Термины и определения

Трекер – программная реализация процедуры отслеживания объекта.

Вступление

Визуальное отслеживание заключается в последовательном определении местоположения целевого объекта на каждом кадре видеопотока. Оно по-прежнему остается открытой проблемой во многих областях, так или иначе связанных с обработкой видеозаписей и получением из них некоторой информации. В качестве примеров можно привести системы человеко-машинного взаимодействия, системы видеонаблюдения, системы анализа спортивных матчей, появившиеся в недавнее время системы «умного дома», 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.

В последнее время достаточно широкое распространение получила технология «фильтра частиц» (particle filter), предоставляющая широкие возможности и гибкий инструментарий для отслеживания деформируемых объектов в видеопоследовательностях со сложным изменяющимся фоном. В данной статье речь пойдет о конкретном примере применения фильтра частиц для покадрового определения местоположения объекта. В первой главе будет дан краткий обзор существующих на сегодняшний день подходов к отслеживанию объектов в видеопотоке. Далее будут рассмотрены основные принципы метода фильтрации частиц. Третья глава будет посвящена реализации данного метода в рамках задачи отслеживания объекта на видеозаписи. В последующих главах будет произведен анализ полученных с помощью фильтра частиц результатов.

# 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 трекеры. Также они могут предоставлять точные оценки положения цели, увеличивая тем самым точность результатов работы трекера. 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 выделить AdaBoost [1]. Основная идея данного алгоритма – представить проблему отслеживания в виде задачи бинарной классификации. Принцип работы заключается в следующем: зная положение объекта на текущем кадре , классификатор вычисляет как можно больше возможных положений объекта на следующем кадре заданной области поиска, после чего анализируется полученная в результате вычислений карта достоверности (confidence map) с целью найти наиболее вероятное положение объекта, и, наконец, происходит обновление классификатора. Для построения используемых классификатором гипотез применяется процесс сопоставления с эталоном (template matching). Для сравнения используются признаки Хаара (Haar-like features), гистограммы направленных градиентов (Histogram of Oriented Gradients, HOG) и локальные бинарные шаблоны (Local Binary Patterns, LBP).

Другим алгоритмом, использующим отслеживание с помощью распознавания, является алгоритм, основанный на методе опорных векторов (Support Vector Machine, SVM) [2]. Выделение объектов происходит путем определения пикселей фона с помощью смешанной Гауссовой модели (Gaussian Mixture Model, GMM). При этом могут быть выделены некоторые посторонние объекты, не представляющие интереса для трекера. Для разбиения выделенных объектов на классы интересных и неинтересных с точки зрения отслеживания используется классификатор, обученный с помощью метода опорных векторов.

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 описанными ограничениями справляется второй подход, распознавание на основе отслеживания. В этом подходе 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 (Mean Shift) и непрерывно адаптивный сдвиг среднего (Continuously Adaptive Mean-shift, CAM-shift). Они определяют для каждого кадра весовую функцию, такую, что положение объекта соотносится с ее минимальным (или максимальным) значением. В этом случае задача визуального отслеживания сводится к оптимизации весовой функции. Для поиска минимума (максимума) алгоритмы сдвига среднего используют метод градиентного спуска. Проблемой в данном случае является выбор начальной точки для спуска. Таким образом, необходимо решать вопрос о нахождении локального оптимума, который, в конечном итоге, может оказаться достаточно далеко от реального положения объекта. По этой причине детерминированные алгоритмы показывают неудовлетворительные результаты на достаточно долгих видеопоследовательностях, так как внешний вид объекта может коренным образом меняться на протяжении видеозаписи, вследствие чего весовая функция скорее всего будет мультимодальна, и определение его положения станет слишком непредсказуемо.

