

Интеллектуальные системы управления, анализ данных

© 2022 г. А.М. МИХАЙЛОВ, канд. техн. наук (alxmikh@gmail.com),
М.Ф. КАРАВАЙ, д-р техн. наук (mkaravay@yandex.ru),
В.А. СИВЦОВ (TheDeGe@yandex.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

МГНОВЕННОЕ ОБУЧЕНИЕ ПРИ РАСПОЗНАВАНИИ ОБРАЗОВ

Одним из основных недостатков искусственных нейронных сетей является медленное обучение, связанное с необходимостью вычисления большого количества коэффициентов. В статье показано, что обучение можно намного ускорить. Ускорение достигается за счет резкого сокращения числа обучающих образов. Кроме того, как для формирования признаков, так и для последующего распознавания объектов использован метод обратных образов, позволяющий обойтись без коэффициентов, что существенно сокращает объемы вычислений. При мгновенном обучении, как и при глубоком обучении, признаки формируются автоматически. Проведенные вычислительные эксперименты показали инвариантность предложенного метода не только к масштабированию и вращениям, но и к значительным деформациям распознаваемых объектов.

Ключевые слова: распознавание образов, машинное обучение, глубокое обучение, обратные образы, многомерное индексирование.

DOI: 10.31857/S0005231022030102

1. Введение

Традиционно, в задачах распознавания неизвестный образ \vec{x} классифицируется как принадлежащий к одному из N классов. Для этого вычисляются вероятности этих классов $p_1(\vec{x}), p_2(\vec{x}), \dots, p_N(\vec{x})$, после чего образ \vec{x} относится к классу с максимальной вероятностью. Существует много способов нахождения вероятностей классов, например байесовские классификаторы, нейронные сети и т.д. В частности, в нейронных сетях вероятности классов пропорциональны активностям выходных узлов. Поэтому на класс объекта указывает максимально активный узел. В отличие от нейронных сетей, где активность узлов определяется значениями коэффициентов, найденными при обучении, в данной статье коэффициенты не используются. Это становится возможным в результате использования метода обратных образов (МОО) [1], позволяющего находить сходство входного образа с заданными классами образов. Систему распознавания на основе метода МОО будем называть индекстроном.

Идея обратных образов аналогична идее обратных списков, используемых в методе TF-IDF [2]. Другими словами, если признак встречается во всех классах, то роль этого признака для оценки релевантности образа равна нулю. Однако веса признаков в предложенном в статье методе не вычисляются вообще. Также существуют другие отличия, в частности, отметим следующие моменты:

а) В то время как в поисковиках на вход поступает текст, в данной статье поиск проводится по числам-признакам, каждое из которых может изменяться в широком диапазоне в условиях шумов.

б) В поисковиках адреса, т.е. числа, в которые переводятся слова — это адреса единой адресной области. В данной статье создаются несколько адресных областей, число которых равно размерности вектора признаков, представляющего образ.

в) В поисковиках отсутствует понятие радиуса обобщения, который регулирует степень генерализации. В результате в поисковиках автоматически не создаются классы документов, а модели векторного пространства требуют использования дополнительных средств кластеризации. В данной статье в зависимости от радиуса обобщения число классов образов определяется автоматически в зависимости от заданной степени обобщения.

г) Поисковики автоматически не решают задачи предсказания. В данной статье предложенный метод обращения образов позволяет единообразно решать как задачу поиска ближайшего образа, так и задачу предсказания времени событий.

В данной статье при классификации объектов изображений с произвольным задним планом использован двухуровневый классификатор. На первом уровне в качестве признаков используются пиксели, а на втором уровне — частоты классов выделенных пикселей. При этом использование частотного порога переводит классы в разряд бинарных объектов — либо класс присутствует в описании объекта, либо отсутствует. При распознавании для выделения пикселей объектов на изображениях с произвольным задним планом используются гистограммы классов пикселей объектов, которые создаются при обучении. В таких гистограммах представлены существенные классы, т.е. классы пикселей с частотами выше пороговой. При распознавании порог частоты может понижаться в 2 раза, если существенных классов не найдено при верхнем значении порога. Критерием выделения пикселя является его принадлежность к существенному классу. Такие пиксели группируются в кластеры с помощью алгоритма окраски связанных объектов, т.е. объектов, пиксели которых отстоят друг от друга не более чем на заданный радиус.

