

Розроблено модифікований метод розпізнавання стандартизованого тексту на зображенні. Виділені етапи обробки зображення перед використанням методу: перетворення зображення в чорно-біле, виправлення спотворень символів, детектування окремих символів. Розпізнавання символів здійснено нейронною мережею за допомогою методу шаблонів. Представлені два варіанти рішення поставленої задачі – перший типовий, другий модифікований. В результаті проведених експериментів підтверджується ефективність модифікованого методу

Ключові слова: розпізнавання тексту, метод шаблонів, еталон, нейронна мережа, перцептрон, OpenCV

Разработан модифицированный метод распознавания текста на изображении. Выделены этапы обработки изображения перед применением метода: преобразование изображения, исправление искажений, детектирование символов. Распознавание символов осуществлено нейронной сетью с помощью метода шаблонов. Представлены два варианта решения поставленной задачи – первый реализован по известному алгоритму, второй модифицированный. В результате проведенных экспериментов подтверждается эффективность модифицированного метода

Ключевые слова: распознавание текста, метод шаблонов, эталон, нейронная сеть, перцептрон, номерной знак, OpenCV

РАЗРАБОТКА МОДИФИЦИРОВАННОГО МЕТОДА РАСПОЗНАВАНИЯ ТЕКСТА НА СТАНДАРТИЗИРОВАННОМ ИЗОБРАЖЕНИИ

К. Н. Касьян

Кандидат технічних наук, доцент*

E-mail: konst_k@yahoo.com

В. В. Братчиков*

E-mail: bratchvv@gmail.com

В. В. Шкарупило

Кандидат технічних наук*

E-mail: vadshkar@yandex.ua

*Кафедра комп'ютерних систем та мереж

Запорізький національний технічний університет

вул. Жуковського, 64, м. Запоріжжя, Україна, 69063

1. Введение

Распознавание текста на изображениях – очень актуальная тема для исследований, которая позволяет решать ряд научных и прикладных задач. Современные методы распознавания символов используются для решения широкого круга задач, как офисных, так и специализированных, например, распознавание изображений маркировки оборудования и др.

Существует большое количество программного обеспечения для распознавания текстов, например, ABBYY FineReader, CuneiForm и др. Кроме того, существует много коммерческих и государственных проектов по распознаванию текста для узкоспециализированных задач, например, системы по распознаванию номерных знаков автомобилей.

При создании сложных, высоконагруженных систем по распознаванию текста номерных знаков или другого стандартизованного текста, может возникнуть проблема скорости распознавания данных. Поэтому актуальным представляется разработка улучшенного метода распознавания текста для повышения производительности при обработке больших массивов данных. При этом необходимо проанализировать и вы-

брать оптимальный метод распознавания текста для поставленной задачи.

Преимущества разработанного метода могут оказаться полезными в системах распознавания текстов с большой нагрузкой, которые ориентированы на распознавание коротких стандартизованных текстов, например, номерных знаков автомобиля. Разработанный метод может позволить увеличить производительность в распознавании больших массивов коротких стандартизованных текстов, если они содержат повторяющиеся символы.

2. Анализ литературных данных и постановка проблемы

Проблема распознавания текста на изображениях очень популярна, существует множество научных работ на эту тему. Для построения систем распознавания текста популярны два подхода: метрики [1] и нейронные сети [2, 3]. В ходе исследований было принято решение использования нейронных сетей для разработки системы распознавания номерных знаков. Существует много методов для распознавания текста: метод структурных фреймов [4], метод биоалгоритмов

анализа изображений [5], метод геометрических моментов [6], метод дескрипторов Фурье [6], метод вейвлет-преобразования, метод главных компонент [7], метод шаблонов [8] и др.

В существующих работах по распознаванию номерных знаков автомобилей [9–12] не проводились исследования по улучшению алгоритма распознавания текста, подобно тому методу, который был разработан в данном научном исследовании. Если система по распознаванию номерных знаков автомобилей имеет централизованный характер и охватывает целый город, район города или шоссе с большим потоком автотранспорта, то она создает большую нагрузку на сервер, что приводит к временным задержкам.

