

Способ уменьшения вычислительной сложности процедуры обучения детектора лиц на базе метода Виолы-Джонса

Козыревский В.К.*, Веселов А.И.**

Санкт-Петербургский государственный университет

аэрокосмического приборостроения,

ул. Большая Морская, 67, Санкт-Петербург, 190000, Россия

**e-mail: vadikko2@mail.ru*

***e-mail: felix@vu.spb.ru*

Аннотация

Предложен способ уменьшения вычислительной сложности процедуры обучения в методе Виолы-Джонса за счет сокращения числа анализируемых признаков. Предлагаемый подход представляет собой модификацию алгоритма адаптивного бустинга. За счет предварительного анализа корреляции между значениями различных признаков формируются подмножества с некоррелированными признаками. На каждой итерации алгоритма обучения анализируются только классификаторы, использующие признаки из одного подмножества. Таким образом, уменьшение вычислительной сложности достигается за счет анализа только классификаторов с некоррелированными выходами. Предложенный алгоритм позволяет снизить сложность обучения алгоритма Виолы-Джонса, что является важной и актуальной задачей в системах детектирования объектов.

Ключевые слова: компьютерное зрение, детектирование лиц, метод Виолы-Джонса, бустинг, AdaBoost, Fast Feature Select, уменьшение сложности.

Введение

Возможность видеть и распознавать объекты – естественная и привычная возможность для человека. Однако для компьютера, пока что – это чрезвычайно сложная задача. Сейчас предпринимаются попытки научить компьютер хотя бы доле того, что человек использует в повседневной жизни, даже об этом не задумываясь. В реальных задачах, которые ставятся перед системами компьютерного зрения, факторов очень много. Определить их вручную и просчитать зависимости – непосильная для человека задача. Поэтому в таких случаях без машинного обучения, позволяющего подобрать зависимости между различными факторами автоматически, составляющего формулы, позволяющие принимать решения, не обойтись никак.

Детектирование лиц является важной задачей в таких областях, как компьютерное зрение, биометрия, организация видеоконференций, создание интеллектуальных систем безопасности и контроля доступа, и т. д. В свою очередь детектирование объектов является трудной задачей как с практической, так и с теоретической точек зрения. Одним из решений этой задачи является метод детектирования лиц на основе алгоритма Виолы-Джонса [1]. Хотя этот метод был разработан и представлен ещё в 2001 году Полом Виолой и Майклом Джонсом, он до сих пор является популярным алгоритмом для поиска объектов на изображении в реальном времени.

Алгоритм Виолы-Джонса включает в себя два основных шага: обучение и детектирование. Стадия детектирования не является сложной задачей, чего нельзя сказать об обучении. В случае метода Виолы-Джонса, речь может идти о длительности обучения порядка нескольких месяцев. Отсюда и появляется задача снижения этого времени, рассматриваемая в данной работе.

Детектирование лиц с использованием метода Виолы-Джонса

Перед описанием метода Виолы-Джонса рассмотрим задачу обучения в общем виде. Допустим, дано конечное множество объектов (прецедентов) по каждому из которых собраны некоторые данные – описание прецедента. В качестве таких описаний, чаще всего, используется признаковое пространство – совокупность из N показателей, присущая для всех прецедентов (в рамках статьи имеются в виду числовые признаки). В свою очередь совокупность таких описаний называется обучающей выборкой [2]. Будем считать, что объекты получены из некоторой неизвестной закономерности. Задача обучения сводится к восстановлению данной закономерности по обучающей выборке. Как правило, решение данной задачи сводится к поиску набора правил, указывающих как действовать при том или ином случае. Отличительной особенностью методов машинного обучения является то, что правила формируются автоматически. Отметим, что в некоторых источниках данные правила также называют алгоритмами, чтобы подчеркнуть, что они должны быть реализуемы в виде программы. В таком случае, задача обучения формулируется в терминах подбора таких параметров алгоритмов, при которых достигается оптимум некоторого критерия качества, показывающего, например,

число ошибок, допущенных алгоритмом на обучающей выборке. Таким образом, можно сделать вывод, что при решении задачи обучения некоторого алгоритма надо решить ряд подзадач.

