Департамент образования муниципия Кишинев.

Лицей имени Н. М. Спэтару

Сектор информатики.

**Тема :**

**“Speed Supervisor”**

**программа для повышения безопасности в школах.**

Ученик 9Б

Мельник Роман

Преподаватель:

Майкова В.Э.

Кишинев, 2018

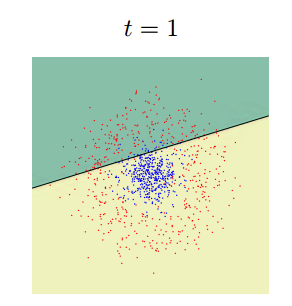
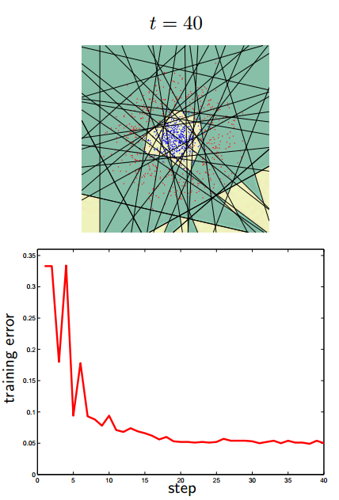
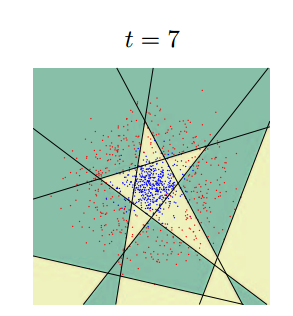
Содержание

1. Цели проекта.
2. Задачи проекта.
3. Принцип работы.
   1. Этапы.
   2. Детектирование.
      1. Метод Виолы-Джонса.
      2. Интегральное представление изображения.
      3. Гистограмма ориентированных градиентов.
   3. Трекинг.
   4. Вычисление скорости.
   5. Обработка результатов
4. Описание реализации проекта и способов применения.
5. Дальнейшие планы.
6. Библиография.

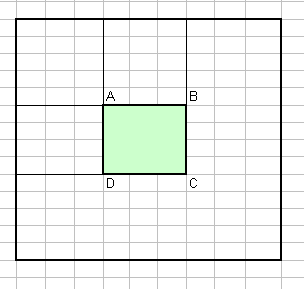
Принцип работы

Работу программы можно разделить на три части : поиск координат человека на кадре (детектирование), попытка установления соответствия найденных координат определенного человека с его же координатами на прошлом кадре(трекинг), расчёт условной скорости движения. Обнаружение объекта реализовано двумя способами : алгоритмом Виолы-Джонса и гистограммой ориентированных градиентов.

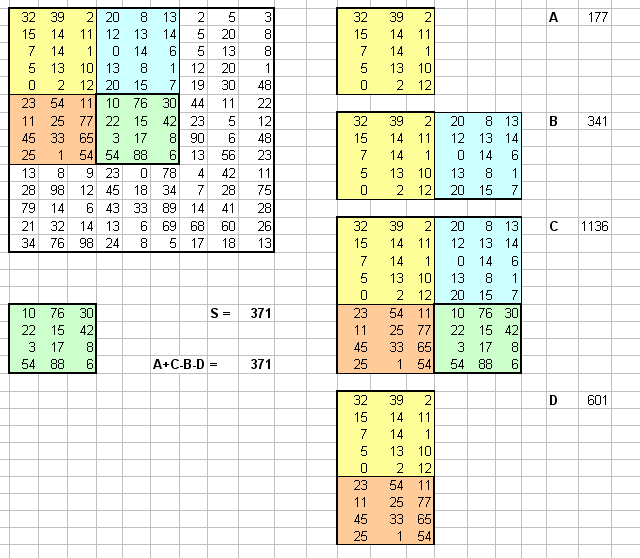
# Метод Виолы-Джонса

Метод Виолы-Джонса заключается в построении каскада слабых классификаторов на основе алгоритма AdaBoosting. Во время обучения из множества слабых классификаторов составляется один сильный. На картинках ниже представленно то, как увеличивается точность алгоритма со временем обучения на примере двумерного пространства.

