tilde{x}_k^+ состояния, которая строится после наблюдения. Введем ошибки e_k^- (47) и e_k^+ (48), связанные с каждой оценкой состояния.

$$e_k^- = x_k - \tilde{x}_k^- \quad (47)$$

$$e_k^+ = x_k - \tilde{x}_k^+ \quad (48)$$

Обобщенные уравнения для обновления состояния и модели (2)

- ❑ Фильтр Кальмана оперирует разностью $z_k - H_k \tilde{x}_k^-$, в которую вносит вклад ошибка e_k^- и случайный шум v_k .
- ❑ В идеальном случае шум отсутствует и оценка состояния идеальна, поэтому указанная разность обращается в ноль.
- ❑ Задача состоит в том, чтобы построить **матричный коэффициент Кальмана** K_k с целью обновления апостериорной оценки согласно (49). Отсюда, если K_k известен, то известен закон получения обновленной модели x_k .

$$\tilde{x}_k^+ = \tilde{x}_k^- + K_k(z_k - H_k \tilde{x}_k^-) \quad (49)$$

Обобщенные уравнения для обновления состояния и модели (3)

- Уравнения (47) – (49) позволяют получить выражение связи априорной и апостериорной ошибки (50).

$$\begin{aligned} e_k^+ &= x_k - \tilde{x}_k^+ = x_k - ((E - K_k H_k) \tilde{x}_k^- - K_k z_k) = \\ &= x_k - (E - K_k H_k) \tilde{x}_k^- + K_k (H_k x_k + v_k) = \\ &= (E - K_k H_k) e_k^- + K_k v_k \end{aligned} \quad (50)$$

- Из определения матрицы ковариации пары случайных величин следуют выражения (51).

$$P_k^- = M(e_k^- e_k^{-T}), \quad P_k^+ = M(e_k^+ e_k^{+T}), \quad R_k = M(v_k v_k^T) \quad (51)$$

- Вследствие независимости ошибок e_k^- и e_k^+ получаем нулевую ковариацию для величин e_k^- и v_k (52).

$$M(e_k^- v_k^T) = M(v_k e_k^{-T}) = 0 \quad (52)$$

Обобщенные уравнения для обновления состояния и модели (4)

- Прделав несложные преобразования (53) (умножение (50) на e_k^{+T} и определение математического ожидания от полученного выражения), можно перейти к уравнению с матрицами ковариации (54).

$$\begin{aligned} e_k^+ e_k^{+T} &= (E - K_k H_k) \\ &= (E - K_k H_k) e_k^- e_k^{-T} (E - K_k H_k)^T + (E - K_k H_k) e_k^- v_k^T K_k^T \\ &\quad + K_k v_k e_k^{-T} (E - K_k H_k)^T + K_k v_k v_k^T K_k^T \end{aligned} \quad (53)$$

$$P_k^+ = (E - K_k H_k) P_k^- (E - K_k H_k)^T + K_k R_k K_k^T \quad (54)$$

- Отсюда получаем, что матрицу K_k необходимо выбрать так, чтобы сумма диагональных элементов (**след**) матрицы P_k^+ был минимален.

$$trace(P_k^+) \rightarrow \min_{K_k} \quad (55)$$

Обобщенные уравнения для обновления состояния и модели (5)

- Решение данной задачи определяется посредством дифференцирования функции $trace(P_k^+)$ по K_k (56).

$$-2(E - K_k H_k) P_k^- H_k^T + 2K_k R_k = 0 \quad (56)$$

- Как следствие, получаем формулу (57) для вычисления матрицы K_k .

$$K_k = P_k^- H_k^T (R_k + H_k P_k^- H_k^T)^{-1}, \quad (57)$$

где

$$P_k^- = F_k P_{k-1}^+ F_k^T + Q_{k-1}. \quad (58)$$

$$P_k^+ = (E - K_k H_k) P_k^- \quad (59)$$

Схема работы фильтра Кальмана

- **Предсказание.** Предполагает вычисление априорной оценки состояния \tilde{x}_k^- (45) и наблюдения z_k (46).
- **Коррекция.** Включает определение матричного коэффициента Кальмана K_k (57) и построение апостериорной оценки состояния \tilde{x}_k^+ согласно (49).