1. Получить множество объектов, по которым будет восстанавливаться неизвестная зависимость. Сформировать обучающую выборку, включающую в себя признаковое описание объектов и требуемые выходы алгоритма (ответы).
2. Сформулировать критерий качества алгоритма.
3. Определиться с параметрическим семейством алгоритмов, с помощью которых будет восстанавливаться неизвестная закономерность.

После того, как эти подзадачи выполнены, выбирается метод обучения, с помощью которого делается подбор параметров. В результате применения метода обучения к обучающей выборке и параметрическому семейству алгоритмов, формируется итоговый алгоритм, способный предсказывать ответы на новых объектах, полученных из закономерности [3].

Перейдем к рассматриваемой в данной работе задаче классификации лиц на изображениях. Будем считать, что обучающая выборка представлена множеством изображений фиксированного размера, которые либо содержат изображение лица, либо нет. Если на изображении представлено лицо, тогда:

- оно занимает существенную часть изображения;
- никакие другие объекты на изображении не присутствуют.

Необходимо построить алгоритм, способный для нового изображения выдавать бинарное решение о том, присутствует ли на нем лицо.

В настоящее время одним из популярных способов решения данной задачи является метод Виолы-Джонса, в котором строится так называемый «сильный» классификатор, представляющий собой алгоритмическую композицию «слабых» классификаторов:

$$a_s(x) = \sum_{i=1}^T b_i a_i(x), \quad (1)$$

где x – изображение; $a_i(x)$ – слабый классификатор, т.е. любой алгоритм, решающий задачу классификации чуть лучше, чем случайное угадывание; b_i – весовой коэффициент i -го слабого классификатора в композиции; $a_s(x)$ – сильный классификатор.

Ключевыми идеями метода Виолы-Джонса являются:

- способ выбора слабых классификаторов, а также способ их эффективного с вычислительной точки зрения применения;
- использование алгоритма *Adaboost* для формирования алгоритмической композиции.

Применение этих идей позволяет построить детектор, способный работать в режиме реального времени. За счет своей скорости и эффективности, данный детектор является одним из популярных методов для поиска лиц на изображениях. Рассмотрим основные аспекты данного алгоритма.

Признаковое описание прецедентов

Для описания прецедентов, как уже было сказано, используется признаковое пространство [4]. В рассматриваемом алгоритме Виолы-Джонса используются признаки Хаара (рисунок 1) [5]. Признак Хаара – это результат расчёта свёртки между изображением и некоторой двоичной маской заданной формы и размера. Выбор Виолы и Джонса пал на признаки Хаара с учётом наблюдения, что область вокруг глаз темнее области на носу, область рта темнее области вокруг рта, и т. д. Области на масках выбираются из тех же соображений (рисунок 2).

