

Локализация и распознавание автомобильных номеров на изображениях

Рассмотрены новые алгоритмы поиска и распознавания автомобильных номеров на изображениях. Распознавание номерного знака выполняется путем выделения особых точек на контурах символов и вычисления модифицированного расстояния Левенштейна между входными и эталонными цепочками этих точек. Приведены результаты экспериментальной проверки предложенных алгоритмов.

Ключевые слова: изображения, локализация и распознавание автомобильных номеров, расстояние Левенштейна, преобразование Хафа.

Розглянуто нові алгоритми пошуку і розпізнавання автомобільних номерів на зображеннях. Розпізнавання номерного знаку виконується шляхом виділення особливих точок на контурах символів і обчислення модифікованої відстані Левенштейна між вхідними та еталонними ланцюжками цих точок. Подано результати експериментальної перевірки запропонованих алгоритмів.

Ключові слова: зображення, локалізація і розпізнавання автомобільних номерів, відстань Левенштейна, перетворення Хафа.

Введение. Распознавание изображений автомобильных номеров – частный случай поиска и распознавания текстовой информации в окружающем реальном мире. Эти результаты используются в автоматизированных системах контроля доступа автомобилей, взимания оплаты за проезд и парковку, мониторинга перемещения и поиска угнанных автомобилей. Несмотря на наличие ряда систем распознавания автомобильных номеров, задача разработки более эффективных алгоритмов актуальна по следующим основным причинам:

- надежность распознавания известных систем составляет порядка 95 процентов при условии выполнения ограничений на дальность и ориентацию камеры относительно автомобиля, освещение, скорость движения и уровень помех, что не всегда соответствует практическим потребностям и уступает распознающим возможностям человека;

- предпочтительна реализация распознающих модулей на автономных устройствах, что накладывает дополнительные ограничения на быстродействие используемых алгоритмов;

- наличие номеров со сложным фоном, которые помимо текстовой информации могут содержать различного вида цветные рисунки, особенно так называемые *личные* номера.

Компьютерная технология распознавания изображений автомобильных номеров обычно состоит из следующих основных частей [1]:

- быстрый поиск на изображении участков, которые могут включать номерной знак (номер) автомобиля; дополнительная верификация найденных участков и более точная локализация номеров в их пределах;

- нормализация изображения номера по яркости, поворотам и размерам;

- сегментация изображения знака на составляющие части и последующее распознавание каждой из этих частей.

Большинство известных алгоритмов локализации номера на изображении выделяют на первом этапе участки изображения с большим количеством контурных точек, значение градиента яркости в которых превышает пороговое значение [2–5, 13]. В [2] контурные точки выделяют с помощью *LoG* фильтра 9×9 (лапласиан гауссовского фильтра). После этого выполняется морфологическая операция расширения множества контурных точек и, как следствие, объединение этих точек во многих случаях в связное множество. В [4] выполняется выделение вертикальных границ символов с помощью морфологических операций, слияние граничных точек путем расширения в связные области и последующее нахождение номеров в

виде последовательностей этих областей. В [3, 5] положение номера на изображении определяется на основе вычисления интегральных проекций контурных точек по горизонтали и вертикали, но этот способ применим только при достаточно больших размерах и малых углах поворота номера на изображении.

В [6] поиск номеров основан на предварительном выделении максимально стабильных экстремальных областей (*Maximally Stable Extremal Regions – MSER*) на изображении и последующей группировки этих областей в строку. Каждая из экстремальных областей представляет собой связную бинаризованную компоненту изображения и более устойчива к изменениям порога бинаризации в сравнении с другими областями. Каждому символу на изображении в общем случае могут соответствовать не одна, а несколько *MSER*. В [6] показано, что выделение номеров на основе *MSER* устойчиво в сравнении с [4] к яркостным изменениям, а также к изменению ориентации камеры относительно автомобиля.

Результат первого этапа – некоторое множество участков изображения, которые дополнительно верифицируются для уменьшения количества ошибочно выбранных фрагментов окружающего фона вместо номерных знаков.

На втором этапе выполняется нормализация выделенного изображения знака по яркости и геометрическим параметрам (медианная фильтрация, растяжение интервала яркости изображения [2], оценивание и компенсация угла поворота [3]). На последнем этапе выполняется бинаризация изображения с использованием адаптивного порога, выделение связных областей на бинаризованном изображении, вычисление признаков и распознавание выделенных областей с помощью структурного анализа [2], нейронных сетей [2–5] или других методов *OCR* [6].