Кластеры последовательно распознаются на 2-м уровне, входными признаками которого служат существенные классы в бинарной форме. Выходом 2-го уровня служит гистограмма классов второго уровня, где класс второго уровня — это идентифицированное имя неизвестного объекта, представленного кластером выделенных точек. Если на изображении без объекта был ошибочно найден объект, то классы пикселей, приведшие к ошибке, ингиби-

руются, т.е. удаляются из гистограммы классов пикселей, представляющих ошибочно найденный объект.

Это возможно, так как для обучения каждому классу достаточно одного разностного изображения. Под разностным изображением, называемым также подписью объекта, понимается изображение с объектом минус изображение без объекта. Для получения разностного изображения исходные кадры 800×600 редуцируются в кадры 400×300 с помощью фильтра (см. подраздел 2.8), позволяющего компенсировать возможное смещение сравниваемых изображений, возникающее из-за малых сдвигов камеры. Если объект движется, то для получения разностного изображения достаточно двух кадров. Такой подход позволяет обойтись при обучении без сотен и тысяч изображений объекта одного класса — объекта, снятого при разных ракурсах и масштабах. Одного кадра достаточно во многих приложениях, когда цветовая гамма пикселей объекта мало зависит от ориентации, например от вида кошки сзади, спереди или сбоку. Очевидно, что при использовании разностного изображения задний план пропадает и обучение ведется только по пикселям объекта.

В статье описывается эксперимент с изображениями цветов и кошек с произвольным задним планом. Результаты представлены и аннотированы в разделе 3, п. 1.

Для сравнения производительности МОО и искусственных нейронных сетей было выбрано еще одно приложение, в котором использован набор данных CIFAR-10 [3]. (См. раздел 3, п. 2). Комментарии представлены в разделе Заключение. Подробно МОО рассмотрен в разделе 2.

2. Материалы и методы

2.1. Извлечение признаков

В приложении, связанном с распознаванием трехмерных объектов, использовались признаки 2 уровней. На первом уровне признаками служили красная, зеленая и синяя составляющие цвета каждого пикселя, образующие трехмерный вектор. Каждый такой вектор использовался для классификации пикселя. На втором уровне признаками служили классы пикселей, а именно: классы, выделяемые с помощью гистограммы классов пикселей распознаваемого объекта. При этом класс включался в множество признаков второго уровня, если его частота превышала установленный порог.

2.2. Классификация образов

При классификации образов, как правило [4], неизвестный вектор \vec{x} относится к классу с наибольшей условной вероятностью $p_1(\vec{x}), p_2(\vec{x}), \dots, p_N(\vec{x})$. В настоящей статье для нахождения сходства входного образа с заданными классами используется понятие обратных образов, введенное в [5] и рассмотренное также в [1]. Этот метод напоминает технологию обратных файлов, используемую в поисковой системе Google [6] и в подходе “bag of words” [7].

В то время как в подходе [6, 7] для поиска объектов информация преобразуется в текст, индексстрон оценивает сходство классов, анализируя обратные образы числовых признаков.

В задачах классификации обратный образ признака x — это множество $\{n\}_x$ классов n , связанных с этим признаком. Класс n связан с признаком x , если x является элементом образа из этого класса. Например, для двух классов А, А, А и Т, Т признак “горизонтальная линия” участвует как в А-, так и в Т-классе, тогда как признак “вертикальная линия” участвует только в классе Т:

$$\{A, T\}_{0^\circ}, \quad \{T\}_{90^\circ}.$$

Пусть N классов образов представлены строками матрицы, где индексы строк n указывают на классы образа, а элементы строки x_{nm} — это признаки образа:

$$[x_{nm}]_{N \times M}.$$

Определение 1. Обратный образ $\{n\}_{x,m}$ признака x_{nm} — это множество классов n , связанных с этим признаком, т.е.

$$(1) \quad \{n\}_{x,m} = \{n : |x_{nm} - x| \leq R\}.$$

Здесь R — радиус обобщения, обсуждаемый в подразделе 2.5.