Сопровождение объектов

- ❑ Классификация методов сопровождения
- ❑ Методы сопоставления ключевых точек
- ❑ Методы сопровождения объектов, основанные на вычислении оптического потока
- ❑ Применение метода «сдвиг среднего» к решению задачи сопровождения
- ❑ Применение фильтра Кальмана для оценки положения объекта
- ❑ Применение фильтра частиц к задаче сопровождения**

Фильтры частиц

- ❑ Для некоторых практических задач, чтобы получить более точную оценку состояния системы, необходимо уйти от предположения, что шум имеет Гауссово распределение.
- ❑ Вводится понятие **мультимодального распределения** шума, а для моделирования подобных систем используются фильтры частиц.
- ❑ Мультимодальное распределение зачастую представляется смесью нескольких распределений.
- ❑ **Фильтры частиц** являются более общим подходом к решению задачи сопровождения с применением вероятностных методов.

Алгоритм CONDENSATION. Условия и предположения (1)

- Предположим, что система может находиться в состояниях $X_t = \{x_1, x_2, \dots, x_t\}$ в момент времени t .
- Так же, как и для фильтра Кальмана, последовательность наблюдений будем обозначать $Z_t = \{z_1, z_2, \dots, z_t\}$.
- Наряду с этим введем предположение о том, что состояние x_t зависит только от предыдущего состояния x_{t-1} — условие Марковской цепи.

Алгоритм CONDENSATION. Цель (2)

- ❑ Техники фильтрации частиц представляют распределение вероятности в виде коллекции взвешенных выборок – частиц, появление которых регулируется посредством введения весов.
- ❑ Тогда множество S_t определяет функцию плотности вероятности для состояния x_t при заданном наборе наблюдений Z_t .
- ❑ S_t задает приближенное распределение $p(x_t|Z_t)$.

$$S_t = \{(s_i^t, \pi_i^t), i = \overline{1, N}, \sum_{i=1}^N \pi_i^t = 1\} \quad (60)$$

- ❑ Задача состоит в том, чтобы построить **метод восстановления множества S_t на основании S_{t-1}** .



Алгоритм CONDENSATION. Схема алгоритма (3)

- Пусть коллекция взвешенных выборок в момент времени $t - 1$ построена (61).

$$S_{t-1} = \{(s_i^{t-1}, \pi_i^{t-1}), i = \overline{1, N}, \sum_{i=1}^N \pi_i^{t-1} = 1\} \quad (61)$$

Дополнительно вычислим интегральные веса:

$$c_i = c_{i-1} + \pi_i^{t-1}, i = \overline{1, N}, c_0 = 0 \quad (62)$$

- Определим n -ый экземпляр выборки S_t . Для этого случайным образом выберем число r из отрезка $[0, 1]$ и вычислим $j = \arg \min_i \{c_i > r\}$. Отсюда получаем текущую оценку состояния s_j^{t-1} .

Алгоритм CONDENSATION. Схема алгоритма (4)

- Выполним **предсказание** следующего состояния. Предсказание (63) выполняется аналогично фильтру Кальмана (40), разница лишь в том, что нет ограничений, связанных с линейностью системы и видом распределения шума.

$$s_n^t = F_{t-1} s_j^{t-1} + w_{t-1} \quad (63)$$

- Выполним **коррекцию**. Используя текущее наблюдение z_t и его распределение, необходимо установить вес полученного экземпляра согласно (64).

