

УДК 004.932.72'1

Подходы к отслеживанию объектов в видеопотоке. Применение фильтра частиц в задаче отслеживания

*Фроловская Е. А., студент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Рудаков И.В., к.т.н, доцент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана
irudakov@bmstu.ru*

Введение

Визуальное отслеживание заключается в последовательном определении местоположения целевого объекта на каждом кадре видеопотока. Эта задача находит практическое применение во многих областях, так или иначе связанных с обработкой видеозаписей для получения из них некоторой информации. Примерами систем, в которых применяются технологии отслеживания, являются системы человеко-машинного взаимодействия, видеонаблюдения, анализа спортивных матчей, появившиеся в недавнее время системы «умного дома», системы дополненной реальности и многие-многие другие.

Несмотря на значительное количество научных работ и исследований, проводимых в данной области, точное и устойчивое к ошибкам отслеживание объектов на видео остается сложной проблемой. Небольшие размеры объектов, меняющийся фон, резкие ускорения и смены траектории движения объектов, частичные или полные перекрытия, изменение масштаба отслеживаемой цели при ее приближении или удалении от камеры являются основными трудностями на пути к построению точной траектории перемещения объекта на кадрах видеозаписи.

Данная работа посвящена рассмотрению различных подходов к визуальному отслеживанию объектов, в особенности получившей в последнее время широкое распространение технологии фильтра частиц (англ. particle filter), предоставляющей широкие возможности и гибкий инструментарий для отслеживания деформируемых объектов в сложных с точки зрения данной задачи условиях. Статья построена следующим образом. В первой главе приводится краткий обзор существующих подходов к визуальному отслеживанию, выделяется место фильтра частиц среди них. Далее дается формальное

описание основных принципов, лежащих в основе работы фильтра частиц. В третьей главе рассматриваются различные методы отслеживания, основанные на фильтре частиц.

1. Подходы к отслеживанию объектов в видеопотоке

Результаты визуального отслеживания сильно зависят от эффективности определения положения объекта на отдельно взятом кадре, что делает задачи отслеживания и распознавания объекта тесно связанными между собой. На этой связи основаны два общих подхода к отслеживанию объектов: отслеживание с помощью распознавания и распознавание с помощью отслеживания.

В первом подходе ключевую роль играют алгоритмы распознавания. С их помощью определяются новые объекты, появляющиеся в сцене, и инициализируются трекары. В этом случае процесс отслеживания может быть сведен к задаче сопоставления объектов, распознанных на текущем кадре, с объектами, выявленными на предыдущем кадре.

В качестве примера алгоритма, реализующего данный подход, можно привести алгоритм AdaBoost, описанный в работе [Ошибка! Источник ссылки не найден.]. Для представления объекта авторы работы используют признаки Хаара (англ. Haar-like features), гистограммы направленных градиентов (англ. Histogram of Oriented Gradients, HOG) и локальные бинарные шаблоны (англ. Local Binary Patterns, LBP). Задача сопоставления объектов на соседних кадрах решается с помощью процесса сопоставления с эталоном (англ. template matching) и бинарного классификатора.

Необходимо отметить, что на алгоритмы, реализующие отслеживание путем распознавания, накладываются два основных ограничения: во-первых, отслеживаемый объект должен иметь отличительные внешние характеристики (цвет, текстура и т. п.), а во-вторых, не должен претерпевать значительных изменений на соседних кадрах. Таким образом, эти алгоритмы неэффективны в тех случаях, когда нельзя заранее предсказать визуальные характеристики цели, например, при быстрых перемещениях объекта, когда его внешний вид также меняется достаточно быстро. Данный класс алгоритмов не подходит также для отслеживания целей небольших размеров и целей, не имеющих достаточных отличительных признаков.

С некоторыми из описанных ограничений справляется второй подход: распознавание на основе отслеживания. В этом подходе определение возможных областей, занимаемых объектами, и установление соответствий между ними выполняется совместно путем итеративного обновления положения объекта и информации о занимаемой им области на основе данных, полученных на предыдущих кадрах. Обычно используется слабая

низкоуровневая модель внешнего представления объекта. На каждом шаге вычисляется функция плотности распределения вероятности положения объекта. Алгоритмы, реализующие данный подход, делятся на два основных класса: детерминированные и стохастические (вероятностные).