Постановка задачи

Существенный недостаток известных методов заключается в том, что при отсутствии строгих ограничений (тип, размер и положение номеров) на втором и третьем этапах поступает большое число изображений фона в виде надписей, фрагментов автомобиля и окружающих

объектов. Остальные являются изображениями номеров только условно, так как часто захватывают фрагменты фона или представляют только часть номера автомобиля. Также не всегда обеспечивается правильная сегментация символов номера, что приводит к дополнительным ошибкам распознавания. Слияние этих символов друг с другом и с рамкой номера случается при наличии помех, нефронтальном расположении камеры по отношению к номеру, а также малых размерах номера на изображении. Кроме того, существуют номера, в которых символы касаются рамки или других фрагментов даже на эталонных изображениях, т.е. вследствие их дизайна. При таких условиях сложно или даже невозможно подобрать локальные пороги, обеспечивающие правильную сегментацию по яркости этих символов, и следовательно, свести задачу распознавания всего изображения номера к более простой задаче распознавания каждого из сегментированных символов.

Отмеченные причины приводят к уменьшению скорости обработки и снижению надежности результатов распознавания. С целью повышения точности локализации и распознавания номеров автомобиля в статье рассматривается алгоритм локализации номерного знака на основе выделения контурных линий его рамки с помощью метода Хафа [7, 8]. Рассматриваются также новые алгоритмы распознавания символов на основе выделения и использования их особых точек, предварительной обработки и сегментации изображения номера на отдельные фрагменты.

Локализация номерного знака на изображении

Процедура локализации состоит из трех этапов, первые два из которых не обязательны. На первом этапе выполняется предварительная обработка входных изображений, содержащих малоконтрастные участки, следующим алгоритмом. Входное изображение сканируется окном с размерами w, h и значениями s_x, s_y шага сканирования соответственно по горизонтали и вертикали. При размерах изображения 640×480 значения w, h, s_x, s_y могут быть соответственно равны 80, 60, 20 и 15. Используются два массива I_{sum} и

I_n с нулевыми исходными значениями и размерами изображения. При каждом положении окна участок изображения в его пределах преобразуется таким образом, чтобы средняя яркость и дисперсия стали равными заданным значениям, например, 127 и 32. Полученные значения яркости прибавляются к значениям соответствующих ячеек I_{sum} , а значения I_n в этих ячейках увеличиваются на единицу. По окончании сканирования значения в I_{sum} после деления на значения I_n представляют новые более контрастные яркости входного изображения.

Основное назначение второго этапа – уменьшение времени поиска номеров на изображениях. С этой целью выделяются участки изображения, имеющие сравнительно большое количество вертикальных краев, что характерно для символов, и поэтому могут содержать номера автомобилей. Для определенности будем предполагать поиск номеров с черными символами на белом фоне. Будем считать, что некоторой клетке $tl(x, y)$ изображения соответствует левый вертикальный край линии, если

- свертки $f_a(t), f_b(t)$ оператора Собеля (рис. 1) в этой точке таковы, что $f_a(t)$ больше некоторого порога $\delta > 0$, $f_a(t) > |f_b(t)|$;

- существует клетка $tr(xr, y)$, $xr > xl$, в которой $f_a(tr) < -\delta$, $|f_a(tr)| > f_b(tr)$ и разность $(xr - xl)$ находится в интервале допустимых значений ширины линий символов.

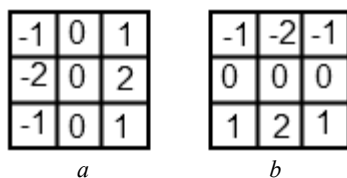


Рис. 1. Операторы Собеля для вычисления градиентов яркости $f_a(t), f_b(t)$: а – по горизонтали; б – по вертикали

Выделение контурных точек выполняется в процессе однократного просмотра строк изображения. При этом если в клетке $tl(xl, y)$ выделяется левый край, то выполняется переход в точку $t(xr + 1, y)$. В процессе просмотра формируется также так называемое *интегральное* изображение (впервые предложенное Шлезинге-

ром М.И. в [9]) – в каждой клетке интегрального изображения хранится количество выделенных контурных точек, расположенных слева и выше этой клетки. Интегральное контурное изображение позволяет с помощью нескольких операций вычислить количество контурных точек в любом прямоугольнике на изображении.