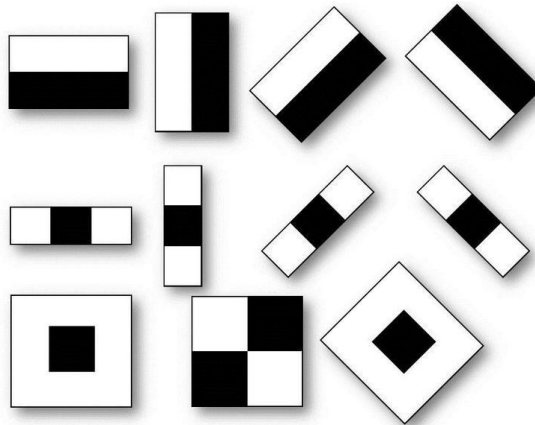


Рисунок 1 – Внешний вид признаков Хаара

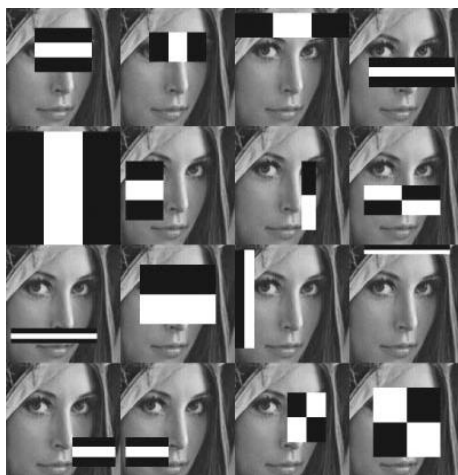


Рисунок 2 – Расположение признаков Хаара на изображениях

При расчёте значения признака маска позиционируется на изображении, после чего вычисляется разность между суммами значений яркостей пикселей, лежащими под областями разного цвета. Эта разность и будет числовым значением признака (откликом). Таким образом, значение признака рассчитывается по формуле:

$$f_j(x) = W - B,$$

где j – номер признака, x – элемент выборки, W – сумма значений яркости пикселей под белой областью признака Хаара, B – сумма значений яркости пикселей под чёрной областью признака Хаара.

Предположим, что размер изображения (одного элемента выборки) 19x19 пикселей, тогда существует порядка 65 тысяч признаков Хаара для конкретной маски, а число масок равно 10. В таком случае размер множества (длина признакового описания объекта) всех возможных признаков Хаара будет равен 650 тысяч.

Построение алгоритмической композиции с помощью алгоритма AdaBoost

В настоящее время использование алгоритмов, основанных на композициях простых классификаторов, является одним из наиболее распространенных подходов к применению методов машинного обучения на практике. Бустингом принято называть процедуру последовательного построения цепочки классификаторов, в которой каждый последующий классификатор исправляет ошибки предыдущих [6]. *AdaBoost* (Adaptive Boosting, адаптивный бустинг) был предложен в работе [7] для решения задачи двоичной классификации. В последующем данный подход был расширен на другие задачи. Отличительной особенностью *AdaBoost* является критерий качества алгоритма для гладкой аппроксимации ошибки классификации используется взвешенная экспоненциальная функция потерь:

$$E((x_i, y_i)_{i=1}^n) = \sum_{i=1}^n w_i I(a_s(x_i) \neq y_i) \leq \sum_{i=1}^n w_i e^{-y_i a_s(x_i)}, \quad (2)$$

где w_i – «вес» ошибки на объекте i ; y_i – правильный ответ для объекта i .

Веса w_i показывают значимость объекта при выборе слабого классификатора. Модификация весов на каждой итерации позволяет выбирать слабые классификаторы, исправляющие разные ошибки. Использование функции (2) позволяет в явном виде выписать последовательность шагов для поиска оптимальных параметров.

Детальное формальное описание *AdaBoost* выходит за рамки данной работы. Для пояснения идеи *AdaBoost* продемонстрируем его работу на задаче разделения объектов двух классов на плоскости (рисунок 3). Предположим, что имеется двумерное пространство, содержащее в себе объекты различных классов с метками

(1, -1). Будем считать, что слабым классификатором является любая прямая линия, отделяющая объекты разных классов друг от друга с вероятностью ошибки меньше 50%. Композиция таких прямых линий, в свою очередь, будет называться сильным классификатором: ошибка композиции будет существенно меньше, чем у каждого слабого классификатора по отдельности.