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 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 (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. Кроме того, фильтр Кальмана в чистом виде не применим к нелинейным системам. Для решения этой проблемы были разработаны модификации данного метода: расширенный фильтр Кальмана (Extended Kalman Filter, EKF), применяющий процесс линеаризации, и ансцентный фильтр Кальмана[[1]](#footnote-1) (Unscented Kalman Filter, UKF), использующий детерминированные выборки. Однако в большинстве своем эти решения показывают удовлетворительные результаты только при задании строгих ограничений на область применения алгоритма. Обычно они требуют задания 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.

Значительный интерес в области вероятностного отслеживания вызывают 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. В следующей главе этот алгоритм будет рассмотрен подробнее.

# Метод фильтра частиц

2.1. Принцип вероятностного отслеживания.

Вероятностное отслеживание основано на байесовском подходе, который заключается в том, чтобы построить функцию плотности распределения вероятности (probability density function, pdf) 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:

(2.1)

(2.2)

Уравнение описывает динамику системы, то есть изменение ее состояния при переходе от кадра к кадру . Фактически состояние зависит от предыдущего состояния объекта и стохастической ошибки , представляющей собой погрешность в обновлении состояния. Поскольку ошибка является случайной величиной с известной статистикой, уравнение неявно задает функцию плотности распределения вероятности . Уравнение (2.2) описывает зависимость наблюдения от текущего состояния и вектора ошибки , который задает погрешность процесса измерения состояния объекта. Аналогично (2.1), поскольку является стохастической переменной, уравнение (2.2) неявно задает функцию правдоподобия .

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

(2.3)

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

(2.4)

где – 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 алгоритмов для этого является фильтр частиц [3].

* 1. Фильтр частиц

В основе алгоритма лежит аппроксимация функции плотности распределения вероятности состояния объекта коллекцией взвешенных выборок – частиц, поведение которых регулируется с помощью их весов. Фактически, частица является представлением некоторого возможного состояния объекта. Множество (2.5) задает приближенную функцию плотности распределения вероятности для состояния при заданном наборе наблюдений .

(2.5)

Алгоритм воспроизведения условной плотности (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 на основании . Пусть построена коллекция взвешенных выборок для кадра :

. Согласно [4] алгоритм состоит из следующих этапов:

1. Вычислить интегральные веса согласно .

(2.6)

1. Повторить шаги 2.1 – 2.3 раз для построения множества .
   1. Определить экземпляр выборки . Для этого выбрать случайное число и вычислить . Таким образом, частица попадает в множество с вероятностью, равной ее весовому коэффициенту.
   2. Предсказать следующее состояние согласно уравнению динамики (2.1):

(2.7)

* 1. Скорректировать вес полученной частицы в зависимости от текущего наблюдения и его распределения:

(2.8)

1. Нормализовать веса частиц:

(2.9)

1. Вычислить оценку текущего состояния как среднюю частицу полученного набора:

(2.10)

* 1. Пример реализации фильтра частиц

Для того чтобы фильтр частиц можно было использовать для решения конкретной задачи, необходимо настроить параметры вышеописанного формального алгоритма в зависимость от области его применения: задать вектора состояния и наблюдения, выбрать функции, используемые в уравнениях динамики и измерения, а также их стохастические погрешности, выбрать функцию правдоподобия.

Далее будет описана простая реализация алгоритма для решения задачи отслеживания объекта в видеопотоке, а также вариант с адаптацией погрешностей, предложенный в работе [5].

Для обеспечения компромисса между выделением контуров из кадра и вычислительными затратами объект часто аппроксимируется ограничивающей прямоугольной областью. Чтобы предсказать примерное положение объекта в следующем кадре, в качестве уравнения динамики (2.1) обычно выбирают модель случайных блужданий (англ. Random Walk Model) [6], модель динамики первого порядка [5, 7] или авторегрессионную модель второго порядка [8]. В данной работе рассматривается модель динамики первого порядка. С учетом этого, можно предложить следующий вариант вектора состояния объекта (2.11), включающий в себя информацию о положении и размере описывающего прямоугольника, а также о его скорости их изменения [5]:

(2.11)