Поиск прямоугольных участков изображения, которые могут содержать номерной знак автомобиля, выполняется следующим образом. С помощью интегрального представления контурного изображения находим положения прямоугольника с минимально допустимыми размерами номерного знака, в пределах каждого из которых количество краевых точек больше заданного порога. Каждый из этих прямоугольников расширяется вдоль его сторон, пока добавляемые части содержат достаточно большое количество краевых точек.

На третьем этапе выполняется поиск номерных знаков на исходном изображении или на изображениях, полученных на втором этапе. Поиск номерного знака выполняется на основе выделения горизонтальных контурных линий его рамки с помощью рандомизированного метода Хафа [8]. Предварительно на изображении выделяются контурные точки, в которых значение градиента яркости по модулю превышает некоторый порог. При поиске прямых линий контура используется нормальная форма представления линий в виде $r = x \cos \theta + y \sin \theta$, где r – расстояние от начала координат до ближайшей точки на линии, а θ – угол между осью x и линией, соединяющей начало координат с этой точкой. Для каждой контурной точки t на изображении (в [8] точки выбираются случайным образом) вычисляются параметры (r, θ) линий, проходящих через точку t , после чего значения соответствующих дискретизированных ячеек накопительной матрицы увеличиваются на единицу. После этого из матрицы последовательно выбираются ячейки с наибольшими значениями и прослеживаются на изображении отрезки линий с соответствующими параметрами и допустимыми длинами разрывов.

Сначала выделяются отрезки квази-горизонтальных линий верхней стороны контура рамки знака (положительные значения градиента яркости по вертикали), а затем нижней. Вертикальные линии рамки выделяются путем прослеживания с использованием данных о горизонтальных проекциях вертикальных краев на изображении. Для уменьшения вероятности выделения линий на символах знака предварительно выполняется операция эрозии изображения в горизонтальном направлении (сжатие областей клеток с малой яркостью). Результатом процедуры уточнения положения номерного знака является некоторое количество четырехугольников, упорядоченных по значениям оценки P правдоподобия соответствия контуру рамки знака. Значение оценки зависит от операций редактирования выделенных отрезков линий при формировании четырехугольника, разностей углов наклона противоположных сторон, а также отношения длин горизонтальной и вертикальной сторон.

Предварительная обработка и распознавание изображения номера

На последующих этапах выполняется обработка и распознавание участков исходного изображения, содержащих выделенные четырехугольники с наибольшими значениями P . Вначале выполняется поворот и растяжение изображения для преобразования четырехугольника в прямоугольник с заданными размерами. После этого выполняется нормализация изображения по яркости, удаление рамки знака, бинаризация изображения с использованием локального порога, а также сегментация символов. Нормализация изображения по яркости (растяжение диапазона яркостей) выполняется по формуле

$$I_N(x, y) = \max(0, I(x, y) - brmi) \frac{255}{(brma - brmi)},$$

где $brmi$, $brma$ – значения яркости, соответствующие пяти и 80 процентам всех клеток на кумулятивной гистограмме яркости клеток исходного изображения I . При этом большая часть клеток символов и фона знака на исходном изображении имеют яркости, близкие к значениям $brmi$ и $brma$, а на нормализованном изображении – ноль и 255.

При бинаризации нормализованного изображения яркость $I_N(x, y)$ каждой клетки изображения сравнивается с двумя пороговыми значениями $thrlow=100$ и $thrhigh=150$: $I_N(x, y)=0$, если $I_N(x, y) < thrlow$, и $I_N(x, y)=255$, если $I_N(x, y) > thrhigh$. Если яркость клетки находится в промежутке между двумя этими значениями, то решение принимается на основе сравнения с локальным порогом. Значение порога вычисляется по локальной окрестности, размер которой примерно равен половине размера символа. Исследовались два способа вычисления локального порога: в виде средней яркости клеток окрестности и с помощью алгоритма Отсу [10] на этой окрестности. Второй способ более эффективен, но требует большего количества операций.