Однако, вопросы, которые напрямую связаны с распознаванием текста на стандартизованном изображении, особенно для систем с большой нагрузкой, исследованы не до конца. В частности, неясными остаются аспекты оптимальности и элементы ее доказательной базы.

3. Цель и задачи исследования

Целью проведения данного исследования была разработка алгоритма распознавания текста на стандартизованном изображении на основе уже известных алгоритмов. В качестве стандартизованного изображения был использован номерной знак автомобиля.

Для достижения поставленной цели решались следующие задачи:

- поиск оптимального алгоритма распознавания текста номерного знака автомобиля;
- реализация разработанного алгоритма, его тестирование и сравнение с первоначальным вариантом.

4. Материалы и методы исследований в разработке алгоритма распознавания текста на стандартизованном изображении

4. 1. Исследуемые материалы и оборудования, что использовались в эксперименте

Исследования проводили на макете автомобильного номерного знака с использованием камеры с матрицей в 5 Мп.

Для программной реализации задачи распознавания текста был выбран подход, основанный на нейронных сетях [13]. В этом исследовании была разработана нейронная сеть [14], возможностей которой достаточно для демонстрации разработанного метода.

В первую очередь, необходимо выбрать, какой метод распознавания изображения оптимально использовать. Для этого, в свою очередь, нужно проанализировать методы распознавания и особенности исходных изображений. Шаблоновые системы [15, 16] преобразуют изображение отдельного символа в растр, нормализуя при этом его размеры, толщину штриха и угол наклона; а затем сравнивают его со всеми шаблонами, имеющимися в базе данных системы, и выбирают шаблон с наименьшим количеством точек, отличающихся по расположению в растре от входного изображения. Преимуществом шаблонных систем является малая чувствительность к дефектам

печати, а недостатком – практическая неспособность распознавать символы при их незнакомом начертании. В признаковых системах [17] усредненное изображение каждого символа представляется как объект в пространстве признаков, значения которых вычисляются при распознавании входного изображения. Полученный вектор сравнивается с эталонными и изображение соотносится с наиболее подходящим из них. Признаковые системы лучше адаптируются к начертанию, но более чувствительны к дефектам печати. В структурных системах [18] объект описывается в терминах частей буквы и соотношений между ними. Например, системе известно, что буква “Р” состоит из вертикальной палочки и, расположенной в верхней части этой палочки, справа от нее, фигуры, похожей на полукруг. Система анализирует не первоначальное изображение буквы, а ее контур, который вычисляется перед тем, как буква начинает распознаваться. Недостатком структурных систем является более медленная работа и высокая чувствительность к дефектам изображения, нарушающим составляющие буквы элементы.

Номерной знак автомобиля имеет стандартизованный вид, шрифт, а символы имеют одинаковый размер и цвет. Цвет символов к тому же черный на белом фоне. Исходя из всего этого, был выбран метод шаблонных систем. Кроме того, разрабатываемый алгоритм не имеет жесткой связи с методом шаблонов, поэтому он может быть применен и для структурных методов распознавания текста.

Изображения эталонов для реализации такой системы показаны на рис. 1.

1234567890
АВСІЕНКМОРТХ

Рис. 1. Изображения эталонов символов номерного знака

Словесное описание алгоритма шаблонных систем выглядит следующим образом:

1. На исходном изображении удаляются шумы, а изображение конвертируется в черно-белое.
2. На изображении выделяются регионы, содержащие в себе символы номера, после чего обрезаются в отдельные изображения.
3. Обрезанное изображение с символом конвертируется в новое изображение с размером, соответствующим эталонам изображений в нейронной сети.
4. Из нового изображения создается двумерный массив, где индексы массива соответствуют координатам пикселя изображения, а значение – наличию цвета пикселя.
5. Последним шагом является распознавание символа нейронной сетью, что осуществляется сравнением весов синапсов эталонного изображения и исходного с определенным порогом значений, который вычисляется экспериментально.
6. Определенный символ заносится в результирующий вектор.