Распределение связей образа $\vec{x} = (x_1, x_2, \dots, x_m, \dots, x_M)$ с классами — суперпозиция

$$(2) \quad F_n^{(R)}(\vec{x}) = \sum_{m=1}^M f_n^{(R)}(x_m), \quad n = 1, 2, \dots, N.$$

Здесь $f_n^{(R)}(x_m) = 1$, если $n \in \{n\}_{x,m}$, и $f_n^{(R)}(x_m) = 0$, если $n \notin \{n\}_{x,m}$, — функция связанности признака x_m с классом n .

2.3. Определение класса

Можно определить класс с помощью обобщенного расстояния Хэмминга между векторами \vec{x} и \vec{y} . Такое расстояние определяется как количество позиций, где соответствующие координаты отличаются более чем на радиус обобщения R . Например, $H_{R=1}(\vec{x}, \vec{y}) = 1$, если $\vec{x} = (1, 2, 3)$, $\vec{y} = (2, 2, 5)$.

По определению класс образа, который представлен вектором \vec{x}_n , включает все векторы \vec{y} такие, что $H_R(\vec{x}_n, \vec{y}) = 0$. Следовательно, любые две строки x_n, x_m матрицы $[x_{n,k}]_{N \times K}$ принадлежат разным классам, если $H_R(\vec{x}_n, \vec{x}_m) > 0$.

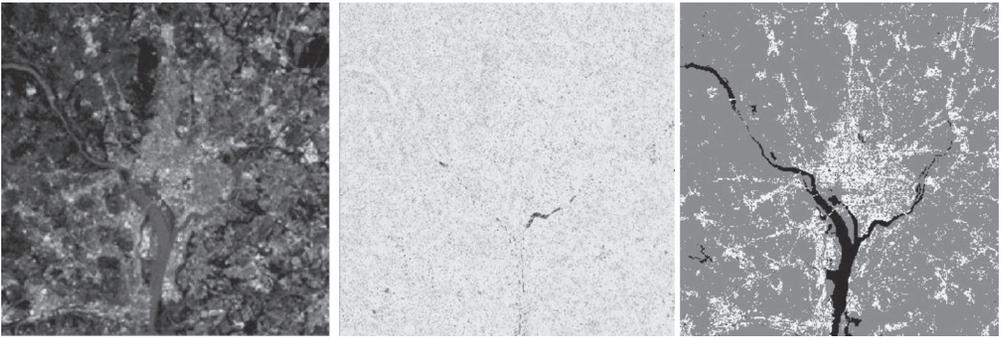


Рис. 1. Пример сегментации изображения.

2.4. Обучение

Чтобы получить распределение связей образа, его признаки x_m , $m = 1, 2, \dots, M$, использовались в качестве адресов списков классов, называемых колонками. Распределение (2) рассчитывается как гистограмма классов, находящихся в колонках. При обучении если $F_n^{(R)}(\vec{x}) < M$, то вектор \vec{x} не может быть классифицирован и должен рассматриваться как представитель нового класса N^+ . При этом обратные образы обновляются как $\{n\}_{xm} = \{n\}_{x,m} \cup N^+$, $m = 1, 2, \dots, M$, $N^+ = N + 1$.

Если в процессе обучения N превышает предел N_{\max} , то память обнуляется и значение радиуса обобщения увеличивается. Обычно $R = 5\text{--}20\%$ от диапазона признаков X . При этом степень обобщения растет с ростом R .

2.5. Радиус обобщения

Влияние радиуса обобщения R проиллюстрировано на рис. 1. В этой задаче сегментации требуется пиксели левого изображения разделить на три класса (вода, растительность, городские строения) и раскрасить так, как показано на правом изображении. Каждый пиксель на левом рисунке представлен 4-мерным вектором, компоненты которого принадлежат 256-уровневым диапазонам красного, зеленого, синего и инфракрасного спектров (показан только один компонент, исходные изображения см. в [8]). (По техническим причинам на рисунке красному, зеленому, синему цветам соответствуют белый, серый, черный цвета).