Наиболее известными представителями класса детерминированных алгоритмов отслеживания являются сдвиг среднего (англ. Mean Shift) и непрерывно адаптивный сдвиг среднего (англ. Continuously Adaptive Mean-shift, CAM-shift). Основная идея заключается в задании для каждого кадра весовой функции, такой, что искомое положение объекта соотносится с ее минимумом или максимумом, и сведении задачи отслеживания к оптимизации данной функции. Главная проблема такого подхода заключается в том, что оптимизируемая функция может принять мультимодальный вид (например, если в кадре несколько похожих объектов, или если внешний вид объекта претерпел значительные изменения с момента начала отслеживания), и в этом случае определение корректного положения объекта становится затруднительным.

Стохастические (вероятностные) алгоритмы отслеживания основаны на теории байесовского оценивания. В них задача отслеживания сводится к итеративной оценке функции плотности распределения вероятности вектора состояния объекта, производимой на основе измерения его значений на последовательных кадрах. Вектор состояния объекта может описывать его положение, размер, скорость, цвет или другие характеристики. Основные трудности в таком подходе связаны с процессом измерения значения вектора состояния. Этот процесс чувствителен к шумовым воздействиям, что может негативно повлиять на получаемые значения вектора. Чтобы получить фактическое значение вектора состояния объекта по зашумленным наблюдениям, необходимо применить процесс фильтрации.

Широко известным представителем данного класса отслеживателей является фильтр Кальмана (англ. Kalman Filter). Он использует предположение, что функция плотности распределения вероятности вектора состояний является Гауссовой, что позволяет получить аналитическое решение ее оценки. К сожалению, данная гипотеза во многих случаях не верна. Для решения этой проблемы были разработаны модификации данного метода: расширенный фильтр Кальмана (англ., Extended Kalman Filter, EKF), применяющий процесс линеаризации, и ансцентный фильтр Кальмана¹ (англ. Unscented Kalman Filter, UKF), использующий детерминированные выборки. Они относятся к категории параметрических

¹ В названии алгоритма приведена транслитерация оригинального названия, поскольку устоявшегося термина в русском языке пока нет, а дословный перевод не имеет смысла (unscented – «без запаха»). Также данный алгоритм иногда называют сигма-точечным фильтром Кальмана (англ. Sigma-Point Kalman Filter)

методов, работающих с нелинейными функциями плотности распределения [**Ошибка! Источник ссылки не найден.**]. Однако в большинстве своем эти решения показывают удовлетворительные результаты только при задании строгих ограничений на область применения алгоритма. Обычно они требуют наличия большого количества параметров, настройка которых поможет адаптировать алгоритм для решения конкретной задачи. Но и в этом случае остается проблема обработки длинных видеопоследовательностей и ситуаций с резкими изменениями формы и внешнего вида объекта.

К непараметрическим техникам борьбы с нелинейностью процессов измерения состояния объекта относятся методы Монте-Карло, в частности, фильтр частиц. Эти методы работают с несколькими гипотезами одновременно, что обуславливает их естественную способность адаптироваться к изменениям, происходящим с отслеживаемым объектом. Более того, фильтр частиц способен справляться с нелинейными процессами измерения, оставаясь при этом простым и эффективным для реализации. Но и он предполагает наложение ряда ограничений, особенно в тех случаях, когда целевой объект не имеет характерных отличительных внешних признаков. Данный метод и способы его применения будут подробнее рассмотрены в последующих главах.

2. Фильтр частиц

Принцип вероятностного отслеживания

Вероятностное отслеживание основано на байесовском подходе, который заключается в том, чтобы построить функцию плотности распределения вероятности вектора состояния, используя всю доступную к данному моменту информацию. Эта функция характеризует текущую степень знания о векторе состояния, и с ее помощью можно определить оптимальный (с точки зрения выбранной пользователем функции стоимости) порядок действий.