Блок-схема этого алгоритма показана на рис. 2.



Рис. 2. Блок-схема исходного алгоритма

Разработанное приложение написано на языке программирования Java 7 [19] с использованием фреймворка JavaCV [20], что является реализацией библиотеки компьютерного зрения OpenCV [21], написанном на C++.

4. 2. Методика улучшения метода распознавания текста на стандартизированном изображении

Учитывая то, что на номерном знаке цифры могут повторяться, алгоритм был модифицирован. Модификация состояла в том, чтобы перед распознаванием текста находить одинаковые символы, запоминать их позиции и распознавать только один из одинаковых символов, и размещать значение на сохраненные места.

Было проведено исследование об актуальности использования такой модификации. Рассчитав вероятность появления хотя бы двух одинаковых цифр на номерном знаке, состоящем из четырех цифр, было получено значение:

$$P = 1 - \frac{10 \cdot 9 \cdot 8 \cdot 7}{10^4} = 0.5.$$

Судя по этим расчётам, можно сказать, что каждое второе транспортное средство будет иметь повторяющиеся цифры в номере. Также экспериментально было установлено, что из 200 случайно выбранных автомобилей, 148 имели две и больше повторяющиеся цифры в номере, то есть 74 % автомобилей. Это объясняется желанием владельцев выбирать себе, при возможности, легко запоминающиеся номерные знаки. Судя из этих исследований, применять данную модификацию имеет смысл.

Словесное описание модифицированного алгоритма имеет следующий вид:

1. Выполняются первые три шага из метода шаблонов.
2. Из нового изображения создается двумерный массив, где индексы массива соответствуют координатам пикселя изображения, а значение – наличие цвета пикселя с определенным устанавливаемым порогом значений RGB. Значения могут быть такими: “1” – цвет есть, “0” – цвет белый.
3. Все полученные массивы добавляются в список.
4. Из массивов в списке формируются векторы для сравнения. Эти векторы представляют собой часть значений пикселей из массива с шагом 5 по горизонтали и шагом 4 по вертикали.
5. Рассчитанные векторы сравниваются между собой, и на результате их сравнения строится выходной вектор сравнений. Например, при значении номера “5566” выходной вектор будет следующий – (0,0,2,2). В тексте “5566” производится поиск одинаковых символов по контрольным точкам, в итоге для распознавания выделяется только два символа “5” и “6”. В качестве индекса символа используется индекс символа во входящем массиве, поэтому для символа “5” индекс равен 0, а для символа “6” индекс равен 2. Эти индексы используются при формировании выходного вектора.
6. Дополнительно к вектору с порядковыми номерами формируется карта, содержащая те символы, которые необходимо распознать в том случае, который описан выше. Это будет один символ “5” и один символ “6”.
7. Происходит распознавание символов в нейронной сети.
8. Распознанные символы расставляются в соответствии с выходным вектором сравнений.
9. Формируется вектор результата.

Блок-схема модифицированного алгоритма показана на рис. 3, 4.