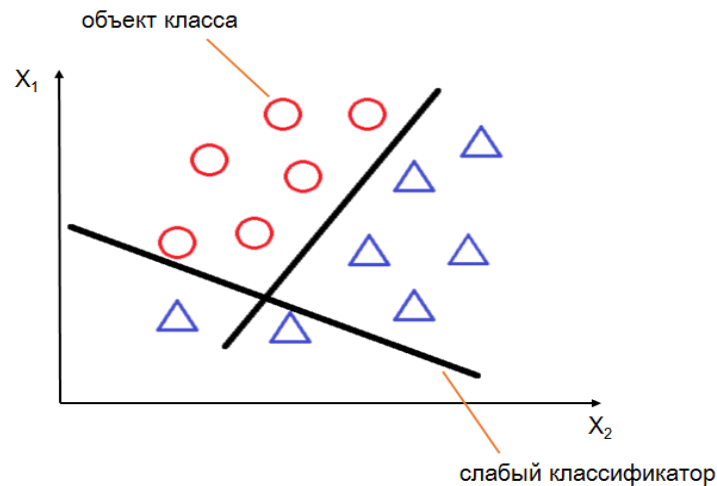


Рисунок 3 – Плоскость с объектами разных классов

На каждой итерации AdaBoost выполняет следующую последовательность шагов.

1. Выбирает из всего множества слабых классификаторов один, обеспечивающий минимальную взвешенную ошибку разделения объектов разных классов.
2. Определяет позиции ошибочно классифицированных объектов и увеличивает их вес.
3. Определяет значимость слабого классификатора в композиции (b_i в формуле (1)).

Возможные шаги работы данного алгоритма представлены на Рисунке 4.

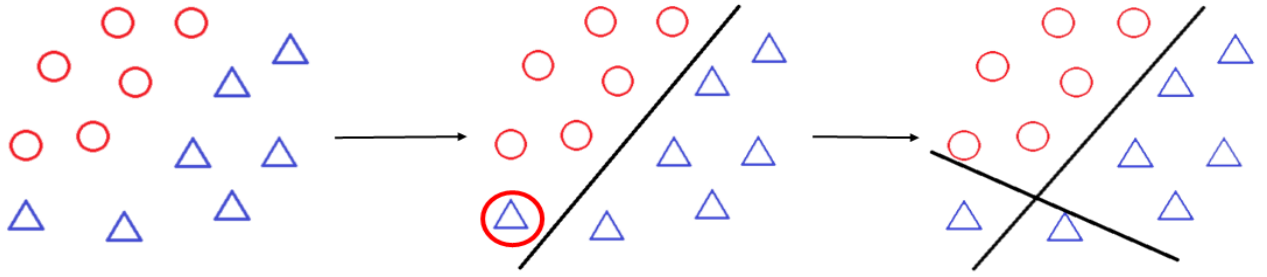


Рисунок 4 – Общий вид работы алгоритма *AdaBoost*

Отметим, что на каждой итерации *AdaBoost* на шаге 1, в общем случае, должен перебрать все возможные слабые классификаторы. В случае разделения объектов на плоскости прямыми линиями существуют эффективные алгоритмы, позволяющие не выполнять подобный перебор в явном виде. Однако, с ростом сложности слабых классификаторов и увеличением размерности признакового пространства подобные алгоритмы уже не могут быть применены и задача выбора слабого классификатора становится очень вычислительно сложной. Уменьшению сложности выбора слабого классификатора и посвящена данная работа.

Введём в рассмотрение понятие слабого классификатора в методе Виолы-Джонса. В данном случае слабый классификатор задаётся как

$$h_j(x, \theta_j) = [|f_j(x)| < \theta_j],$$

где j – номер признака, $f_j(x)$ – значение признака, x – элемент выборки, θ_j – пороговое значение, $[]$ – индикаторная функция. Очевидно, что множество возможных слабых классификаторов определяются декартовым произведением множества признаков и множества возможных порогов. Что означает, что подобный перебор не может быть выполнен на практике. Однако можно заметить, что для определения оптимального слабого классификатора не требуется перебирать все

возможные пороговые значения θ . Учитывая конечность обучающей выборки, можно выписать все возможные значения θ для неё. Данное наблюдение использует в алгоритме *Fast Feature Select (FFS)* [8].