$$\pi_n^t = p(z_t | x_t = s_n^t) \quad (64)$$

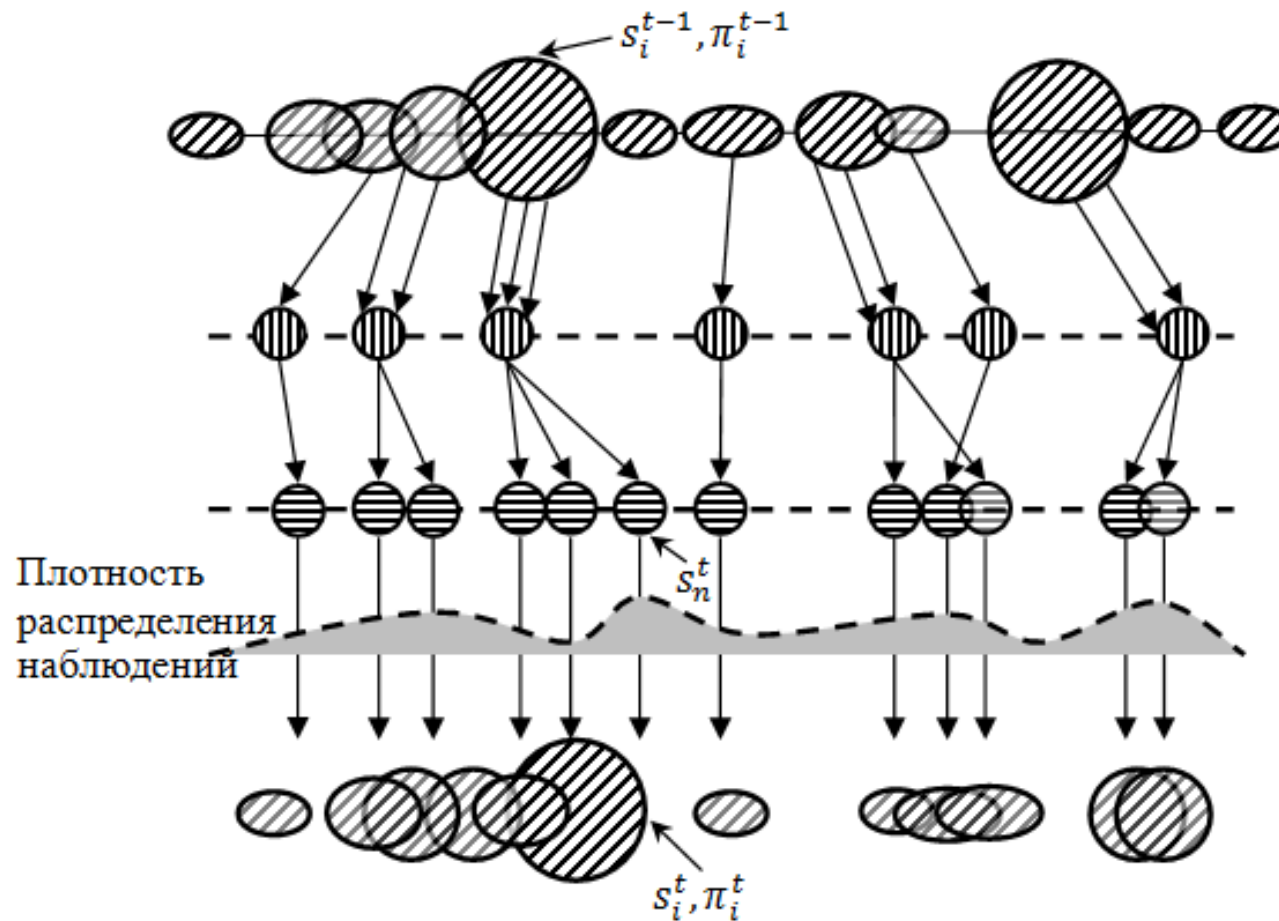
- Построим множество частиц S_t , повторив шаги 2 – 4 N -раз.

Алгоритм CONDENSATION. Схема алгоритма (5)

- Нормализуем последовательность весов π_i^t так, чтобы $\sum_{i=1}^N \pi_i^t = 1$.
- Вычислим наилучшую оценку для состояния x_t , например, как линейную свертку полученного набора экземпляров выборки (65). Таким образом, фактически определим некоторую среднюю частицу.

$$x_t = \sum_{i=1}^N \pi_i^t s_i^t \quad (65)$$

Графическая интерпретация алгоритма CONDENSATION



Контрольные вопросы (1)

- ❑ В каких случаях схема вычитания фона в задаче определения движения работает неэффективно?
- ❑ Докажите, что формула (25) справедлива, т.е. образ Фурье от корреляции пары функций равен произведению образу Фурье первой функции на комплексно-сопряженный образ второй функции.
- ❑ Основное свойство аффинного преобразования?
- ❑ Приведите примеры задач, в которых используется модель перспективной проекции (гомография).

Контрольные вопросы (2)

- ❑ Докажите справедливость формул (58) и (59).
- ❑ Получите формулы вычисления матричного коэффициента Кальмана для случая равномерного движения.
- ❑ Приведите схему построения фильтра Кальмана в случае одномерного состояния и наблюдения при некоторых фиксированных значениях $F_k = F = \text{const}$ и $H_k = H = \text{const}$.

Авторский коллектив

- ❑ Кустикова Валентина Дмитриевна,
ассистент кафедры
Математического обеспечения ЭВМ факультета ВМК ННГУ
valentina.kustikova@gmail.com