Сегментация строки символов знака на отдельные области может быть выполнена путем выделения связных объектов на бинаризованном изображении. После сегментации изображения на области, содержащие отдельные символы или группы символов, выполняется предварительная обработка и бинаризация исходного изображения уже в пределах каждой из этих областей описанным выше алгоритмом. В некоторых случаях сегментация этих областей на отдельные символы по горизонтали может быть выполнена на основе вычисления интегральных проекций изображения по вертикали. Пример сегментации и бинаризации отдельных символов на изображении представлен на рис. 2.



Рис. 2. a – исходное изображение; b – нормализация по яркости исходного изображения; c – бинаризация; d – сегментированные символы после удаления шума и повторной обработки участков изображения в пределах двух групп неразделенных символов (0391 и CH) на бинаризованном изображении

Известные алгоритмы распознавания символов можно условно разбить на три группы. К первой группе относятся алгоритмы, для использования которых необходимо устранить преобразования изображения, обусловленные нефрон-

тальным расположением камеры по отношению к номерному знаку автомобиля, что не всегда возможно. Ко второй группе принадлежат алгоритмы, у которых устойчивость к искажениям достигается путем обучения на больших выборках, содержащих фронтальные и повернутые изображения. И наконец, к третьей группе относятся алгоритмы, использующие признаки, устойчивые к указанным ранее преобразованиям. Ниже описан один из таких алгоритмов, основанный на выделении так называемых *особых* точек на изображении символа и использующий структурные и метрические признаки сравниваемых контуров.

Основные особые точки находятся на кусочно-линейно аппроксимированном внешнем контуре символа и соответствуют концевым точкам линий символа, а также точкам изменения направления («по часовой стрелке» на «против часовой» или наоборот) обхода контура (рис. 3). Дополнительные особые точки – геометрические центры внутренних контуров на изображении. Контур (граница) объекта на бинарном изображении представляет собой цикл из горизонтальных, вертикальных и диагональных отрезков линий, общие точки которых называются вершинами контура. При кусочно-линейной аппроксимации выполняется замена некоторых цепочек отрезков исходного контура на один отрезок, если расстояния удаленных промежуточных вершин контура до этого отрезка не превышают заданного значения, называемого точностью аппроксимации. Любой контур представляет собой замкнутую последовательность отрезков линий. Пусть $\{v_i, i = 0, \dots, n-1\}$ – упорядоченное множество вершин контурного полигона, такое что за $(n-1)$ -й следует нулевая вершина. В некоторой i -й вершине меняется направление обхода контура, если выполняется следующее условие: $(i+1)$ и $(i-2)$ вершины находятся по разные стороны от линии, проходящей через i и $(i-1)$ вершины контура. Особые точки, соответствующие концевым точкам линий символа, выделяются на основе локального анализа длин и направлений отрезков аппроксимированного контура.

Каждой i -й основной особой точке контура соответствуют следующие основные признаки:

- нормированные координаты $x_n = x*100/width$, $y_n = y*100/height$, где $width, height$ – ширина и высота минимального прямоугольника, ограничивающего контур;
- углы направлений двух отрезков контура с общей точкой i ;
- направление отрезка линии между i и следующей особой точкой при обходе контура;
- тип точки – конец линии или точка изменения направления обхода контура;
- вес $w(i)$ (от нуля до единицы); зависит от информативности точки и значений ее признаков ($w(i)$ увеличивается при выполнении более строгих условий выделения особой точки).



Рис. 3. *a* – бинаризованное изображение символа; *b* – четыре особые точки (обозначены крестиками) на этом изображении: 3 основные особые точки на аппроксимированном контуре изображения (концы линий) и одна дополнительная особая точка (центр внутреннего контура на изображении буквы *A*)

Параметрами дополнительной особой точки есть нормированные координаты этой точки и размеры минимального прямоугольника, ограничивающего соответствующий внутренний контур.

Расстояние между двумя контурами (модифицированное расстояние Левенштейна [11]) определяется как минимальная суммарная стоимость следующих операций редактирования особых точек первого контура при установлении их соответствия особым точкам второго контура, разделенная на сумму весов этих точек:

- замещение или установление соответствия двух точек; стоимость операции вычисляется

как взвешенная сумма разностей значений признаков этих точек, умноженная на сумму весов этих точек;

- удаление точки; стоимость операции равна произведению веса точки на предварительно заданный коэффициент;

- добавление точки; стоимость операции равна произведению веса точки на заданный коэффициент.

