Минобрнауки России

Федеральное государственное автономное образовательное учреждение высшего образования   
«Национальный исследовательский университет   
«Московский институт электронной техники»

Факультет микроприборов и технической кибернетики

Кафедра высшей математики №1

Димаков Владислав Сергеевич

Бакалаврская работа   
по направлению 01.03.04 «Прикладная математика»

Обнаружение и слежение за объектами в реальном времени на основе самообучающегося классификатора

Студент Димаков В.С.

Научный руководитель,

доцент, кандидат физико-математических наук Козлитин И.А.

Москва 2017

Оглавление

[Введение 3](#_Toc478548654)

[Актуальность проблемы 3](#_Toc478548655)

# Введение

## Актуальность проблемы

В настоящее время от систем видеонаблюдения требуется не только предоставление возможности воспроизведения и записи видеопотока с камеры, но и возможности решения в автоматическом режиме множество задач без участия человека, начиная от простого детектирования движения в области наблюдения, заканчивая высокоточным подсчётом проехавших машин или прошедших людей.

Большинство задач, решаемых системами видеонаблюдения, направлены на получение различных данных об объектах в области наблюдения, поэтому для сбора и последующей обработки информации наиболее важным вопросом является определение положений требуемых объектов на каждом кадре видеопотока.

## Цели и задачи выпускной квалификационной работы

Целью работы является исследование и анализ эффективности методов обнаружения и слежения за объектами в реальном времени на основе данных получаемых из видеопотока.

## Цифровое растровое полутоновое изображение

Изображение – это визуальное представление чего-либо. Цифровым изображением называется массив данных, полученный путем дискретизации (аналого-цифрового преобразования) оригинала. Существуют три основных способа цифрового представления изображений: растровая графика, векторная графика и фрактальная графика. Растровое изображение – это изображение, представляющее собой двумерную матрицу (также называемую битовой картой), каждый элемент которой характеризует цвет соответствующего пикселя. Пикселем называется наименьший логический элемент цифрового растрового изображения. В любое растровое изображение входит некоторое число пикселей, которые формирую объекты, изображенные на нем, а величина, определяющая количество пикселей изображения на единицу площади называется разрешением. Как и все растровые изображения, полутоновое кодируется в цифровом виде с помощью матрицы, хранящей значения пикселей. Каждый пиксель полутонового изображения может кодироваться различным количеством бит, что определяет количество возможных полутонов, называемых уровнями серого.

В данной работе используются растровые полутоновые изображения с разрешением пикселей и числом бит на пиксель равным .

Подвести к необходимости применения классификатора.

## Задача классификации

Вероятностная постановка задачи классификации выглядит следующим образом. Пусть множество пар «объект, метка класса» является вероятностным пространством с неизвестной вероятностной мерой . Имеется конечная обучающая выборка наблюдений , сгенерированная согласно вероятностной мере . Требуется построить алгоритм , способный классифицировать произвольный объект

Существуют различные методы решения задачи классификации, такие как наивный байесовский классификатор, метод ближайших соседей, деревья и леса решений, нейронные сети.

Далее под обучающей выборкой будем понимать независимую выборку из некоторого (неизвестного) распределения . Здесь – векторы признаков (называемые прецедентами), координаты которых представляют значения признаков (независимых переменных), измеряемых на некотором объекте (образе); – метки классов, .

## Деревья решений

Граф состоит из конечного непустого множества , элементы которого называются вершинами, и множества пар вершин , называемых ребрами.

Путем в графе называется последовательность ребер вида . Если , то такой путь называется циклом.

Если пара вершин , образующая ребро , является упорядоченной, то такое ребро называется ориентированным или дугой, ведущей из вершины в вершину.

Если все ребра графа ориентированы и сам граф не имеет циклов, то он называется деревом. Под корневым деревом понимается дерево, в котором одна вершина выделена и называется корнем.

Далее рассматриваются только ориентированные корневые деревья, в которых дуги направлены по направлению от корня. Заметим, что такие деревья удовлетворяют следующим условиям:

* существует только одна вершина, называемая корнем, в которую не ведет ни одна дуга;
* в каждую вершину (исключая корень) ведет только одна дуга;
* существует единственный путь от корня к любой вершине.

Если – некоторая дуга, то вершина называется родителем вершины , а вершина – потомком .

Вершина, не имеющая потомков, называется терминальной вершиной или листом.

Дерево называется бинарным, если каждая его вершина (за исключением терминальных) имеет ровно двух потомков.