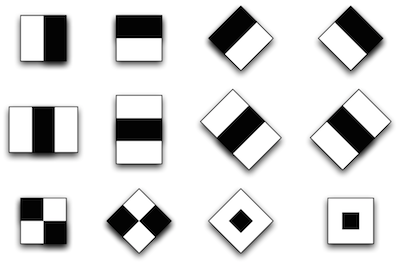
Для определения принадлежности к классу в каскаде, находиться сумма значений слабых классификаторов этого каскада. Каждый слабый классификатор выдает два значения в зависимости от того больше или меньше заданного порога значение признака, принадлежащего этому классификатору. В конце сумма значений слабых классификаторов сравнивается с порогом каскада и выносится решения найден объект или нет данным каскадом. Классом, в нашем случае, является человек. Для построения каскада используется обучающая выборка положительных примеров изображений, где объект присутствует, вместе с его координатами, и отрицательными, где объект отсутствует. Каждое изображение конвертируется в интегральное представление - матрицу, размерность которой совпадает с размерностью исходного изображения. Каждый элемент матрицы II(x,y) содержит в себе сумму пикселей изображения в прямоугольнике от (0,0) до (x,y). Одной из важнейших особенностей интегрального представления является возможность очень быстро вычислить сумму пикселей произвольного прямоугольника (или любой другой фигуры, которую можно аппроксимировать несколькими прямоугольниками).Например, интересующий нас прямоугольник – ABCD.



Тогда сумма пикселей внутри этого прямоугольника будет равняться II(A)+II(C)-II(D)-II(B). Это легко можно проверить в таблице Excel’я.



Данное свойство очень полезно, так как признаками для классификаторов служат примитивы Хаара, которые состоят из смежных прямоугольных областей. Они позиционируются на изображении, далее суммируются интенсивности пикселей в областях (благодаря интегральному представлению сделать это не составляет труда), после чего вычисляется разность между суммами закрашенных и незакрашенных областей. Эта разность и будет значением определенного признака, определенного размера, определенным образом спозиционированного на изображении.



Примеры примитивов Хаара.

Для каждого признака строится слабый классификатор, который дает наименьшую ошибку на всех тренировочных примерах.

Выходом алгоритма обучения является база классификаторов в формате XML, которая позже будет использоваться для алгоритма обнаружения. На его вход поступает изображение, которое затем конвертируется в интегральное представление, а результатом работы является массив прямоугольников R(x,y,w,h), определяющих положение найденных объектов на изображении. Алгоритм работает по принципу "скользящего окна". То есть рамка, размером, меньшим, чем исходное изображение, двигается с некоторым шагом по изображению, и с помощью каскада слабых классификаторов определяет, есть ли в рассматриваемом окне искомый предмет. Чем меньше рамка и шаг, тем точнее и медленнее детектирование.

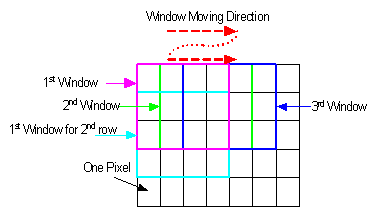
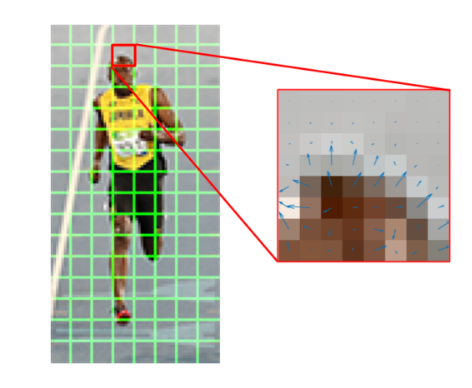