На изображении в центре результат сегментации при $R = 0$. При этом большинство пикселей остаются неклассифицированными. На правом изображении показана качественная сегментация, достигнутая при $R = 10\%$.

2.6. Алгоритм индекстрона

Обучение начинается с распознавания. Если входной вектор $\vec{x}(m)$, $m = 1, 2, \dots, M$, не распознан, то он в инверсном виде включается в обратные множества, становясь представителем нового класса. Поэтому сначала рассмотрим алгоритм распознавания.

Алгоритм распознавания. Значения компонент входного вектора $\vec{x}(m)$, $m = 1, 2, \dots, M$, используются в качестве адресов обратных образов — образов, являющихся множествами классов. Адресация к обратным образам производится для вычисления частот f_n , $n = 1, 2, \dots, N$, встречаемости классов в обратных образах $\{n\}_{x_m}$. Вычисление выполняется в следующем цикле:

$m = 1, 2, \dots, M$ для всех r из $[-\varepsilon, \varepsilon]$ и для всех n из $\{n\}_{x_{m+r}, m}$: $f_n = f_n + 1$.

Критерий принятия решения о классе i входного образа:

$$i : f_i = TM,$$

где T ($0 < T \leq 1$) — порог распознавания.

Если критерий не выполнен, то входной вектор инверсируется и вводится в обратные образы в качестве нового класса. Введение выполняется следующим образом. Если уже были созданы N классов, то запускается следующий алгоритм обновления обратных множеств:

1) если $N < N_{\max}$, то $N = N + 1$. В противном случае СТОП.

2) $m = 1, 2, \dots, M$: $\{n\}_{x(m), m} = \{n\}_{x(m), m} \cup N$.

Пример. Сначала набор обратных множеств пуст, т.е. $N = 0$. Затем, при поступлении 1-го вектора $\vec{x}(m)$, $m = 1, 2, \dots, M$, на первом шаге увеличивается на единицу число созданных классов: $N = 1$. На втором шаге это новое число добавляется в соответствующие обратные множества. В результате будет создан первый набор, содержащий M обратных множеств $\{1\}_{x(m), m}$, $m = 1, 2, \dots, M$.

Предположим, что при поступлении второго вектора $\vec{y}(m)$, $m = 1, 2, \dots, M$, значения всех его координат, за исключением первой, совпадают по абсолютной величине с координатами первого вектора с точностью до R . Тогда набор обратных множеств на второй итерации принимает вид: $\{1\}_{x(m), 1}$, $\{2\}_{y(m), 1}$, $\{1, 2\}_{y(m), m}$, $m = 2, \dots, M$. В этом наборе множество $\{1\}_{x(m), 1}$ осталось от первой итерации, множество $\{2\}_{y(m), 1}$ появилось впервые, а $M - 1$ множеств $\{1, 2\}_{y(m), m}$, $m = 2, \dots, M$, обновились, включив в свой состав второй элемент.

2.7. Архитектура индекстрона

Основными параметрами индекстрона являются размерность образа M , диапазон признаков X и максимальное количество классов N , которые могут быть им созданы. Другими словами, индекстрон содержит M групп адресов, где каждая группа m содержит X адресов в диапазоне от 0 до $X - 1$. Каждый адрес (x, m) указывает на колонку или обратный образ $\{n\}_{x, m}$. Каждая колонка содержит имена классов n , а ее высота равна мощности соответствующего обратного образа. Макроколонка определяется радиусом обобщения R и содержит соседние колонки с адресами из интервала $[x - R, x + r]$.