Обозначим – множество всех возможных значений векторов признаков (пространство образов). Тогда деревом решений будет называться дерево, с каждой вершиной которого связаны:

* Некоторое подмножество ; с корневой вершиной связывается все пространство образов ;
* Подвыборка обучающей выборки , такая, что ; таким образом с корневой вершиной связывается вся выборка ;
* Некоторая функция (правило) (здесь – количество потомков вершины ), определяющая разбиение множества на непересекающихся подмножеств. С листьями дерева не связывается никакая функция.

Обозначим вершину, являющуюся -м потомком вершины . Тогда множество и функция определяют множества следующим образом: .

Цель построения дерева решений состоит в классификации векторов из распределения . Процесс принятия решений начинается с корневой вершины и состоит в последовательном применении правил, связанных с вершинами дерева. Результатом этого процесса является определение листа такого, что . В этом случае вектор относится к классу, являющемуся мажорантным (наиболее часто встречающимся) в подвыборке , соответствующей данному листу.

## Алгоритм CART

Алгоритм CART (Classification and Regression Tree) предназначен для решения задач классификации и регрессии построением бинарного дерева решений. На каждом шаге построения дерева правило , формируемое в узле , делит обучающую выборку на две более однородные подвыборки:

Обычно вместо меры однородности используется противоположная по смыслу мера загрязненности. Пусть – некоторая вершина дерева решений, – подвыборка, связанная с этой вершиной, – загрязненность вершины. Необходимо потребовать, чтобы загрязненность вершины была равна , если содержит прецеденты только одного класса и была максимальной в случае, если содержит одинаковое число прецедентов каждого класса.

Одной из наиболее используемых является мера загрязненности вершины, формализованная в индексе Gini:

где – доля примеров класса в подвыборке .

**Оптимальное расщепление вершин**

Правило разбиения множества , связанное с каждой вершиной дерева решений, называется расщеплением. Бинарное расщепление вершины можно рассматривать как функцию , где в случае вектор относится к первому (левому) потомку, а в случае – ко второму (правому).

Расщепление подвыборки естественно осуществлять таким образом, чтобы максимально уменьшить загрязненность. Уменьшение загрязненности вершины для бинарных деревьев определяется как

где и – доли примеров подвыборки , соответствующие левому и правому потомкам ( и ). Наилучшим расщеплением вершины считается разбиение, которое максимизирует величину , т.е. расщепление выполняется таким образом, чтобы .

Определим величину , используя в качестве меры загрязненности вершины индекс Gini. Обозначим – размер подвыборки ; и – размеры подвыборок, соответствующих потомкам и ; и – число экземпляров класса в потомках и . Тогда наилучшим расщеплением вершины будем считать расщепление, которое минимизирует величину :

Так как задачу минимизации можно свести к задачи максимизации , получим следующее выражение для величины :

## Ансамбли классификаторов

Ансамбль классификаторов представляет собой множество классификаторов, чьи решения комбинируются некоторым образом для получения окончательной классификации наблюдений. Обычно синтез решений отдельных классификаторов, осуществляется путем их голосования (возможно, взвешенного). Основная причина, обусловившая интерес к данной тематике, состоит в том, что при определенных условиях ансамбли классификаторов обладают точностью, значительно превосходящей точность отдельных классификаторов и устойчивы к «зашумлению» обучающей выборки. Необходимым и достаточным условием высокой точности ансамбля классификаторов является то, чтобы составляющие его классификаторы были достаточно точны и различны, т.е. совершали ошибки на различных прецедентах. Существует несколько групп методов построения ансамблей классификаторов:

1. Манипулирование примерами обучающей выборки;
2. Манипулирование признаками;
3. Инъекция случайности в индуктивный алгоритм;
4. Байесовское голосование.

## Случайные леса

Случайный лес – статистический метод, предназначенных для решения задач классификации и регрессии. Понятие *случайный лес* впервые было введено в научный обиход в работах (6,7,8!). В этих статьях рассматривалось множество корневых лесов с помеченными вершинами, на котором задавалось равномерное распределение вероятностей. Позднее в статье (25!) был предложен новый метод классификации и регрессии, также получивший название *случайный лес*. В этом смысле термин *случайный лес* широко используется таких дисциплинах как машинное обучение, распознавание образов и, в меньшей степени, в прикладной статистике.

Случайные леса синтезируют методы построения ансамблей классификаторов первых трех групп. Для манипулирования примерами обучающей выборки используется метод, называемый баггингом (bagging), предложенный Брайманом (22). Он основан на формировании обучающей выборки для каждого классификатора ансамбля с помощью бутстрепа, т.е. случайной выборки (того же объема, что и исходная обучающая выборка) с возвращением из исходной обучающей выборки и использования метода голосования для агрегирования решений отдельных классификаторов.