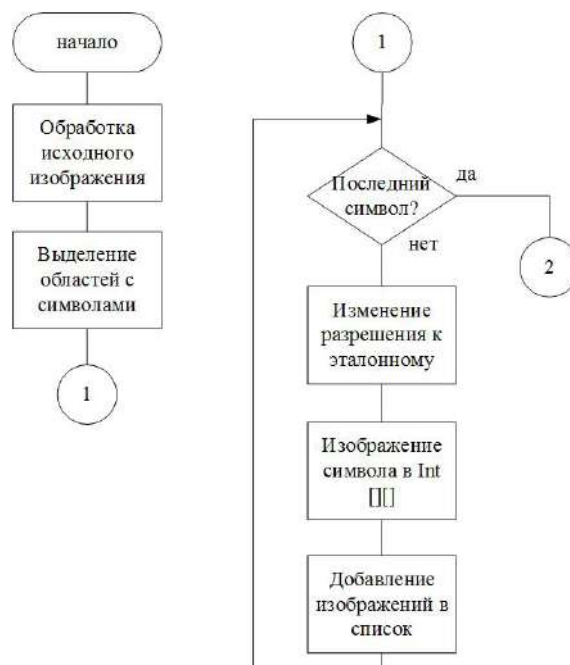


Рис. 3. Блок-схема модифицированного алгоритма, часть 1

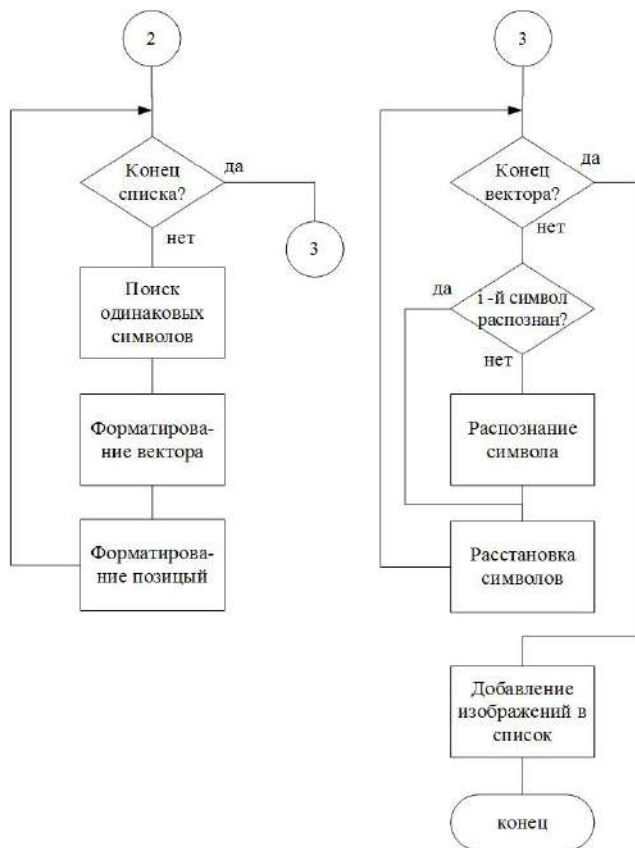


Рис. 4. Блок-схема модифицированного алгоритма, часть 2

Исходя из проведенных исследований, можно сделать вывод, что использование модифицированного алгоритма актуально для стандартизированных изображений. Разработанный алгоритм позволяет ускорить процесс распознавания текста на изображении, если этот текст содержит повторяющиеся символы.

5. Результаты исследований по улучшению метода распознавания текста на стандартизированном изображении

Для создания приложения, реализующего поставленную задачу, было выбрано язык программирования JAVA 7. На вход приложению подается изображение номерного знака автомобиля, а на выходе получается вектор с распознанными символами. Нахождение и обрезание номерного знака из изображения с автомобилем не относится к исследованию, поэтому за исходные изображения используются уже обрезанные знаки. Разработанное приложение основано на нейронной сети, представляющей однослойный перцептрон. Такая нейронная сеть хоть и имеет недостатки из-за своей простоты, но ее достаточно для тестирования исследуемых методов. Разработанный алгоритм не зависит от нейронной сети, на которой построена система, и она может быть заменена на другую, например, многослойный перцептрон или сверточную нейронную сеть. Схема однослойного перцептрона показана на рис. 5.

Для обработки изображения, а именно фильтрации, преобразования в черно-белое, выделения и об-

резание областей, а также изменения разрешения на эталонное [22], было принято решение о использовании библиотеки OpenCV версии 2.4.8, написанную на C++. Чтобы можно было работать с этой библиотекой на языке Java, использовалась ее native реализация – JavaCV.

В качестве примера будет показано преобразование номерного знака, который изображен на рис. 6.