Проблему отслеживания можно сформулировать с помощью следующего математического представления пространства состояний объекта:

$$x_k = f_k(x_{k-1}, v_{k-1}) \quad (2.1)$$

$$z_k = h_k(x_k, n_k) \quad (2.2)$$

Уравнение (2.1) описывает динамику объекта, то есть изменение его состояния при переходе от кадра $k - 1$ к кадру k . Фактически состояние x_k зависит от предыдущего состояния объекта x_{k-1} и стохастической ошибки v_{k-1} , представляющей собой погрешность в обновлении состояния. Поскольку ошибка v_{k-1} является случайной величиной с известной статистикой, уравнение (2.1) неявно задает функцию плотности распределения вероятности

$p(x_k|x_{k-1})$. Уравнение (2.2) описывает зависимость наблюдения z_k от текущего состояния x_k и вектора ошибки n_k , который задает погрешность процесса измерения состояния объекта. Аналогично (2.1), поскольку n_k является стохастической переменной, уравнение (2.2) неявно задает функцию правдоподобия $p(z_k|x_k)$.

С точки зрения байесовского подхода проблема отслеживания заключается в рекурсивном вычислении некоторой степени доверия к состоянию объекта x_k на кадре k с учетом всех имеющихся к текущему моменту наблюдений. Таким образом, необходимо вычислить функцию плотности распределения вероятности $p(x_k|z_{1:k})$, называемую постериорной (англ. posterior). Вычисление данной функции проводится в два этапа: предсказание и обновление (коррекция). Полагая известным значение искомой функции $p(x_{k-1}|z_{1:k-1})$ на предыдущем кадре, с помощью уравнения Чепмена-Колмогорова можно вычислить априорную функцию плотности распределения вероятности состояния объекта на кадре k :

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1} \quad (2.3)$$

В этом заключается этап предсказания. На кадре k становится известным наблюдение z_k , которое может быть использовано для обновления априорной функции по правилу Байеса:

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}, \quad (2.4)$$

где $p(z_k|z_{1:k-1}) = \int p(z_k|x_k)p(x_k|z_{1:k-1})dx_k$ — нормирующий множитель, зависящий от функции правдоподобия.

Рекурсивные соотношения (2.3) и (2.4) формируют базис для нахождения оптимального байесовского решения. Однако данный подход является концептуальным с той точки зрения, что в общем случае это решение не может быть получено аналитическим путем. Примером частного случая, когда аналитическое решение поставленной задачи все-таки существует, является упомянутый в предыдущей главе фильтр Кальмана. В более сложных случаях искомое решение можно получить с помощью аппроксимации. Одним из возможных способов для этого является фильтр частиц [**Ошибка! Источник ссылки не найден.**].

Аппроксимация с помощью фильтра частиц

В основе метода лежит аппроксимация функции плотности распределения вероятности состояния объекта набором частиц, поведение которых регулируется с помощью их весов. Фактически, частица является представлением некоторого возможного

состояния объекта. Множество S_k (2.5) задает приближенную функцию плотности распределения вероятности $p(x_k|z_{1:k})$ для состояния x_k при заданном наборе наблюдений $z_{1:k}$.

$$S_k = \left\{ (s_k^i, w_k^i), i = \overline{1, N}, \sum_{i=1}^N w_k^i = 1 \right\} \quad (2.5)$$

Алгоритм воспроизведения условной плотности (англ. Conditional Density propagation, Condensation) — базовый алгоритм фильтрации частиц, на основании которого строится большинство алгоритмов данной группы, применяемых в компьютерном зрении. Задача состоит в том, чтобы построить метод восстановления множества S_k на основании S_{k-1} . Пусть построена коллекция взвешенных выборок для кадра $k - 1$:

$$S_{k-1} = \left\{ (s_{k-1}^i, w_{k-1}^i), i = \overline{1, N}, \sum_{i=1}^N w_{k-1}^i = 1 \right\}.$$

Согласно [**Ошибка! Источник ссылки не найден.**], алгоритм состоит из следующих этапов:

1. Вычислить интегральные веса по формуле (2.6).

$$c_i = c_{i-1} + w_{k-1}^i, i = \overline{1, N}, c_0 = 0 \quad (2.6)$$

2. Повторить шаги 2.1 – 2.3 N раз для построения множества S_k .

- 2.1. Определить n – й экземпляр выборки S_k . Для этого выбрать случайное число $r \in [0, 1]$ и вычислить $j = \arg \min_i \{c_i > r\}$. Таким образом, частица s_{k-1}^j попадает в множество S_k с вероятностью, равной ее весовому коэффициенту.