Параллельная обработка M групп колонок может быть реализована в микросхеме ПЛИС, что приводит к практически мгновенному обучению. Память,

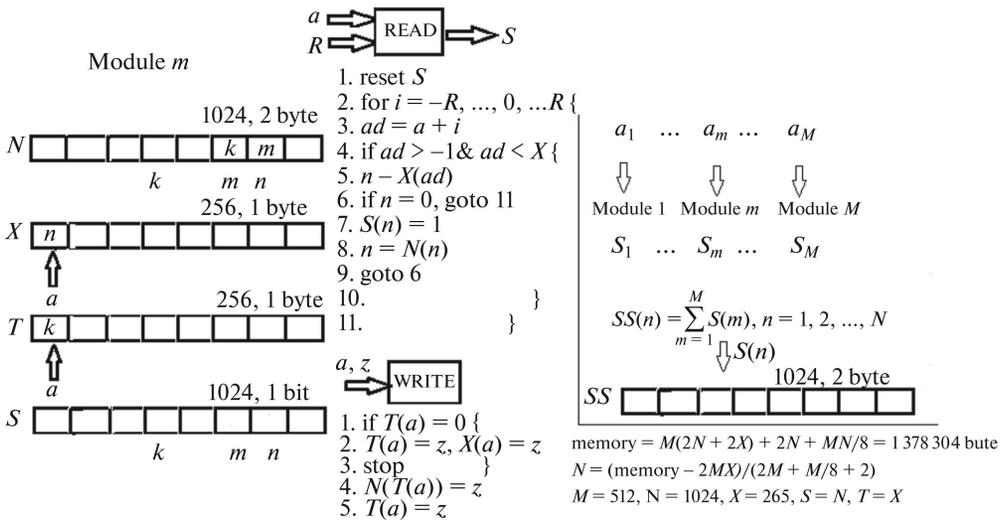


Рис. 2. Концептуальная схема индекстрона.

которая должна быть выделена для групп колонок, может быть организована тремя способами: (а) статические колонки, (б) динамически выделяемые колонки, (в) итерированные одномерные карты [9]. Для этих трех методов потребуется следующее количество ячеек памяти MXN , $2MN$ и MN соответственно. Здесь метод (в) выигрывает и делает возможной ПЛИС-реализацию индекстрона $(M, X, N) = (512, 256, 1024)$ на микросхеме с памятью в 1,44 МБ (рис. 2).

На схеме показан один модуль из M , содержащий четыре массива (N, X, T, S) плюс общий выходной массив SS , в котором суммируется информация всех модулей. Заметим, что количество доступных классов N обратно пропорционально M , т.е. длине входного вектора \vec{x} : $N = (\text{память} - MX - MT)/(2M + M/8 + 2)$.

Нормализованный диапазон дискретных признаков $X = 256$ часто достаточен для многих приложений распознавания образов. При этом как признаки с плавающей запятой, так и целочисленные признаки могут быть преобразованы в 256 дискретных уровней без ущерба для общей точности.

В этой схеме параллельно работают M идентичных модулей, где в m -й модуль поступает m -й компонент a_m вектора признаков $a_1, \dots, a_m, \dots, a_M$. Компонент a_m служит адресом ячейки 256-байтового массива X .

При идентификации вектора выполняется READ-код (см. рис. 2). Выход модуля — это 1024-битное слово S_m . Затем выходы M модулей суммируются в массиве SS , содержащем 1024 ячейки памяти, элементы которой представлены 2-байтовыми словами $SS(n) = \sum_{m=1}^M S(m)$, $n = 1, 2, \dots, N$. Массив SS содержит распределение связей образа $a_1, \dots, a_m, \dots, a_M$.

Если образ не идентифицирован, то после завершения цикла обучения выполняется WRITE-код. Для этого в m -й модуль поступает m -й компонент a_m вектора признаков $a_1, \dots, a_m, \dots, a_M$, а также имя создаваемого клас-

са z . Компонент a_m служит адресом 256-байтового массива T . Например, при $M = 512$, $N = 1024$, $X = 256$, $T = 256$, $S = 1024$ и $SS = 2N$ общая память составит 1 378 304 байта.

2.8. Свойства преобразования образов

Каждая группа колонок, $m = 1, 2, \dots, M$, содержит N различных классов:

$$\sum_{x=0}^{X-1} |\{n\}_{x,m}| = N.$$

Колонки группы не пересекаются: $\{n\}_{x,m} \cap \{n\}_{y,m} = \emptyset$ ($x \neq y$).