FFS

Одним из методов уменьшения сложности подбора порогового значения для одного классификатора является алгоритм *Fast Feature Select (FFS)*, который перебирает не все возможные границы, а только некоторое «допустимое» подмножество. Предположим, размер обучающей выборки составляет N изображений. Тогда для некоторого признака с индексом j есть конечное множество значений $\{v_1^{(j)}, \dots, v_N^{(j)}\}$. Отсортируем значения признаков и получим переупорядоченный вектор значений $(v_{i_1}^{(j)}, \dots, v_{i_N}^{(j)})$. Очевидно, что если верно условие $v_{i_k}^{(j)} \leq \theta_1, \theta_2 \leq v_{i_{k+1}}^{(j)}$ для одного целого k и двух различных θ_1 и θ_2 , выбор границы признака для отклика, равного θ_1 , будет иметь то же самое влияние, что и выбор признака для отклика θ_2 . Таким образом, не нужно проверять все возможные значения признака. Достаточно проверить $N + 1$ возможное значение, чтобы найти оптимальное значение границы θ .

Приняв в рассмотрение описанный выше подход, алгоритм *FFS* можно описать следующим образом. Вход: признаковое описание $V = \{v_i^{(j)}\}$, $Y = \{y_i\}$ – метки принадлежности элементов из множества объектов к тому или иному классу, значения весов $w_{t,1}, \dots, w_{t,N}$ на итерации t и j – индекс слабого классификатора, для которого необходимо определить оптимальный порог.

Отсортируем значения признаков следующим образом:

$$v_{i_1}^{(j)} \leq \dots \leq v_{i_N}^{(j)}$$

Инициализируем начальное значение ошибки как:

$$e = \sum_{i=1}^N w_{t,i} * y_i.$$

Таким образом, если выбирается граница меньшая значения v_1 (когда классификатор ошибается на всех изображениях, где есть искомый объект) алгоритм пересчёта значения ошибки выглядит следующим образом:

for $k = 1$ *to* N

if $y_{i_k} = 1$

$$e = e - w_{i_k}, e_{i_k} = e$$

else

$$e = e + w_{i_k}, e_{i_k} = e$$

Оптимальная граница, в таком случае, может быть найдена как

$$q = \mathit{arg} \min_i e_i,$$

$$\theta = v_q^{(j)}$$

Отметим, что данный алгоритм уменьшает сложность подбора границы для одного слабого классификатора, но никак не упрощает перебор всех возможных классификаторов.

Оценим сложность алгоритма *AdaBoost* с учётом *FFS*. Пусть T — количество слабых классификаторов в итоговой композиции, M — количество всех возможных признаков на одном элементе выборки, а N — размер обучающей выборки. Видно,

что сложность выбора всех возможных признаков составляет $O(MN)$. Тогда, сложность алгоритма *AdaBoost* с учётом *FFS*:

$$O(TMN \log N + MN).$$

Предлагаемый алгоритм

Необходимо отметить тот факт, что признаковое описание объектов обладает высокой избыточностью: есть признаки с высокой корреляцией. С точки зрения алгоритма *AdaBoost* данный факт приводит к тому, что на этапе выбора оптимального слабого классификатора проверяются классификаторы, несущие избыточность (дающие схожий результат). Чтобы продемонстрировать это утверждение, построим график зависимости значений признаков от элементов выборки (рисунок 5). Данный график построен для изображений из тестового множества CBCL Database Face # 1 [9].