- 2.2. Используя уравнение динамики (2.1), предсказать следующее состояние объекта (2.7):

$$s_k^n = f_k(s_{k-1}^j, v_{k-1}) \quad (2.7)$$

- 2.3. Скорректировать вес полученной частицы в зависимости от текущего наблюдения z_k и его распределения (2.8):

$$w_k^n = p(z_k | x_k = s_k^n) \quad (2.8)$$

3. Нормализовать веса частиц по формуле (2.9):

$$w_k^i = \frac{w_k^i}{c_N}, i = \overline{1, N} \quad (2.9)$$

4. Вычислить оценку текущего состояния x_k как среднюю частицу полученного набора (2.10):

$$x_k = E[S_k] = \sum_{i=1}^N s_k^i w_k^i \quad (2.10)$$

3. Применение фильтра частиц в задачах отслеживания

Как уже было отмечено ранее, алгоритм воспроизведения условной плотности является базовым алгоритмом, реализующим основные принципы фильтра частиц. Существует множество его модификаций, позволяющих адаптировать технологию фильтра частиц для решения конкретной задачи отслеживания. Далее будут рассмотрены некоторые из них.

В общем случае, для того, чтобы фильтр частиц можно было использовать в реальных условиях, необходимо настроить его параметры: задать вектора состояния и наблюдения, выбрать функции, используемые в уравнениях динамики и измерения, а также их стохастические погрешности, выбрать функцию правдоподобия. Для начала приведем пример реализации алгоритма воспроизведения условной плотности, формально описанного в предыдущей главе, для решения задачи отслеживания футбольных игроков на видеозаписи матча [7].

Вектор состояния объекта зависит от выбранного способа его представления. Подробную классификацию возможных вариантов представления объекта на изображении можно найти в работе [Ошибка! Источник ссылки не найден.]. Использование описывающего прямоугольника для выделения объекта в кадре является компромиссным вариантом с точки зрения вычислительных затрат на его определение. Такой способ применяется и в работе [7]. В вектор состояния объекта, таким образом, включаются координаты верхнего левого угла описывающего прямоугольника. Предполагается, что размер прямоугольника в ходе отслеживания не меняется

$$\mathbf{x}_k = [x_k, y_k] \quad (3.1)$$

В качестве уравнения динамики объекта (2.1) авторы [7] используют модель случайных блужданий (англ. Random Walk Model) (3.2).

$$x_k = Fx_{k-1} + v_{(k-1)}, \quad (3.2)$$

где $F = I_2$ — единичная матрица размера 2×2 , v_{k-1} — аддитивный изотропный гауссов шум с нулевым математическим ожиданием и матрицей ковариации Q .

Для получения наблюдения используется модель цветового распределения (англ. Color Distribution Model). Дискретным представлением цветового распределения является гистограмма с m карманами, которая строится по внутренней области описывающего

прямоугольника, заданного вектором состояния объекта. Цветовое распределение $p_y = \{p_y^{(u)}\}_{u=1\dots m}$ для прямоугольной области R_y с левым верхним углом в точке u вычисляется как (3.3):

$$p_y^{(u)} = \frac{1}{I} \sum_{i=1}^I \delta[b(x_i) - u], \quad x_i \in R_y, \quad (3.3)$$

где I — общее количество пикселей в области, δ — дельта-функция Кронекера, u — номер кармана гистограммы, $b(x_i)$ — функция, сопоставляющая цвету в точке x_i области R_y номер соответствующего кармана гистограммы.

Предполагается, что эталонная цветовая гистограмма q_{ref} целевого объекта заранее известна. Процесс измерения заключается в сравнении эталонной гистограммы с гистограммами, построенными по областям, задаваемым каждой из частиц. За текущее наблюдение будет принята та область, цветовая гистограмма которой наиболее близка к эталонной. Для сравнения гистограмм авторы [7] применяют расстояние Бхаттачария (англ. Bhattacharyya distance). Для двух гистограмм $p = \{p^{(u)}\}_{u=1\dots m}$ и $q = \{q^{(u)}\}_{u=1\dots m}$ данная мера определяется как (3.4)

$$B^2(p, q) = 1 - \sum_{u=1}^m \sqrt{p^{(u)} q^{(u)}} \quad (3.4)$$

Поскольку гистограммы строятся отдельно для каждого измерения цветового пространства, итоговое расстояние вычисляется суммированием полученных значений для каждого измерения.