Первая компонента определяет статическую часть вектора состояния и включает в себя координаты верхнего левого угла описывающего прямоугольника, его ширину и соотношение его сторон , где - высота прямоугольника. Вторая компонента вектора состояний описывает его динамическую часть и содержит, таким образом, производные элементов статической компоненты. Уравнение (2.1) в рамках модели динамики первого порядка принимает следующий вид:

(2.12)

где – единичная матрица размера , – шаг по времени. Так как скорость измеряется (в пиксель/кадр) между последовательными кадрами, . – аддитивный изотропный Гауссов шум с нулевым математическим ожиданием и дисперсией, задаваемой вектором стандартных отклонений (2.13), в котором также были выделены статическая и динамическая компоненты.

(2.13)

Для процесса измерения используется модель цветового распределения (англ. Color Distribution Model). Представление целевого объекта с помощью цветового распределения позволяет повысить ошибкоустойчивость получаемых наблюдений для подвижных объектов, учитывая их возможное вращение и частичные перекрытия. Дискретным представлением цветового распределения является гистограмма с карманами. Она может быть построена как в цветовом пространстве RGB, так и в HSV, причем в последнем случае снижается чувствительность к изменениям освещения, поскольку в пространстве HSV значение интенсивности (Value) отделено от цветовых составляющих: оттенка(Hue) и насыщенности (Saturation), и при построении гистограммы интенсивность можно не учитывать. Гистограмма строится по области описывающего объект прямоугольника.

Цветовое распределение для прямоугольной области с левым верхним углом в точке вычисляется как:

(.)

где – общее количество пикселей в области, – дельта-функция Кронекера, - номер кармана гистограммы, – функция, сопоставляющая цвету в точке области номер соответствующего кармана гистограммы.

Процесс измерения заключается в сравнении цветовой гистограммы, построенной по известной области целевого объекта с гистограммами, построенными по областям, задаваемым каждой частицей. В результате за текущее наблюдение будет принята та область, цветовая гистограмма которой наиболее близка к эталонной. Таким образом, рассматриваемый метод предполагает, что эталонная гистограмма отслеживаемого объекта известна заранее и не претерпевает значительных изменений на протяжении всей видеозаписи, что является одним из его ограничений. Различие между гистограммами может быть выражено такими количественными оценками, как корреляция, пересечение, расстояние Пирсона (), расстояние (или коэффициент) Бхаттачария (Bhattacharyya distance). Последнее является наиболее распространенной оценкой при сравнении цветовых гистограмм. Для двух гистограмм и расстояние Бхаттачария определяется следующим образом:

(2.15)

Поскольку для каждого измерения цветового пространства HSV строится отдельная гистограмма, итоговое расстояние вычисляется как сумма расстояний (2.15) для каждой из гистограмм:

(.)

где – эталонные гистограммы для оттенка и насыщенности соответственно, - аналогичные гистограммы для области, заданной частицей .

Чем ближе расстояние от цветовой гистограммы, построенной по области, задаваемой частицей, до эталонной гистограммы, тем выше должен быть вес частицы. Таким образом, текущее наблюдение в кадре можно получить с помощью распределения Гаусса (2.17).

(2.17)

На этапе инициализации частицы могут быть равномерно распределены по всей области изображения [6], или могут быть сгруппированы в областях возможного появления объекта [7].