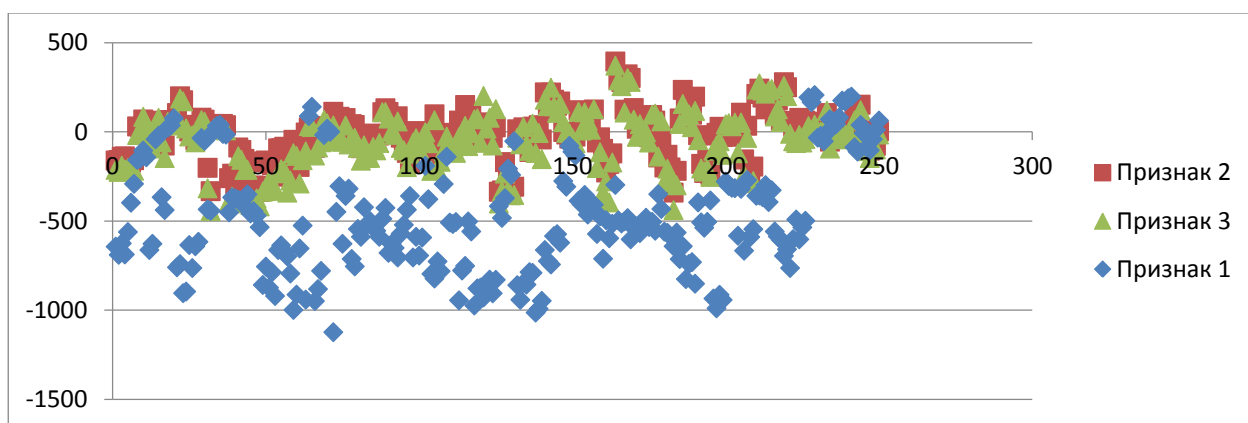


Рисунок 5 – График зависимости значений признаков от элементов выборки

Для оценки силы связи значений различных признаков друг от друга воспользуется коэффициентом корреляции Пирсона (3)[10].

$$r_{XY} = \frac{\sum(x-\bar{X})(Y-\bar{Y})}{\sqrt{\sum(x-\bar{X})^2} \sqrt{\sum(Y-\bar{Y})^2}}, \quad (3)$$

где $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$, $\bar{Y} = \frac{1}{n} \sum_{i=1}^n Y_i$ – среднее значение выборки.

Применив формулу (9) к кривым, представленным на Рисунке 4, получим, что коэффициент корреляции между вторым и третьим признаком равен 0,95, а между вторым и первым 0,25.

Допустим, на итерации $t - 1$ алгоритма *AdaBoost* для классификатора был выбран второй признак. Т. к. коэффициент корреляции между вторым и третьим признаками равен 0,95, то можно не учитывать третий признак на t -й итерации, так как он не будет исправлять ошибки, допущенные на предыдущей итерации.

В обоснование этих утверждений приведём гистограмму коэффициентов корреляции между признаками, по которым *AdaBoost* строил слабые классификаторы (рисунок б) при обучении детектора на основе метода Виолы-Джонса. По гистограмме видно, что значения корреляции между выбираемыми признаками колеблются в диапазоне от $-0,5$ до $+0,5$. Это означает, что *AdaBoost* действительно выбирает слабые классификаторы на основе некоррелированных признаков. Следовательно, можно применить следующий подход. Необходимо оценить связь между всеми признаками и не учитывать на текущей итерации те, которые не позволяют исправлять ошибки, допущенные на предыдущей итерации.

Для решения этой задачи необходимо на основе признакового описания $V = \{v_i^{(j)}\}$ построить множество списков $L = \{l_1, \dots, l_M\}$, содержащих в себе индексы признаков, некоррелированных с текущим, т.е.

$$l_i = \{k: r_{k,i} < thr, k = 1..M\},$$

где $r_{k,i}$ – значение корреляции, thr – некоторый, заранее определённый порог.