Из этих свойств следует, что для вектора \vec{x} максимум распределения связей $F_n^{(R)}(\vec{x}) = M$ достигается тогда и только тогда, когда $H_R(\vec{x}_n, \vec{x}) = 0$.

2.9. Получение разностного изображения

Для получения разностного изображения объекта использовалась функция, которая проводила сравнение каждого пикселя изображения с объектом, со всеми пикселями в определенном радиусе (окне), равном 31×31 пикселям, вокруг соответствующего пикселя изображения без объекта. Результатом сравнения была сумма разницы соответствующих каналов цветов двух пикселей. Если после сравнения всех пикселей в окне наименьшая сумма в этом окне превышала пороговую величину 32, то текущий пиксель изображения с объектом добавлялся в подпись.

3. Результаты

1. Индекстрон был обучен на 2 разных объектах (цветок и кошка). Для тестирования те же объекты были показаны под другими ракурсами и при

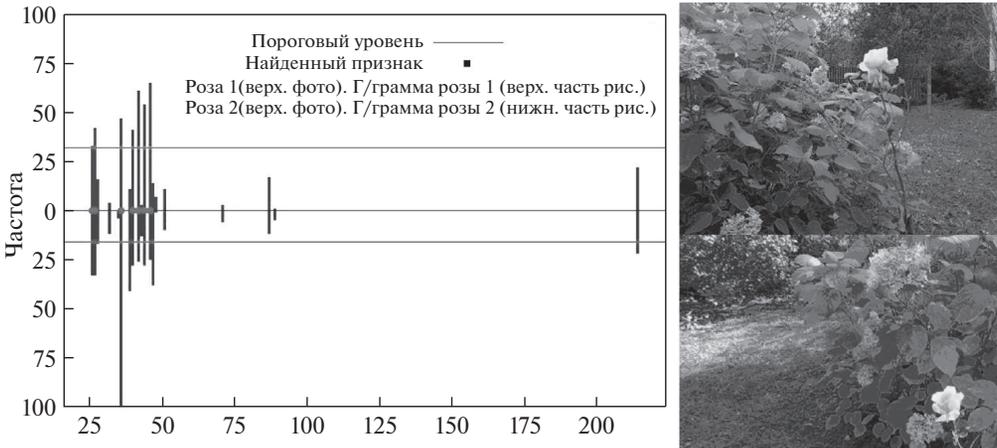


Рис. 3. Сравнение гистограмм подписей изображений объекта цветок с разных сторон при порогах 32 и 16. Количество общих признаков равно семи.

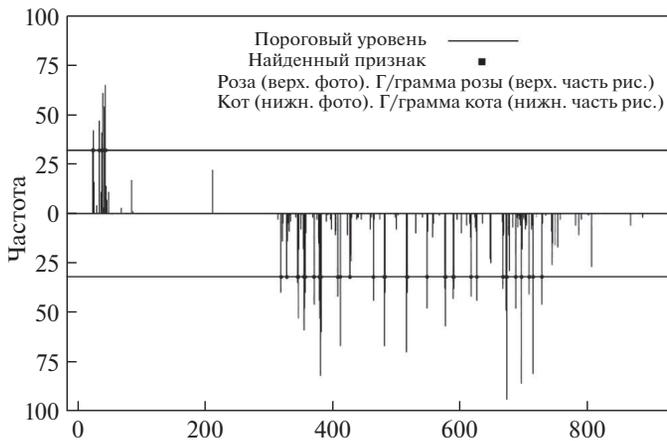


Рис. 4. Сравнение гистограмм подписей изображений объектов цветов и кошки. Общих признаков не найдено.



Рис. 5. Сравнение гистограмм подписей изображений объектов кошку (вид спереди) и кошку (вид сбоку). Количество общих признаков равно шести.