Фильтр частиц с адаптацией погрешности

Описанный выше простой алгоритм отслеживания, использующий технику фильтрации частиц, имеет ряд проблем. Во-первых, модель динамики первого порядка не позволяет напрямую учитывать ускорение движущегося объекта, только с помощью вводимой погрешности. При этом динамическая составляющая ошибки с предыдущих кадров оказывает влияние на статическую часть вектора состояния на текущем кадре, уменьшая тем самым точность определения положения и размера объекта. В работе [5] показано, что значения стандартных отклонений должны как минимум на один порядок превышать значения стандартных отклонений , чтобы значение моделируемого шума для динамической части вектора состояния было достаточно высоко, чтобы учитывать изменения скорости объекта, но при этом не оказывало значительного влияния на погрешность для статической части. Во-вторых, так как фильтр распространяет частицы в ограниченной области пространства состояний, что с одной стороны позволяет более точно определять направление движения объекта, возникают проблемы с отслеживанием объекта, движущееся к камере или от нее. Такое движение приводит к значительным увеличениям размера объекта в кадре. Чтобы учитывать это изменение, необходимо, чтобы ему было сопоставлено аналогичное изменение величины шума, влияющего на размер объекта. И в-третьих, поскольку модель цветового распределения является достаточно неустойчивой для представления объекта, для повышения точности результатов необходимо уменьшать область распространения частиц, где это возможно.

Для решения вышеуказанных проблем авторы [5] предлагают адаптировать стандартные отклонения и шума из уравнения (2.12) к получаемым результатам отслеживания, сделав их зависимыми от текущего кадра. Исходные значения и выбираются исходя из специфики конкретной задачи, а значения отклонений шума на последующих кадрах вычисляется на основе данных величин с учетом масштабирования объекта и способности трекера предоставлять точные оценки динамики объекта. При этом адаптация к размеру объекта задается линейным соотношением, а к точности отслеживания – сигмоидальной функцией (2.18), где - расстояние между цветовой гистограммой, построенной для вычисленной на кадре области, занимаемой объектом, и - параметры, задающие крутизну и положение функции. При и достигаются наиболее точные результаты [5].

(2.18)

Стандартные отклонения вычисляются следующим образом:

(2.19)

Чем хуже полученная оценка состояния объекта на кадре , тем выше значение функции , тем меньше влияние динамической части вектора погрешности и тем ближе поведение трекера к модели случайных блужданий. В противном случае, когда трекер показывает достаточно точные результаты, снижается вклад статической составляющей вектора ошибок и увеличивается значение его динамической части; таким образом, большее внимание уделяется наблюдаемой динамике объекта.

Чтобы ускорить процесс восстановления работоспособности трекера после потери объекта, авторы [5] предлагают дополнительное снижение влияния динамической части шума путем ее искусственного взвешивания:

(2.20)

Из (2.20) видно, что при высокой точности отслеживания множитель не имеет особого практического эффекта, но при ухудшении получаемых оценок состояния объекта позволяет ускорить переход к модели случайных блужданий.

# Эксперименты и результаты

Простой алгоритм фильтра частиц и его вариант с адаптацией погрешности были реализованы на языке C++ с использованием графической библиотеки OpenCV. Эталонная гистограмма и реальные состояния объекта (ground truth data) известны заранее. Эксперименты проводились на данных, предоставленных авторами работы [5] и находящимися в открытом доступе на сайте <http://www.micc.unifi.it/dini/research/particle-filter-based-visual-tracking>.

.На видеозаписях представлена движущаяся игрушечная машинка с дистанционным управлением. Видеозаписи сделаны в лабораторных условиях с неподвижной камеры и предоставляют возможность проверить работу алгоритма при различных условиях: движение объекта с ускорением, с резкими изменениями направления траектории, приближение и удаление от камеры, перекрытия. Всего в наборе данных представлено 10 видеопоследовательностей.

Для оценки точности определения состояния объекта на кадре был использован критерий совпадения с реальными данными, предложенный в работе [9]:

(3.1)

где и – описывающие прямоугольники для реальных данных и оценки состояния, полученной с помощью фильтра частиц, соответственно, – площадь прямоугольника . Таким образом, показатель качества представляет собой отношение площади пересечения описывающих прямоугольников для реальных и вычисленных состояний объекта к максимальной из их площадей. Для оценки качества отслеживания для всей видеозаписи берется среднее значение величины (3.1):

(3.2)

где - количество кадров в видеозаписи. Поскольку данная оценка включает в себя и те кадры, в которых объект был потерян вследствие перекрытия, вычисленная на ее основе точность определения состояния объекта (его положения и размера) может оказаться заниженной. Поэтому для сравнения рассмотренных алгоритмов с точки зрения качества определения состояния объекта был введен индекс качества , который вычисляется аналогично (3.2), но только для тех кадров, в которых объект был определен трекером (считается, объекта успешно определен в кадре , если ). Также для определения способности трекера восстанавливаться после потери объекта вычисляется среднее время восстановления . Эта величина определяется как усредненное по всей видеопоследовательности количество кадров, прошедших с момента потери объекта до его обнаружения трекером [5].