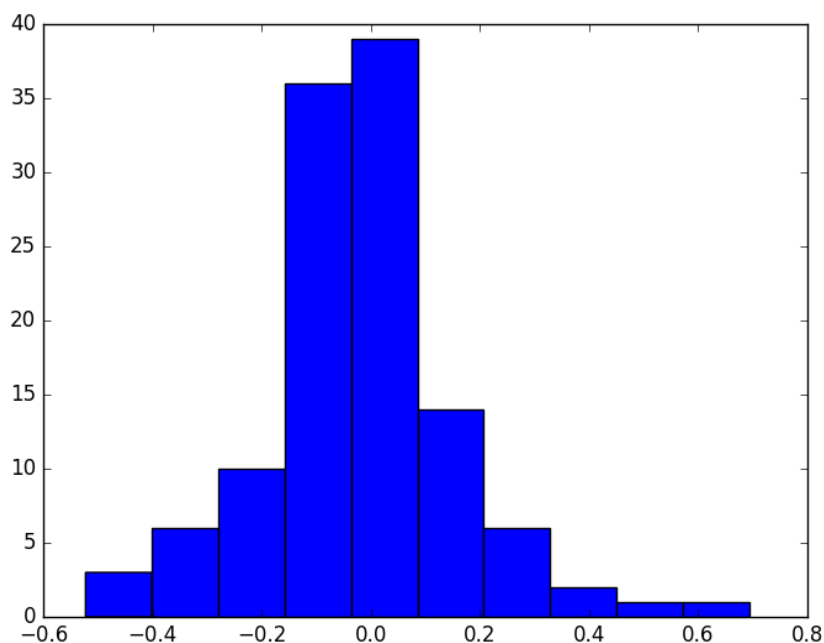


Рисунок 6 – Гистограмма коэффициентов корреляции от номера признака

Подобное представление позволяет не выбирать на стадии работы алгоритма *AdaBoost* $t + 1$ -й слабый классификатор из всего множества слабых классификаторов.

Предположим, что был выбран k -й слабый классификатор. Предположим, что k – индекс признака, на основе которого он был построен. Это означает, что следующий слабый классификатор будет выбираться на основе признака, индекс которого содержится в списке l_k , размер которого $|l_k| < M$.

Заключение.

В данной работе был рассмотрен детектор лиц на базе метода Виолы-Джонса. Приведены результаты анализа процедуры обучения детектора, из которых видно, что наибольшей вычислительной сложностью обладает стадия выбора слабого

классификатора. Было отмечено, что уменьшение сложности данной стадии возможно за счёт уменьшения размера множества признаков, перебираемых при построении композиции алгоритмов. Приведены результаты эксперимента, подтверждающие возможность уменьшения количества анализируемых слабых классификаторов за счёт исключения из рассмотрения на конкретной итерации признаков, обладающих высокой корреляцией. Полученные результаты позволили сформулировать подход к уменьшению сложности алгоритма *AdaBoost*.

В дальнейшем планируется разработка программного комплекса, реализующего улучшенный алгоритм обучения с целью его апробации на практике.

Работа выполнена при финансовой поддержке Минобрнауки России в рамках базовой части государственного задания вузу на 2016 год (проект № 2452).

Библиографический список

1. Viola P., Jones M. Robust Real-time Object Detection // International Journal of Computer Vision. 2004. Vol. 57. № 2. Pp. 137-154.
2. Вьюгин В.В. Математические основы теории машинного обучения. - М.: МЦНМО, 2013. - 387 с.
3. Местецкий Л.М. Математические методы распознавания образов - М: МГУ, ВМиК, 2002–2004. С. 42–44.
4. Hastie T., Tibshirani R., Friedman J. The elements of statistical learning. – New York: Springer, 2001. - 533 p.

5. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features // Accepted conference on computer vision and pattern recognition. Cambridge. 2001.
URL: <http://www.cs.cmu.edu/~srini/15-829/readings/ViJo01.pdf>
6. Freund Y., Schapire R.E. A decision theoretic generalization of on-line learning and an application to boosting // Journal of Computer and System Sciences. 1997. Vol. 55. No. 1. Pp. 119-139.
7. Wu J, Brubaker S., Mullin M., Rehg J. Fast Asymmetric Learning for Cascade Face Detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2008. Vol. 30. № 3. Pp.369-382.
8. CBCL Face Database #1, MIT Center For Biological and Computation Learning, 2000. URL: <http://cbcl.mit.edu/software-datasets/FaceData2.html>
9. Шмойлова Р.А. Общая теория статистики. - М.: Финансы и Статистика, 2002.
– 560 с.