Рассмотрим два алгоритма вычисления расстояния между контурами на основе нахождения соответствия между особыми точками. Первый алгоритм использует представление контура в виде последовательности (цепочки) особых точек и вычисляет расстояние между двумя контурами как расстояние Левенштейна между цепочками особых точек этих контуров. Для этого необходимо установить соответствие одной из особых точек первого контура некоторой особой точке на втором контуре, поместить эти точки в начало цепочек, а затем добавить все остальные особые точки в процессе обхода контуров. После создания цепочек необходимо применить алгоритм вычисления расстояния Левенштейна между этими цепочками с изменениями, связанными с введением весов точек.

Известно большое количество способов нахождения соответствия точек на изображениях с помощью оптимизационных алгоритмов. Учитывая особенности символов (малое количество особых точек, небольшие изменения ориентации), поиск соответствия точек может быть также выполнен более простым способом, принадлежащим к классу так называемых *жадных* алгоритмов. Пусть два сравниваемых контура имеют соответственно n_1 и n_2 особых точек, n_{\min}, n_{\max} – меньшее и большее из этих двух чисел, w_{del} – стоимость операции удаления точки контура, w_d – сумма весов удаленных точек контура и w – сумма весов всех особых точек двух контуров. Алгоритм на каждой $i = 1, 2, \dots, n_{\min}$ итерации выполняет поиск пары неразмеченных точек $(p_{1_{j_1(i)}}, p_{2_{j_2(i)}})$, расстояние d_i между которыми наименьшее в сравнении со всеми другими парами неразмеченных точек. Эти две точки считаются соот-

ветствующими и исключаются из поиска на последующих итерациях. Расстояние между двумя контурами вычисляется по формуле:

$$dist = (dist_{del} + dist_{cor}) / w,$$

где $dist_{del} = w_{del}(n_{\max} - n_{\min})$ – стоимость операций удаления оставшихся неразмеченных то-

чек контура и $dist_{cor} = \sum_{i=1}^{n_{\min}} d_i * (w(p_{1_{j_1(i)}}) + w(p_{2_{j_2(i)}}))$

– взвешенная сумма расстояний между соответствующими точками. Если расстояние меньше некоторого порога, то считается, что оба контура принадлежат к одному классу.

Заключительный этап двух алгоритмов состоит в выполнении проверок близких символов, для которых полученные значения расстояний распознаваемого контура от эталонных контуров этих символов могут оказаться равными или близкими друг другу. Эти проверки заключаются в вычислении дополнительных структурных и метрических признаков на контурах символов.

Рассмотрим пример структурного признака. Расстояния между контуром цифры ‘8’ на изображении и эталонными контурами символов ‘8’ и ‘В’ могут быть равны. В этом случае находится левая–нижняя точка распознаваемого контура и принимается решение ‘В’, если угол направления одного из отрезков контура с началом в этой точке близкий к 0° , а второго – к 90° , или ‘8’ в противном случае. Перед рассмотрением метрических признаков введем следующие определения.

Пусть p_j^I, p_{j+1}^I – две особые точки на контуре распознаваемого символа I , соответствующие особым точкам p_i^T, p_{i+1}^T на контуре эталона T и $f : (p_j^I, p_{j+1}^I) \rightarrow (p_i^T, p_{i+1}^T)$ – линейное преобразование (смещение, поворот и масштабирование) для совмещения этих двух пар точек. Фрагмент контура $fr(p_j^I, p_{j+1}^I)$ – последовательность отрезков кусочно-линейно аппроксимированного контура I при обходе от точки p_j^I к точке p_{j+1}^I . Два фрагмента $fr(p_j^I, p_{j+1}^I)$ и $fr(p_i^T, p_{i+1}^T)$ соответствуют друг другу, если

p_j^l соответствует p_i^T и p_{j+1}^l соответствует p_{i+1}^T . Нормализованный фрагмент контура $fr_norm(p_j^l = p_i^T, p_{j+1}^l = p_{i+1}^T)$ – фрагмент $fr(p_j^l, p_{j+1}^l)$ после выполнения линейного преобразования $f : (p_j^l, p_{j+1}^l) \rightarrow (p_i^T, p_{i+1}^T)$.