других задних планах. Все объекты были успешно распознаны. (В связи с техническими трудностями при публикации цветные фотографии заменены на черно-белые, которые не показывают полную картину. При этом реальный эксперимент проводился с цветными фотографиями.) На рис. 3 показаны гистограмма классов выделенных пикселей 1-го объекта и гистограмма классов выделенных пикселей того же объекта в другом ракурсе. При частотных порогах 32 и 16 соответственно найдены 7 общих признаков, выделенных жирными точками. Для сравнения на рис. 4 показаны гистограммы 1-го объекта и 2-го объекта, где видно, что число общих признаков равно нулю. Следовательно, объект в нижней части рис. 3 должен быть идентифицирован как объект 1. Аналогично объекты в нижней части рис. 5 и рис. 6 должны быть идентифицированы как объект 2, так как число общих признаков равно ше-

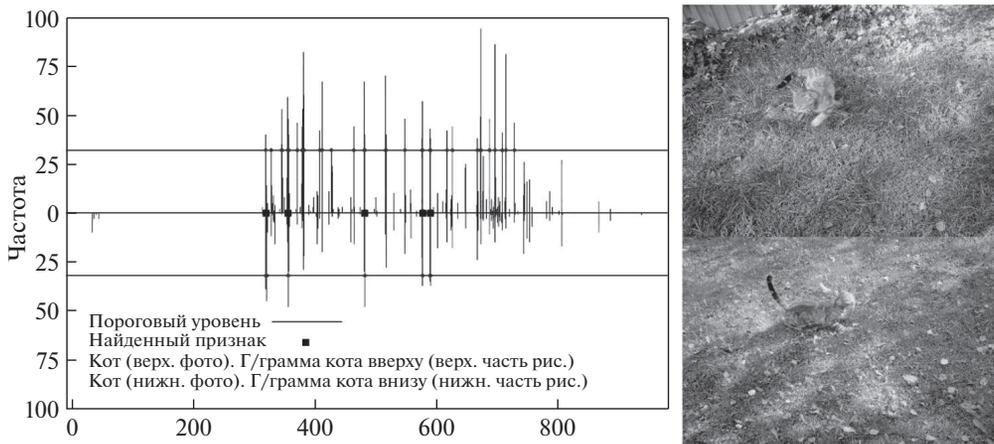


Рис. 6. Сравнение гистограмм подписей изображений объектов кошка (вид спереди) и кошка (другое расположение). Количество общих признаков равно пяти.

сти и пяти соответственно, что больше числа 0, возникающего при сравнении с 1-м объектом. Отметим, что обучение было проведено с использованием только 2 изображений.

2. Набор данных CIFAR-10 [3] использовался для тестирования индекстрона на данных для искусственных нейронных сетей. Для извлечения признаков сверточная нейронная сеть VGG16 с веб-сайта KERAS [10] была предварительно обучена в базе данных ImageNet [11]. Затем VGG16 была обучена на 50 тысячах черно-белых изображений 32×32 (10 категорий) из набора данных CIFAR-10, обеспечивающих 512 признаков на изображение. Наконец, признаки $512 \times 50\,000$ использовались для обучения как 4-слойной нейронной сети, так и индекстрона (таблица, столбец 1). Время обучения указано в столбце 3 таблицы. При тестировании 10 000 изображений ($512 \times 10\,000$ признаков) из вышеуказанного набора данных были классифицированы с использованием как четырехслойной нейронной сети, так и индекстрона. Результаты приведены в столбце 2 таблицы.

Таблица. Нейронная сеть в сравнении с индексным классификатором

Классификатор	Точность %	Обучение (сек)	Оборудование и библиотеки
1	2	3	4
Четырехслойная нейронная сеть Слой: Flatten, Dropout, Dense (256 нейронов, ReLu функция активации), Dense (10 нейронов, softmax)	85,3	900	AMD Ryzen 5 3600, Python, Nvidia GeForce GTX 1660 Super, TensorFlow, Keras, cuDNN
Индекстрон	82,87	16	AMD Ryzen 5 3600, Python

4. Заключение

1. Общей чертой как индекстрона, так и глубокого обучения является автоматическое формирование признаков, что существенно снижает трудоемкость проектирования систем распознавания образов. Вместе с тем производительность индекстрона при обучении намного выше, что связано с используемым в нем методом обратных образов. Этот метод является аналогом индексных поисковиков типа Google, чем в значительной степени и определяется обеспечиваемая методом обратных образов мгновенность обучения каждому новому объекту. Однако если поисковики работают с символьными запросами, то индекстрон — с численными данными.