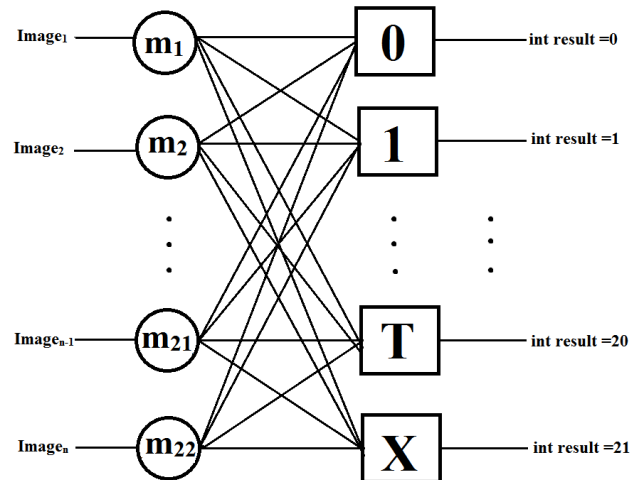


Рис. 5. Схема работы однослойного перцептрона



Рис. 6. Исходное изображение номерного знака

На первом шаге изображение преобразуется из цветного в серое с помощью функции: cvCvtColor (RawImage, GrayImage, CV_BGR2GRAY). Результат преобразованного изображения показан на рис. 7.



Рис. 7. Исходное изображение, преобразованное в серое

После этого из серого изображение трансформируется в черно-белое: cvThreshold (GrayImage, BWImage, 127, 255, CV_THRESH_BINARY).

Результат преобразованного изображения показан на рис. 8.



Рис. 8. Исходное изображение, преобразованное в черно-белое

На третьем шаге преобразования изображения в черно-белое вносятся коррекции размеров искаженных символов для большего придания сходства этих символов с эталонными:

- cvErode (BWImage, WorkingImage, null, Erode Count);
- cvDilate (WorkingImage, WorkingImage, null, Dilate Count).

Результат преобразованного изображения показан на рис. 9.



Рис. 9. Откорректированное черно-белое изображение

После создания рабочего изображения, на нем ищутся области с символами с помощью функции [23] библиотеки OpenCV – BlobAnalysis(), которая выделяет регионы на изображении по определенным границам, в нашем случае черным границам каждого символа. Изображение с выделенными границами показано на рис. 10.



Рис. 10. Исходное изображение с выделенными границами символов

Из этих выделенных областей создаются отдельные изображения, которые преобразуются в изображения с разрешением эталонных с помощью функции cvResize (). Для разработанного метода были взяты эталонные изображения с разрешением 30x50 пикселей.

Для проверки скорости выполнения модифицированного и первоначального методов были написаны JUnit тесты и сравнено время их работы. Из-за того, что символы на буквенной кодификации номера почти не повторяются, было принято решение использовать модифицированный алгоритм только для числового значения номера. Были написаны тесты для случаев, когда все цифры номерного знака разные, когда все цифры одинаковые, а также – тесты, где присутствуют все возможные ситуации. Результаты теста 100 и 1000 номерных знаков для каждого случая показаны в табл. 1. Результаты тестов распознавания номерных знаков при использовании других методов показаны в табл. 2.

Так как JavaCV использует native код, при обработке изображений возникают временные задержки на подключение OpenCV. Для точного и рационального сравнения скорости работы модифицированного и типичного методов были созданы таймеры, которые засекают время выполнения распознавания изображения после обработки их средствами OpenCV. Были получены следующие результаты:

1. Случай, когда в номере все цифры одинаковые, среднее время выполнения не модифицированного метода равно 65 мс. Для модифицированного метода удалось улучшить среднее время выполнения в 2.1 раза; результат составляет 30 мс.

2. Случай, когда в номере все цифры разные, время выполнения не модифицированного метода в среднем равно 60 мс. Для модифицированного метода время выполнения составляет 65 мс.