Каждый из используемых для дополнительной проверки бинарных метрических признаков равен нулю, если расстояние по метрике Хаусдорфа или метрике Фреше [12] между двумя соответствующими нормализованными фрагментами контуров превышает заданное пороговое значение, и единице в противном случае.

После распознавания всех символов выполняется коррекция результатов распознавания на основе синтаксической проверки соответствия полученной строки символов одному из предварительно заданных шаблонов. Для украинских номеров наиболее распространенными являются номера (например, АН1352МА), соответствующие шаблону: две буквы, четыре цифры, две буквы.

Результаты экспериментальной проверки

Первая часть выполненных экспериментов заключалась в проверке алгоритма уточнения положения номерного знака автомобиля на предварительно выделенных фрагментах изображения [13, 14]. Тестовая выборка состоит из 4741 изображения, каждое из которых относится к одному из следующих трех классов:

- номер, содержащий часть фона;
- часть номера;
- фрагменты изображения, не содержащие номер.

Основные результаты обработки этих изображений состоят в следующем:

1. Для 99 процентов изображений из первого класса было уточнено положение номера на основе выделения контурных линий рамки номера, а также на основе выделения одной или двух строк его символов при отсутствии рамки.

2. Для 36 процентов изображений из третьего класса было принято правильное решение о том, что они являются фоном и не содержат номерной знак.

3. На изображениях из второго класса выделены имеющиеся две или три контурные линии рамки номера и указаны направления, в которых надо расширить эти фрагменты на исходном изображении для включения всего номерного знака.

4. Среднее время обработки одного изображения (170×45) составляет 3,1 мсек на компьютере *AMD Athlon 64x2 Dual, Core Processor 4000+*, 2,11 ГГц, 3,25 ГБ ОЗУ.

Примеры изображений из указанных выше трех классов и результаты их обработки представлены на рис. 4. Примеры ошибок уточнения положения номеров (около одного процента) представлены на рис. 5.



Рис. 4. Результаты уточнения положения номеров на предварительно выделенных фрагментах изображений

Вторая часть экспериментов относится к поиску и распознаванию номеров автомобилей на изображениях. Для предварительного тестирования была сформирована сравнительно небольшая выборка из 80-ти изображений. При съемке автомобиля располагались на различных расстояниях и углах по отношению к камере, выбирались различные режимы съемки, в том числе и такие, которые приводили к искажениям цвета и яркости на изображениях. Номера автомобилей на этих изображениях соответствуют трем основным типам или шаблонам (примеры – АА1256СН, Е1081КТ, 191–91КТ), используемым в Украине. Минимальный горизонтальный размер номера на изображении (допустимый для распознавания) равен 65 клеткам. Изображения содержат всего 109 номерных знаков, горизонтальный размер которых больше 40 клеток, из которых 91 номер имеет размер больше 65 клеток. Примеры

изображений и результатов их распознавания представлены на рис. 6. Основные результаты распознавания:

- Из 91 изображения номеров, горизонтальный размер которых превышает минимальное значение (65 клеток), правильно распознано 90. Из 18 изображений номеров, размер которых больше 40, но меньше 64 клеток, правильно распознано 11.

- Среднее время обработки входного изображения размерами 800×600 и 1600×1200 составляет соответственно 0,12 с и 0,45 с.



Рис. 5. Примеры ошибок уточнения положения номеров на изображениях

С уменьшением размера номера усложняется задача его поиска и распознавания на изображении, так как возрастает уровень искажения символов при изменении угла съемки, ошибок дискретизации и бинаризации, а также вероятность их возможного слияния или касания друг с другом и с рамкой номера. С другой стороны, при распознавании номеров с малыми размерами появляется возможность более дальнего видеонаблюдения и контроля.



Рис. 6. Примеры распознавания автомобильных номеров на изображениях

Введенное ограничение 65 клеток на минимальный горизонтальный размер номера на изо-

бражении – сравнительно слабое ограничение. Так например, известная на рынке с 2015-го года система «Номерок 4» [15] обеспечивает 95 процентов правильного распознавания государственных украинских и российских номеров при выполнении следующих условий или ограничений:

- скорость передвижения автомобиля не более 120 км/ч;
- горизонтальный размер номера в кадре не менее 130 клеток;
- номер на стоп кадре распознаётся обычным человеческим глазом.