В отличие от классических алгоритмов построения деревьев решений, в методе случайных лесов при построении каждого дерева на стадии расщепления вершин используется только фиксированное число случайно отбираемых признаков обучающей выборки и строится полное дерево (без усечения).

Алгоритм построения случайного леса может быть представлен в следующем виде:

1. Для (здесь – количество деревьев в ансамбле) выполнить

* Сформировать бутстреп выборку размера по исходной обучающей выборке ;
* По бутстреп выборке построить неусеченное дерево решений с минимальным количеством наблюдений в терминальных вершинах равным , рекурсивно следуя следующему подалгоритму:

1. из исходного набора признаков случайно выбрать признаков (в задачах классификации обычно );
2. из признаков выбрать признак, который обеспечивает наилучшее расщепление;
3. расщепить выборку, соответствующую обрабатываемой вершине, на две подвыборки;
4. В результате выполнения шага 1 получаем ансамбль деревьев решений ;
5. Классификацию новых наблюдений осуществлять следующим образом. Пусть – класс, предсказанный деревом решений , т.е. ; тогда – класс, наиболее часто встречающийся в множестве .

## Потоковые случайные леса

Обычно случайные обучаются до начала их тестирования (off-line режим), то есть фазы обучения тестирования разделены. Тем не менее на практике обучающие данные не могут быть даны заранее, а поступают последовательно (например в приложениях трекинга, где прогнозы необходимо получать на лету). В таких ситуациях алгоритм должен работать «на лету» (в on-line режиме). On-line обучение имеет большое число преимуществ перед off-line методами, например, требуется гораздо меньше памяти, так как записи(примеры) не нужно хранить, обучение происходит гораздо быстрее, off-line методы неприменимы, когда распределение, лежащее в основе меняется на протяжении времени.

Так как случайные леса представляют собой случайные деревья решений, объединённые в ансамбль посредством баггинга, то для реализации потоковых случайных лесов необходимо осуществлять как баггинг, так построение деревьев решений в on-line режиме.

## On-line баггинг

Известно, что каждая бутстреп выборка не содержит примерно 37% наблюдений исходной обучающей выборки (поскольку выборка с возвращением, то некоторые наблюдения в нее не попадают, а некоторые попадают несколько раз).

Пусть обучающая выборка имеет размер , число деревьев в ансамбле равно . Тогда при построении случайного леса необходимо создать бутстреп выборок размера .

Пусть случайная величина принимает значение 1, если пример попал в бутстреп выборку на позицию и значение 0 в противоположном случае. Соответствующие вероятности равны

Тогда - конечная последовательность независимых случайных величин, имеющих одинаковое распределение Бернулли с параметром , а случайная величина , являющаяся числом встреч примера в бутстреп выборке, имеет биномиальное распределение с параметрами и (). Вероятность того, что пример попадет в бутстреп выборку раз равна

При , где – распределение Пуассона с параметром ()

Для баггинга в On-line режиме используется метод, предложенный в работе (16), где последовательно прибывающие данные моделируются с помощью пуассоновского распределения. Каждое дерево обновляется на каждом примере раз подряд, где – случайное число, генерируемое пуассоновским распределением .

## Построение деревьев решений в on-line режиме

Обозначим за – статистику меток классов для вершины ; и – статистики меток классов для левого и правого потомков, разделенных по признаку .

Когда мы говорим об офлайн режиме, каждая вершина дерева решения имеет доступ ко всем данным, проходящим через нее. В онлайн режиме статистика собирается с течением времени, следовательно решение, когда необходимо разделять вершину зависит от следующих факторов:

* Достаточно ли примеров в вершине, чтобы иметь устойчивую статистику
* Достаточно ли хорошо разделение для достижения цели классификации

Так как статистика в последующих вершинах – потомках основывается на выборе значений этих факторов и поскольку ошибка на данной стадии не может быть скорректирована в дальнейшем, необходимо разработать метод который помоет сделать разделение целесообразным.

В работе saffari\_olc\_09 было предложено следующее решение. Вводятся 2 параметра:

1. Минимальное число примеров в узле до разделения
2. Минимальное значение уменьшения загрязненности вершины

После разделения вершины значения и предаются в левый и правый потомки соответственно. Таким образом, новый узел стартует со статистикой, накопленной в узле-родителе и может производить классификацию даже без получения новых данных.