Таблица 1

Результаты тестирования, стандартного и модифицированного методов распознавания текста на номерных знаках при разных комбинациях символов

Метод	Входные параметры	Кол-во, шт.	Время, с
Метод шаблонов	Есть одинаковые символы	100	3,5
Улучшенный метод	Есть одинаковые символы	100	2,1
Метод шаблонов	Есть одинаковые символы	1000	33,2
Улучшенный метод	Есть одинаковые символы	1000	20,1
Метод шаблонов	Нет одинаковых символов	100	3,86
Улучшенный метод	Нет одинаковых символов	100	3,99
Метод шаблонов	Нет одинаковых символов	1000	38,2
Улучшенный метод	Нет одинаковых символов	1000	39,4
Метод шаблонов	Случайные символы	100	3,7
Улучшенный метод	Случайные символы	100	2,4
Метод шаблонов	Случайные символы	1000	38,4
Улучшенный метод	Случайные символы	1000	23,7

Таблица 2

Результаты тестирования разных методов распознавания текста на номерных знаках

Метод	Кол-во, шт.	Время, с
Метод шаблонов	100	3,7
Метод структурных фреймов	100	2,9
Метод геометрических моментов	100	3,1
Метод главных компонент	100	3,3
Разработанный метод	100	2,4

Также было подсчитано количество выполненных операций для различных случаев. Количество операций для типичного метода вычисляется по следующей формуле:

$$N_{no_mod} = 2 \cdot h + n \cdot m + w \cdot h \cdot n = 2 \cdot (2 \cdot w \cdot h + m), \quad (1)$$

где n – количество распознаваемых символов, w – ширина эталонного символа, h – высота эталонного символа, m – количество доступных к распознаванию символов.

Для не модифицированного метода количество операций для распознавания четырёхзначного номера с разрешением эталонного символа 30x50 пикселей равно:

$$N_{\text{no_mod}} = 4 \cdot (2 \cdot 30 \cdot 50 + 10) = 12040.$$

Количество операций для модифицированного метода вычисляется по следующей формуле:

$$N_{\text{mod}} = \frac{2 \cdot n \cdot m}{s^2} + 2 \cdot (k \cdot w \cdot h + m), \quad (2)$$

где n – количество символов без повторов, w – ширина эталонного символа, h – высота эталонного символа, m – количество доступных к распознаванию символов, k – шаг для выборки вектора сравнений.

Для модифицированного метода количество операций отличается в зависимости от ситуаций:

Случай, когда в номере все цифры одинаковые:

$$N_{\text{mod}} = \frac{2 \cdot 30 \cdot 50}{5^2} + 2 \cdot (1 \cdot 30 \cdot 50) = 3490.$$

Случай, когда одинаковых цифр три или по две одинаковые пары цифр:

$$N_{\text{mod}} = \frac{2 \cdot 30 \cdot 50}{5^2} + 2 \cdot (2 \cdot 30 \cdot 50 + 10) = 6500.$$

Случай, когда две одинаковые цифры и две разные:

$$N_{\text{mod}} = \frac{2 \cdot 30 \cdot 50}{5^2} + 2 \cdot (3 \cdot 30 \cdot 50 + 10) = 9510.$$

Случай, когда все цифры разные:

$$N_{\text{mod}} = \frac{2 \cdot 30 \cdot 50}{5^2} + 2 \cdot (4 \cdot 30 \cdot 50 + 10) = 12520.$$

6. Обсуждение результатов работы модифицированного метода распознавания текста на стандартизированном изображении

Разработанный метод на основе метода шаблонов вполне удовлетворяет решение поставленной задачи. Такой метод может быть использован для распознавания текста на изображении, если этот текст имеет стандартизированную форму. Например, номерной знак автомобиля, различные маркировочные номера, серийные номера каких-либо объектов и т. д.

Был проведен анализ скоростей работы обеих алгоритмов на примере распознавания текста номерного знака автомобиля. Разработанный модифицированный метод имеет преимущества над стандартным методом шаблонов, если на картинке есть одинаковые символы. В лучшем случае, для распознавания четырех цифр он работает в 2 раза быстрее по времени и почти в 3.5 раза меньше по количеству итераций. В худшем случае, он лишь на 4 % менее эффективен, чем исходный метод. Использование модифицированного метода рационально при последовательности из четырех и больше цифр, так как вероятность того, что из четырех хотя бы две цифры будут одинаковы, равна 0.5. Для меньшего количества цифр вероятность будет меньше, что не

позволяет использовать модифицированный метод оптимально.