2. Один из вопросов, стоящих перед машинным обучением, — существуют ли некоторые фундаментальные законы, которым должны следовать оптимальные системы распознавания [12]. Возможно, что ответ на этот вопрос связан с простотой и параллелизмом обратных классификаторов, которые могут работать как индексаторы образов. Перспективным направлением исследований был бы анализ, могут ли описания биологических нейронных сетей быть основаны на обратных множествах. Такой анализ мог бы подсказать эксперименты для проверки гипотетического предположения, что индексация образов, а не вычисления лежат в основе биологических распознающих систем.

3. Время обучения и энергозатраты, требуемые для обучения систем распознавания при решении крупных задач, приводят к необходимости использования облачных технологий и высокопроизводительных машинных станций. Однако простота алгоритма индекстрона позволяет реализовать его всего на одной программируемой логической интегральной микросхеме (ПЛИС) даже в случае таких крупномасштабных задач, как, например, прогнозирование времени отказов авиадвигателей. Аппаратная реализация индекстрона делает возможным его использование для различных автономных устройств, где требуется мгновенная реакция, в том числе мгновенное обучение при возникновении новых ситуаций.

СПИСОК ЛИТЕРАТУРЫ

1. *Mikhailov A., Karavay M.* Pattern Recognition by Pattern Inversion // Proc. 2nd Int. Conf. on Image, Video Processing and Artificial Intelligence, Shanghai, China, 2019 SPIE Digital Library, V. 11321.
<https://spie.org/Publications/Proceedings/Volume/11321>,
<https://doi.org/10.1117/12.2539069>
2. <https://monkeylearn.com/blog/what-is-tf-idf/>
3. *Krizhevsky A., Nair V., Hinton G.* CIFAR-10. 2009.
Retrieved from <https://www.cs.toronto.edu/~kriz/cifar.html>
4. *Theodoridis S., Koutroumbas K.* Pattern Recognition, Academic Press (3rd edition), 2006.
5. *Mikhailov A., Karavay M., Farkhadov M.* Inverse Sets in Big Data Processing // 11th IEEE Int. Conf. on Application of Information and Communication Technology (AICT2017, placeCityMoscow, 2017). M.: IEEE, V. 1.

https://www.researchgate.net/publication/321309177_Inverse_Sets_in_Big_Data_Processing.

6. *Brin S., Page L.* The Anatomy of a Large-scale Hypotextual web Search Engine // Computer Networks and ISDN Syst. 1998. V. 30. Iss. 1–7. Stanford, CA, 94305, USA. Retrieved from [https://doi.org/10.1016/S0169-7552\(98\)00110-X](https://doi.org/10.1016/S0169-7552(98)00110-X)
7. *Sivic J., Zisserman A.* Efficient Visual Search of Videos Cast as Text Retrieval // IEEE Trans. Pattern Analysis and Machine Intelligence. 2009. V. 31. Iss. 4. <https://doi.org/10.1109/TPAMI.2008.111>
8. *Gonzales R., Woods R.* Digital Image Processing. Pearson Prentice Hall (3rd edition). 2008. http://www.imageprocessingplace.com/root_files_V3/image_databases.htm
9. *Dmitriev A., Panas A., Starkov S.* Storing and Recognizing Information Based on Stable cycles of One-dimensional Maps // Phys. Lett. 1991.
10. Keras. Retrieved from <https://keras.io>. 2019.
11. Imagenet. Retrieved from <http://image-net.org>. 2016.
12. *Jordan M., Mitchell T.* Machine learning: Trends, perspectives, and prospects // Science. 2015. V. 349. Iss. 6245.

Статъа представена к публикации членом редколегии А.А. Галяевым.

Поступила в редакцию 05.06.2021

После доработки 22.11.2021

Принята к публикации 24.12.2021