Чем ближе расстояние от цветовой гистограммы $p_{s_k^n}$, построенной по области, задаваемой частицей s^n , тем выше должен быть ее вес. Таким образом, текущее наблюдение в кадре k можно получить с помощью гауссовой функции правдоподобия (3.5).

$$w_k^n = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{B^2(q_{ref}, p_{s_k^n})}{2\sigma^2}} \quad (3.5)$$

На этапе инициализации частицы равномерно распределены по всей области кадра. Более подробную информацию о значениях параметров и о результатах, получаемых при помощи описанного алгоритма, можно посмотреть в работе [7].

Перейдем к рассмотрению алгоритмов отслеживания, построенных на базе алгоритма воспроизведения условной плотности. Одна из его модификаций представлена в работе [Ошибка! Источник ссылки не найден.]. Ее отличительной особенностью является

обновление эталонной гистограммы в ходе отслеживания, что позволяет лучше справиться со значительными изменениями внешнего вида объекта. Условием обновления является падение значения функции правдоподобия для оцененного состояния объекта $w_{E[S_k]}$ ниже заданного порога w_T . Для проверки данного условия необходимо получить цветовую гистограмму $p_{E[S_k]}$ для полученной в результате итерации алгоритма области объекта и посчитать расстояние до эталонной гистограммы q_{ref} , а затем вычислить значение функции правдоподобия согласно формуле (3.5). Процесс обновления эталонной гистограммы на шаге k задается выражением (3.6):

$$q_{ref}^{(u)} = (1 - \alpha)q_{ref}^{(u)} + \alpha p_{E[S_k]}^{(u)}, \quad (3.6)$$

где α — весовой коэффициент, показывающий вклад вычисленного значения гистограммы на текущем кадре. Таким образом, влияние оцененного состояния объекта на эталонную модель уменьшается по экспоненциальному закону по мере «удаления» от кадра, на котором данное состояние было получено. С одной стороны, это позволяет справиться с проблемой значительного изменения внешнего вида объекта по ходу отслеживания, но с другой стороны, при потере объекта вследствие, например, его частичного или полного перекрытия, эталонная гистограмма перестанет соответствовать реальности, что сделает невозможным дальнейшее отслеживание.

Другой вариант модификации алгоритма воспроизведения условной плотности предложили авторы работы [Ошибка! Источник ссылки не найден.], разработавшие алгоритм отслеживания на основе фильтра частиц с адаптацией стандартных отклонений стохастической компоненты уравнения динамики (2.1). Данный алгоритм позволяет отслеживать не только положение, но и размер объекта, а также справляется с резкими изменениями скорости и направления движения объекта. Для этого в вектор состояния объекта включается информация о размере описывающего прямоугольника, а также появляется динамическая составляющая, позволяющая описать скорость движения объекта и изменения его размеров (3.7):

$$\mathbf{x}_k = [x_k, y_k, w_k, \rho_k, \dot{x}_k, \dot{y}, \dot{w}, \dot{\rho}]^T = [s_k, d_k]^T \quad (3.7)$$

Первая компонента $s_k = [x_k, y_k, w_k, \rho_k]$ определяет статическую часть вектора состояния и включает в себя координаты (x_k, y_k) верхнего левого угла описывающего прямоугольника, его ширину w_k и соотношение сторон ρ_k . Вторая компонента вектора состояний $d_k = [\dot{x}_k, \dot{y}, \dot{w}, \dot{\rho}]$ представляет его динамическую часть и содержит, таким образом, производные элементов статической компоненты.

В качестве уравнения динамики взята модель динамики первого порядка, в рамках которой уравнение (2.1) принимает вид:

$$x_k = Ax_{k-1} + v_{k-1}, A = \begin{bmatrix} I_4 & I_4 \Delta t \\ 0 & I_4 \end{bmatrix}, \quad (3.8)$$

где I_4 — единичная матрица размера 4×4 , Δt — шаг по времени. Так как скорость измеряется (в пиксель/кадр) между последовательными кадрами, $\Delta t = 1$. Матрица ковариации для дисперсии шума v_{k-1} задается с помощью вектора стандартных отклонений (3.9), в котором также выделены статическая Σ^s и динамическая Σ^d составляющие.