При увеличении количества символов на изображении, модифицированный алгоритм также будет быстрее и будет иметь меньшее количество итераций. Например, при распознавании десяти цифр, в наихудшем случае, он будет иметь на 1120 итераций больше чем в стандартном алгоритме. Вероятность того, что все цифры из десяти будут разные равна 0,003 %, поэтому этим недостатком можно пренебречь. А если в такой последовательности будет хотя бы две одинаковые цифры, вероятность чего равна 0.99, то модифицированный метод уже будет опережать исходный на 1880 итераций.

Разработанная модификация не зависит от количества символов, способных различаться нейронной сетью, но оптимальность ее использования зависит от вероятности повторов символов в последовательности, например, вероятность повторов хотя бы двух букв украинского алфавита в последовательности из четырех равна 0.18. Поэтому лучше всего использовать эту модификацию исключительно для распознавания чисел.

Разработанная система распознавания номерных знаков автомобиля предназначена только для работы на тестовом стенде, потому что в ней до конца не проработаны варианты с искаженными изображениями [24], например, грязным номерным знаком, расфокусированным снимком и т. д. Разработанный метод предназначен для внедрения в уже рабочую систему.

7. Выводы

В данном исследовании был проведен поиск оптимального алгоритма распознавания текста номерного знака для дальнейшего его улучшения. В ходе исследований был определен подходящий метод распознавания текста – метод шаблонов. На основе этого алгоритма был разработан модифицированный алгоритм.

Проведенными исследованиями и разработанным алгоритмом было доказано, что при распознавании текста на стандартизированных изображениях, на примере номерного знака автомобиля, осуществляется прирост производительности, если перед непосредственным распознаванием символов находить одинаковые элементы, запоминать их количество и расположение. В этом случае необходимо лишь один раз распознать символ и продублировать его во всех отмеченных позициях. Недостатком разработанного метода является избыточность вычислений при отсутствии совпадающих элементов на изображении, но это приводит лишь к незначительной потере в производительности.

В лучшем случае, производительность модифицированного метода составляет 300 % от производительности исходного. В худшем случае он работает медленнее всего на 5–10 %. Кроме того, модифицированный алгоритм требует в 3 раза меньше итераций. Таким образом, модифицированный алгоритм распознавания текста на стандартизированном изображении позволяет получить прирост в скорости решения этой задачи.