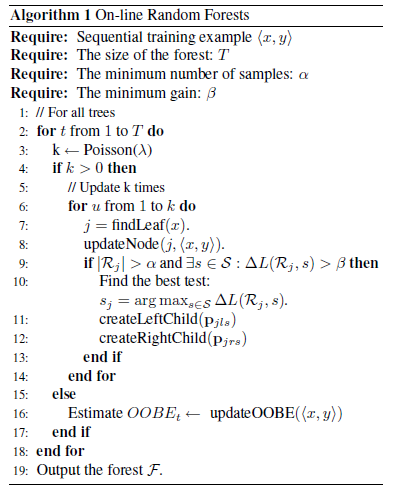
## OOBE (Out-Of-Bag-Error)

Одно из достоинств случайных лесов состоит в том, что для оценки вероятности ошибочной классификации нет необходимости использовать кросс-проверку или тестовую выборку. Оценка вероятности ошибочной классификации случайного леса осуществляется методом «Out-Of-Bag» (OOB), состоящем в следующем. Известно, что каждая бутстреп выборка не содержит примерно 37% наблюдений исходной обучающей выборки (поскольку выборка с возвращением, то некоторые наблюдения в нее не попадают, а некоторые попадают несколько раз). Классифицируем некоторый вектор . Для классификации используются только те деревья случайного леса, которые строились по бутстреп выборкам, не содержащим , и как обычно используется метод голосования. Частота ошибочно классифицированных векторов обучающей выборки при таком способе классификации и представляет собой оценку вероятности ошибочной классификации случайного леса методом OOB. Практика применения оценки OOB показала, что в случае, если количество деревьев достаточно велико, эта оценка обладает высокой точностью. Если число деревьев мало, то оценка имеет положительное смещение (31).

Также введем понятие OOB ошибки для одного дерева (). Она определяется, как частота ошибочной классификации некоторого вектора , не входящего в бутстреп выборку, на основе которой строилось дерево.

## On-line адаптация

Для таких приложений, как трекер, распределение примеров (записей!) может изменяться на протяжении времени. Следовательно, требуется иметь временное знание взвешивания, которое позволяет отучивать старую информацию. Если алгоритм работает по такому сценарию, мы можем позволить лесу исключать целые деревья. Отметим, что Пуассоновский процесс онлайн баггинга делает так, что некоторые деревья не участвуют в обучении на примере (записи) (при k=0). Следовательно, мы можем оценить ошибку OOB онлайн для каждого дерева (). Базируясь на этой оценке, мы предлагаем исключать деревья случайно из ансамбля, где вероятность исключения дерева зависит от . Поскольку в ансамбле деревьев влияние одного дерева относительно низко, то удаление то удаление одного дерева обычно не приносит вреда производительности целому лесу. Однако, непрерывное следование этому алгоритму гарантирует адаптацию на протяжении времени.



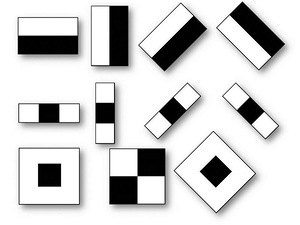
## Интегральное представление изображения

Интегральное представление изображения – это матрица , совпадающая по размерам с исходным изображением. Каждый элемент матрицы представляет собой сумму интенсивностей всех пикселей, находящихся левее и выше данного элемента, и рассчитывается по формуле

Интегральную сумму произвольной прямоугольной области можно вычислить, используя интегральное представление изображения следующим образом:

## Признаки Хаара

Признаки Хаара – это признаки цифрового изображения, представляющие собой прямоугольные области, состоящие из смежных прямоугольных подобластей. Своим названием они обязаны сходством с вейвлетами Хаара и были впервые описаны в работе (Viola Jones).



Значение двух-прямоугольного признака вычисляется, как разность между интегральными суммами пикселей в двух смежных прямоугольных подобластях. Для трех-прямоугольного признака значение вычисляется, как интегральная сумма двух внешних подобластей, вычитаемая из суммы в центральной подобласти. Значение четырех-прямоугольного признака вычисляется, как разность между суммами диагональных пар подобластей.

Рассмотрим вектор признаков , координаты которого представляют значения признаков Хаара, измеряемых в некоторой области, полученной в результате сегментации изображения. Значение признака примем равным единице в случае, если разность интегральных сумм подобластей неотрицательна, и нулю – в противоположном случае.

Написать о том, что у нас всего два класса: «цель» и «не цель».