$$\Sigma = [\sigma^x, \sigma^y, \sigma^w, \sigma^p, \sigma^{\dot{x}}, \sigma^{\dot{y}}, \sigma^{\dot{w}}, \sigma^{\dot{p}}]^T = [\Sigma^s, \Sigma^d]^T \quad (3.9)$$

Основная идея алгоритма заключается в адаптации вектора (3.9) к результатам отслеживания. Когда цветовая гистограмма $p_{E[S_k]}$, построенная на основе вычисленного состояния объекта, близка к эталонной, трекер должен предсказывать следующее состояние на основе рассчитанной по предыдущим кадрам динамики объекта, в противном случае его поведение должно приближаться к модели случайных блужданий для снижения времени на восстановление. Другими словами, в первом случае необходимо увеличивать вклад динамической составляющей путем повышения значений соответствующей компоненты Σ^d вектора стандартных отклонений (3.9). Во втором случае, напротив, нужно увеличивать статическую составляющую Σ^s , как можно сильнее при этом снижая влияние динамической части. Для этого авторы [Ошибка! Источник ссылки не найден.] предлагают задать адаптацию вектора стандартных отклонений к точности отслеживания с помощью сигмоидальной функции (3.10)

$$\zeta(\psi_k) = \frac{1}{2} \left(1 + \frac{2}{\sqrt{\pi}} \int_0^{\alpha(\psi_k - \beta)} e^{-t^2} dt \right), \quad (3.10)$$

где $\psi_k = B^2(q_{ref}, p_{E[S_k]})$, α и β — параметры, задающие крутизну и положение функции соответственно. Тогда значения стандартных отклонений на каждой итерации алгоритма будут вычисляться согласно (3.11):

$$\begin{cases} \Sigma_k^s = \zeta(\psi_k) \min(w_k, h_k) \Sigma_0^s \\ \Sigma_k^d = (1 - \zeta(\psi_k)) \min(w_k, h_k) \Sigma_0^d \end{cases} \quad (3.11)$$

где Σ_0^s и Σ_0^d — исходные значения статической и динамической компонент вектора стандартных отклонений соответственно, подбираемые в зависимости от специфики конкретной задачи отслеживания. Чтобы ускорить процесс восстановления работоспособности трекера после потери объекта, предлагается дополнительно снизить

уменьшить влияние рассчитанной динамики объекта с помощью искусственного взвешивания соответствующей компоненты вектора состояния (3.12):

$$d'_k = (1 - \zeta(\psi_k))d_k \quad (3.12)$$

Данный алгоритм предназначен в основном для работы в сложных с точки зрения задачи отслеживания условиях, когда изменение размера объекта, скорости и направления его движения происходит за короткие промежутки времени, а также когда наблюдаются частые перекрытия объекта другими элементами сцены.

Помимо вычисления цветовых распределений, в качестве процесса измерения могут применяться алгоритмы распознавания объекта, тогда в качестве наблюдений на вход фильтра частиц будут подаваться области кадра, которые вероятно заняты отслеживаемым объектом. Пример такой реализации фильтра частиц представлен в работе [**Ошибка! Источник ссылки не найден.**]. Выделение областей изображения, которые возможно заняты отслеживаемыми объектами, производится путем определения фоновых пикселей. Для этого цветовая модель фона представляется с помощью смешанных Гауссовых моделей (англ. Gaussian Mixture Model, GMM). Для отсеечения ложных областей применяется классификация на основе опорных векторов (англ. Support Vector Classification, SVC). После определения итогового набора областей каждой из них присваивается свой трекер (фильтр частиц). Его основное отличие в данной работе от рассмотренных ранее алгоритмов заключается в выборе функции правдоподобия, в качестве которой авторы применяют регрессионную функцию (3.13), построенную с помощью метода опорных векторов (англ. Support Vector Regression, SRV).

$$w_k^i = \sum_{i=1}^N \alpha_i K(x_k, s_k^i) + b_k \quad (3.13)$$

Здесь α и b являются параметрами регрессионной функции. $K(\cdot)$ — функция ядра, используемая методом опорных векторов. После каждой итерации алгоритма найденные области проверяются на наличие в них объектов с помощью набора данных, полученного на выходе классификатора. Если в некоторой области на протяжении заданного числа L кадров не был обнаружен объект, считается, что он исчез из поля зрения камеры, и соответствующий данной области трекер освобождается.