Дальнейшие результаты приведены для видеозаписи номер 10 из используемого для тестирования набора данных. Длина видеопоследовательности составляет 1188 кадров, отслеживаемый объект меняет скорость и направление движения, несколько раз перекрывается неподвижными элементами сцены, масштаб объекта изменяется по мере его приближения к камере и удаления от нее. Таким образом, данная видеозапись предоставляет возможность проверить работу алгоритма в достаточно сложных с точки зрения отслеживания условиях. Результаты были получены для частиц.

В таблице 3.1 представлены перечисленные характеристики для простого алгоритма отслеживания на основе фильтра частиц и его варианта с адаптацией погрешностей. Приведено также количество кадров , в которых объект был успешно определен. Полученные результаты показывают, что фильтр частиц с адаптацией погрешностей почти в 4 раза быстрее восстанавливается после потери объекта и имеет более точную оценку определения состояния объекта, чем простой фильтр частиц. Графически изменение качественного показателя для обоих алгоритмов приведено на рис. 3.1.

*Таблица 3.1. Сравнительные характеристики алгоритмов*

|  |  |  |
| --- | --- | --- |
|  | Простой фильтр частиц | Фильтр частиц с адаптацией погрешностей |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |
| --- |
| quality.jpg |
| Рис. 3.1 |

Видно, что алгоритм с адаптацией погрешностей показал более точные результаты, чем простой алгоритм фильтра частиц, и успешно справился с частичными и полными перекрытиями объекта, которые происходили несколько раз в ходе видеозаписи, а также с резкими изменениями скорости и направления движения объекта и с изменениями его внешнего вида.

Литература

[1] Helmut Grabner, Michael Grabner, and Horst Bischof. Real-time tracking

via on-line boosting. InBritish Machine Vision Conference, 2006

[2] Guangyu Zhu; Changsheng Xu; Qingming Huang; Wen Gao, "Automatic Multi-Player Detection and Tracking in Broadcast Sports Video using Support Vector Machine and Particle Filter," *Multimedia and Expo, 2006 IEEE International Conference on* , vol., no., pp.1629,1632, 9-12 July 2006

[3] Arulampalam, M.S.; Maskell, S.; Gordon, N.; Clapp, T., "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *Signal Processing, IEEE Transactions on* , vol.50, no.2, pp.174,188, Feb 2002

[4] Кустикова В.Д. Отслеживание движения и алгоритмы сопровождения ключевых точек [лекция].

[5] Alberto Del Bimbo, Fabrizio Dini, Particle filter-based visual tracking with a first order dynamic model and uncertainty adaptation, Computer Vision and Image Understanding, Volume 115, Issue 6, June 2011, Pages 771-786

[6] Jaward, M.; Mihaylova, L.; Canagarajah, N.; Bull, D., "Multiple object tracking using particle filters," *Aerospace Conference, 2006 IEEE* , pp.8-

[7] K. Nummiaro, E. Koller-Meier, and L. Van Gool. An adaptive color-based

particle filter.Image and Vision Computing, 21:99–110, 2002

[8] Yizheng Cai, Nando de Freitas, and James Little. Robust visual tracking

for multiple targets. pages 107–118. 2006

[9] Ihsin T. Phillips and Atul K. Chhabra. Empirical performance evaluation

of graphics recognition systems. IEEE Trans. Pattern Anal. Mach. Intell.,

21(9):849–870, 1999.

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») [↑](#footnote-ref-1)