Схема движения скользящего окна

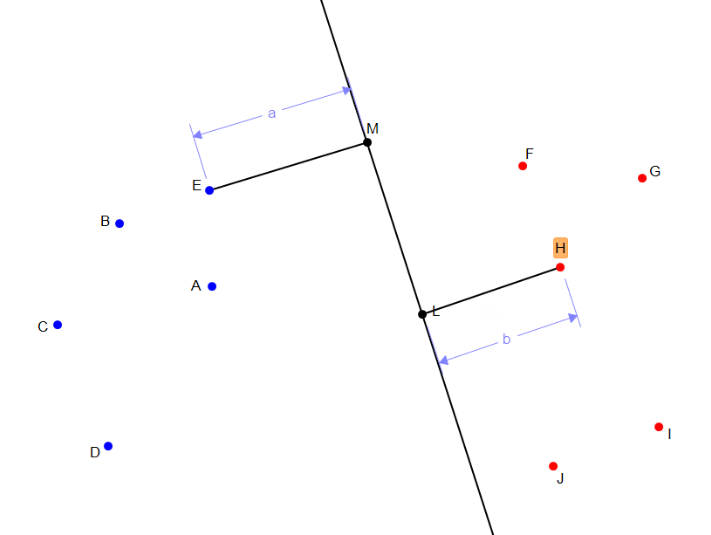
Плюсами метода Виолы-Джонса является очень высокая скорость распознавания при достаточно маленьком количестве ошибок, что дает возможность использовать его для поиска множества предметов на изображений в режиме реального времени.

# Гистограмма ориентированных градиентов (HOG)

Вторым методом для обнаружения людей на изображении является гистограмма ориентированных градиентов(HOG). Это простая процедура выделения признаков состоит в постройке для каждого пикселя градиента яркости, сравнивая его с соседними пикселями, которая показывает направление от светлых пикселей к более темным. Далее изображение разбивается на ячейки 8x8, внутри которых градиенты каждого пикселя группируются и ячейке присваивается стрелочка направленная в ту сторону, куда и большинство пикселей внутри неё. Таким образом мы рассматриваем только направление изменения яркости на изображении.



Пример разбивания изображения на ячейки

Следующим шагом является построение классификатора на основе гистограмм из обучающей выборки. В данном проекте в качестве линейного классификатора для HOG дескриптора используется метод опорных векторов(SVM), который проводит между двумя классами оптимальную разделяющую гиперплоскость, максимально удаленную от двух крайних элементов каждого класса. После чего можно использовать алгоритм обнаружения, который работает почти так же, как и описанный выше.

Разделение двух классов SVM

HOG позволяет получать более корректные и правдивые результаты, по сравнению с каскадами Хаара, однако и скорость работы этого алгоритма медленнее. Выбор метода детектирования зависит от конкретного случая использования программы.

#Доделать

После того, как мы получили расположения объектов на изображениях, необходимо их сопоставить. Этим занимается алгоритм трекинга(KCF). При инициализации на его вход поступают координаты объектов и изображение, на основе которых он обучает несколько моделей классификаторов. На следующих итерациях уже классификаторы алгоритма определяют положение конкретного объекта на изображении. В это же время фоновым процессом продолжается детектирование объектов. Если их количество, найденных детектором, и количество, найденных трекером, не совпадает, следовательно какой-то объект вышел за пределы обзора камеры, либо появился новый. В этом случае происходит реинициализация трекера, при которой классификаторы переобучаются на обнаружение новых объектов. Теперь мы имеем для каждого отдельного человека его координаты на протяжении некоторой секвенции изображений, и можно приступать к заключительному этапу работы программы. Основываясь на том, с какой скоростью и на сколько изменяются координаты человека на протяжении нескольких изображении можно вычислить некоторое значение, которое будет отражать его примерную скорость, по сравнению с остальными объектами. Опытным путем для каждого камеры, с которой будет вестись наблюдение, требуется установить порог этой величины, по пересечению которого можно считать, что человек бежит. Далее, если этот порог был превышен, программа сохраняет кадр, на котором это было сделано, и выделяет человека, чья скорость была слишком высока.