Данный метод применим только в тех случаях, когда фон является статичным, а также если имеется достаточный набор образцов отслеживаемых объектов для обучения классификатора. В работе [**Ошибка! Источник ссылки не найден.**] алгоритм применялся для отслеживания футбольных игроков во время матча.

Еще один алгоритм на базе фильтра частиц, использующий в качестве процесса измерения алгоритм распознавания объекта, описан в работе [Ошибка! Источник ссылки не найден.]. Определение областей изображения, возможно, относящихся к отслеживаемым объектам, производится путем вычитания фона.

Характерная особенность данного метода заключается в том, что вектор состояния объекта не содержит информации о его форме и размерах — они представляются набором частиц. При этом в вектор включается информация о цвете объекта и скорости его передвижения. Таким образом, в данном методе он принимает вид (3.14):

$$x_k = [x_k, y_k, r_k, g_k, \dot{x}_k, \dot{y}_k] \quad (3.14)$$

Цвет объекта представлен двумя компонентами нормализованного пространства RGB (rg-пространства). Уравнение (2.1) динамики объекта здесь представлено тремя различными выражениями: для положения объекта (3.15):

$$\begin{aligned} x_k &= x_{k-1} + \dot{x}_{k-1} + v_{x_{k-1}}, \\ y_k &= y_{k-1} + \dot{y}_{k-1} + v_{y_{k-1}}, \end{aligned} \quad (3.15)$$

цвета (3.16):

$$\begin{aligned} r_k &= r_{k-1} + v_{r_{k-1}}, \\ g_k &= g_{k-1} + v_{g_{k-1}}, \end{aligned} \quad (3.16)$$

и скорости (3.17):

$$\begin{aligned} \dot{x}_k &= \gamma \dot{x}_{k-1} + (1 - \gamma)(x_k - x_{k-1}) \\ \dot{y}_k &= \gamma \dot{y}_{k-1} + (1 - \gamma)(y_k - y_{k-1}) \end{aligned} \quad (3.17)$$

где γ — авторегрессионный параметр для сглаживания оценок скорости объекта, $0 \leq \gamma \leq 1$. В отличие от работы [Ошибка! Источник ссылки не найден.], здесь скорость движения объекта не является стохастической величиной, т.е. не подвержена шумовым воздействиям.

В качестве наблюдений алгоритм получает набор из D областей $\{C_d\}$. Для каждой частицы по всем областям набора вычисляются вектора ошибок (3.18):

$$e_d^i = [e_{x_d}^i, e_{y_d}^i, e_{r_d}^i, e_{g_d}^i, e_{\dot{x}_d}^i, e_{\dot{y}_d}^i] \quad (3.18)$$

Компоненты $e_{x_d}^i$ и $e_{y_d}^i$ вычисляются как кратчайшее расстояние от частицы s^i до границы области C_d . Если частица находится внутри области, значения данных компонент вектора ошибки равны 0. Компоненты, соответствующие цвету объекта, вычисляются согласно (3.19):

$$\begin{aligned} e_{r_d}^i &= C_{r_d} - s_{r_d}^i \\ e_{g_d}^i &= C_{g_d} - s_{g_d}^i \end{aligned} \quad (3.19)$$

где C_{r_d} и C_{g_d} — усредненные значения соответствующей цветовой компоненты по области C_d . Последние два элемента вектора ошибки, соответствующие скорости перемещения объекта, равны 0. Функция правдоподобия для вычисления веса частицы имеет вид (3.20):

$$w^i = \sum_{d=1}^D \exp\left(-\frac{e_d^{i T} R^{(-1)} e_d^i}{2}\right), \quad (3.20)$$

R — матрица ковариации, нормализующая относительное влияние каждого элемента вектора состояния на итоговый вес частицы.

Данный метод, как и предыдущий, требует статичного фона для корректного извлечения областей кадра, занимаемых объектами. Но при этом ему не нужны заранее подготовленные образцы объектов для проведения классификации.

Заключение.

В данной работе был проведен краткий обзор различных подходов к визуальному отслеживанию. В зависимости от того, используются ли алгоритмы распознавания и сопоставления областей объектов на соседних кадрах отдельно или совместно, разделяют отслеживание на основе распознавания и распознавание на основе отслеживания. Второй подход предоставляет более широкие возможности, так как позволяет учитывать всю историю отслеживания, а не только результаты с предыдущего кадра.