Литература

1. Сюезев, В. В. Гибридный метод оптического распознавания текста с коррекцией результатов распознавания [Текст] / В. В. Сюезев, А. А. Ханин // Инженерный журнал: наука и инновации. – 2012. – № 11. – С. 12.
2. Нгуен, Тьен Тхи Кхань. Обнаружение и распознавание текстов на изображениях сложных графических сцен с помощью свёрточных нейронных сетей [Текст] / Нгуен Тьен Тхи Кхань // Электротехнические и компьютерные системы. – 2014. – № 13. – С. 125–130.
3. Мокшин, В. В. Разработка системы распознавания символов с использованием нейронной сети [Текст] / В. В. Мокшин, Л. Р. Габдрахманова // Современные инновации в науке и технике. – 2014. – № 4. – С. 223–225.
4. Зеленцов, И. А. Распознавание образов на основе структурных фреймовых описаний в скорописных текстах XVII века [Текст] / И. А. Зеленцов, Ю. Н. Филипович // Наука и образование: электронное научно-техническое издание. – 2011. – № 12. – С. 28.
5. Кучуганов, А. В. Распознавание старославянских текстов методами, основанными на биоалгоритмах анализа изображений [Текст] / А. В. Кучуганов, П. П. Осколков. – Современные информационные технологии и письменное наследие: от древних текстов к электронным библиотекам. e^lmanuscript-08, 2008. – С. 168–172.
6. Кубрин, С. С. Сравнение метода геометрических моментов и метода дескрипторов фурье в задачах распознавания текста [Текст] / С. С. Кубрин, Н. А. Федченко, А. Б. Исаев, С. А. Мастеров // Горный информационно-аналитический бюллетень (научно-технический журнал). – 2005. – № 3. – С. 106–108.
7. Фан, Нгок Хоанг. Распознавание печатных текстов на основе применения вейвлет-преобразования и метода главных компонент [Текст] / Фан Нгок Хоанг, Буй Тхи Тху Чанг, В. Г. Спицын // Известия томского политехнического университета. – 2012. – № 5. – С. 154–157.
8. Yan, J. Detection and recognition of text superimposed in images base on layered method [Text] / J. Yan, X. Gao // Neurocomputing. – 2012. – Vol. 134. – P. 3–14. doi: 10.1016/j.neucom.2012.12.070
9. González, A. A text reading algorithm for natural images [Text] // Á. González, L. M. Bergasa // Image and Vision Computing. – 2013. – Vol. 31, Issue 3. – P. 255–274. doi: 10.1016/j.imavis.2013.01.003
10. Убоженко, Н. В. Анализ эффективности методов распознавания символов в рамках задачи распознавания номерного знака автотранспорта [Текст] / Н. В. Убоженко // Перспективы развития информационных технологий. – 2013. – № 12. – С. 41–45.
11. Качановский, Ю. П. Разработка алгоритма локализации номерного знака для применения в распределенном аппаратно-программном комплексе распознавания автомобильных номеров [Текст] / Ю. П. Качановский, А. Г. Явтухович // Информационные технологии моделирования и управления. – 2007. – № 39. – С. 508–516.
12. Петров, С. П. Сверточная нейронная сеть для распознавания символов номерного знака автомобиля [Текст] / С. П. Петров // Системный анализ в науке и образовании. – 2013. – № 21. – С. 66–73.
13. Горбань, А. Н. Нейроинформатика [Текст] / А.Н. Горбань, В.Л. Дунин-Барковский, А. Н. Кардин; под ред. Е. А. Новиков. – РАН, Сиб. Отд., Институт выч. Моделирования. – Новосибирск: Наука. Сибирское предприятие РАН, 1998. – 296 с.
14. Уоссерман, Ф. Нейрокомпьютерная техника: Теория и практика [Текст] / Ф. Уоссерман. – М.: Мир, 1992. – 184 с.
15. Федотов, Н. Г. Методы стохастической геометрии в распознавании образов [Текст] / Н. Г. Федотов. – М.: Радио и связь, 1990. – 144 с.
16. Yaser, S. Learning From Data [Text] / S Yaser. – AMLBook, 2012. – 213 p.
17. Flanagan, C. OCR Psychology: AS Revision Guide [Text] / C. Flanagan. – Psychology Press, 2013. – 88 p. doi: 10.4324/9780203796665
18. Parker, J. R. Algorithms for Image Processing and Computer Vision [Text] / J. R. Parker. – Wiley, 2010. – 504 p.
19. Bloch, J. Effective Java. 2nd Edition [Text] / J. Bloch. – Santa Clara, California: Sun Microsystems Inc., 2008. – 369 p.
20. Хоминченко, Д. Настройка JavaCV в windows 7 [Электронный ресурс] / Д. Хоминченко. – Брест, 2013. – Режим доступа: <http://habrahabr.ru/post/190104>
21. OpenCV – Documentation [Electronic resource] / Available from: <http://docs.opencv.org/>
22. Bradski, G. R. Learning OpenCV [Text] / G. R. Bradski, A. Kaehler. – Sebastopol, California: O'Reilly Media, Inc., 2008. – 571 p.
23. Forsyth, D. A. Computer Vision: A Modern Approach [Text] / D. A. Forsyth, J. Ponce. – Pearson, 2011. – 792 p.
24. Убоженко, Н. В. Анализ методов предварительной обработки изображения в рамках задачи распознавания грязных и/или зашумленных номерных знаков автотранспорта [Текст] / Н. В. Убоженко // Перспективы развития информационных технологий. – 2014. – № 18. – С. 57–61.