Среди всего множества методов визуального отслеживания особое место занимает фильтр частиц — вероятностный метод, позволяющий моделировать нелинейное поведение объекта и предоставляющий гибкий и удобный инструментарий для отслеживания объектов в достаточно сложных условиях: изменение освещения сцены, изменение размеров объекта при приближении его к камере и удалении от нее, значительное изменение внешнего вида объекта, резкие изменения скорости и направления его движения, частичные и полные перекрытия объекта. При этом реализация данного подхода достаточно проста и не требует особых вычислительных затрат.

Базовые принципы фильтра частиц реализует алгоритм воспроизведения условной плотности, на его основе разработаны различные алгоритмы отслеживания, в которых к нему применяются незначительные модификации для адаптации его под конкретную задачу. Например, обновление эталонной цветовой гистограммы объекта [**Ошибка! Источник ссылки не найден.**], позволяет повысить точность результатов отслеживания в ситуациях, когда внешний вид объекта претерпевает значительные изменения в ходе отслеживания, а адаптация стохастической компоненты уравнения динамики [**Ошибка! Источник ссылки**

не найден.] улучшает результаты отслеживания в случаях резких изменений размера, скорости, и направления движения объекта, а также снижает время на восстановление после потери объекта. Фильтр частиц может применяться также совместно с алгоритмами распознавания, используя их для получения наблюдений **[Ошибка! Источник ссылки не найден.]**, **[Ошибка! Источник ссылки не найден.]**.

Особый интерес из рассмотренных методов представляет модификация фильтра частиц с адаптацией стохастической составляющей уравнения динамики. В дальнейшем планируется проведение более углубленного изучения данного метода, его реализация и исследование его поведения в различных условиях отслеживания.

Список литературы

1. Кустикова В. Д. Отслеживание движения и алгоритмы сопровождения ключевых точек: лекция. Н. Новгород: Нижегородский государственный университет им. Н.И. Лобачевского, 2013. 34 с.
2. Arulampalam M. S., Maskell S., Gordon N., Clapp T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking // IEEE Transactions on Signal Processing. 2002. Vol. 50, no. 2. Pp. 174-188. DOI: 10.1109/78.978374.
3. Deardena A., Demirisa Y., Graub O. Tracking football player movement from a single moving camera using particle filters // Proceedings of the 3rd European Conference on Visual Media Production (CVMP2006). London 2006. Pp. 29-37.
4. Del Bimbo A., Dini F. Particle filter-based visual tracking with a first order dynamic model and uncertainty adaptation // Computer Vision and Image Understanding. 2011. Vol. 115, no. 6. Pp. 771-786. DOI: 10.1016/j.cviu.2011.01.004.
5. Grabner H., Grabner M., Bischof H. Real-time tracking via on-line boosting // Proceedings on British Machine Vision Conference (BMVC). Edinburgh 2006. Vol. 1. Pp. 47-56.
6. Guangyu Z., Changsheng X., Qingming H., Wen G. Automatic Multi-Player Detection and Tracking in Broadcast Sports Video using Support Vector Machine and Particle Filter // IEEE International Conference on Multimedia and Expo. Toronto, Ont. 2006. Pp.1629-1632. DOI: 10.1109/ICME.2006.262859.
7. Jaward M., Mihaylova L., Canagarajah N., Bull D. Multiple object tracking using particle filters // Aerospace Conference, 2006 IEEE. Big Sky, MT 2006. Pp. 8-. DOI: 10.1109/AERO.2006.1655926.

8. Nummiaro K., Koller-Meier E., Van Gool L. An adaptive color-based particle filter // Image and Vision Computing. 2003. Vol. 21, no. 1. Pp. 99–110. DOI: 10.1016/S0262-8856(02)00129-4.
9. Rui Y., Chen Y. Better proposal distributions: object tracking using unscented particle filter // Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (CVPR 2001). Kauai, Hawaii 2001. Vol. 2. Pp.II-786 – II-793. DOI: 10.1109/CVPR.2001.991045.
10. Yilmaz A., Javed O., Shah M. Object tracking: A survey // ACM Computing Surveys (CSUR). 2006. Vol. 38, no. 4. DOI: 10.1145/1177352.1177355.