Другая известная система интеллектуального видеонализа *SecurOS Auto* [16] обеспечивает распознавание государственных номеров транспортных средств, движущихся со скоростью до 180 км/ч, с качеством распознавания до 96 процентов. При этом на сайте компании не приведены данные об условиях видеонаблюдения, а также о введенных ограничениях на размер номера в кадре.

Заключение. Рассмотренные алгоритмы локализации, обработки и распознавания номеров автомобилей на изображениях со сравнительно небольшими размерами могут быть использованы в системах видеонаблюдения в поиске и распознавании автомобильных номеров на изображениях.

Алгоритм локализации состоит из двух этапов: на первом выделяются участки изображения, которые содержат большое количество краев линий и поэтому могут содержать номерные знаки, и на втором этапе локализуются и верифицируются номерные знаки на основе выделения и связывания отрезков контура рамки номера на изображении. Алгоритм распознавания символов основан на выделении особых точек на контуре символа и вычислении структурных и метрических признаков, устойчивых к изменениям, обусловленных нефронтальным положением камеры к номерному знаку автомобиля. Предложенные алгоритмы могут быть использованы в системах видеонаблюдения для поиска и распознавания автомобильных номеров на изображениях.

1. https://en.wikipedia.org/wiki/Automatic_number_plate_recognition
2. *Wazalwar D., Oruklu E., Saniie J.* A Design Flow for Robust License Plate Localization and Recognition in Complex Scenes // *J. of Transportation Technol.* – 2012. – N 2. – P. 13–21.
3. *Martinsky O.* Algorithmic And Mathematical Principles Of Automatic Number Plate Recognition Systems, Bachelor's thesis, Depart. of Intel. Syst., Faculty of Inform. Technol., Brno Univ. of Technol., 2007. – P. 76.
4. *Shih-Hao Yu, Jun-Wei Hsieh, Yung-Sheng Chen.* Morphology-based License Plate Detection from Complex Scenes // *IEEE Proc. Of Int. Conf, on Pattern Recognition, Quebec, Canada. (Aug. 11–15, 2002).* –2002. – III. – P. 176–179.
5. *Automatic Number Plate Recognition System / Amr Badr, Mohamed M. Abdelwahab, Ahmed M. Thabet et al.* // *Annals of the University of Craiova, Mathematics and Comp. Sci. Series.* – 2011. – 38(1). – P. 62–71.
6. *Matas J., Zimmerman K.,* Unconstrained Licence Plate and Text Localization and Recognition // *IEEE Int. Conf. on Intell. Transp. Systems, New York.* – 2005. – P. 225–230.
7. *Duda R.O., Hart P.E.* Use of the Hough Transformation to Detect Lines and Curves in Pictures // *Comm. ACM, Jan. 1972.* – 15. – P. 11–15.
8. *Kiryati N., Eldar Y., Bruckstein A.M.,* A Probabilistic Hough transform // *Pattern Recognition, 1991.* – 24 (4). – P. 303–316.
9. *Шлезингер М.И.* Быстрая реализация одного класса линейных сверток. Теоретические и прикладные вопросы распознавания изображений: – К.: ИК АН УССР, 1991. – С. 61–69.
10. *Nobuyuki Otsu* A threshold selection method from gray-level histograms // *IEEE Trans. Sys., Man., Cyber.* – 1979. – 9 (1). – P. 62–66.
11. *Левентейн В.И.* Двоичные коды с исправлением выпадений, вставок и замещений символов // *Докл. АН СССР.* – 1965. – 163 (4). – С. 845–848.
12. *Шлезингер М.И., Водолазский Е.В., Яковенко В.М.* Распознавание сходства многоугольников в усиленной хаусдорфовой метрике // *Кибернетика и системный анализ.* – 2014. – № 3. – С. 174–187.
13. *Мурыгин К.В.* Обнаружение автомобильных номеров на основе смешанного каскада классификаторов // *Искусственный интеллект.* – 2010. – № 2. – С. 47–152.
14. *Мурыгин К.В.* Предварительная обработка кандидатов при обнаружении автомобильных номеров на изображениях // *Там же.* – 2013. – № 2. – С. 32–37.
15. <http://viatec.ua/product/nomerok-4>
16. <http://isscctv.com.ua/?page=SecurOSAuto>

Поступила 04.10.2017

Тел. для справок: +38 044 526-3018 (Киев)

E-mail: vkiiiko@gmail.com

© В.М. Кийко, 2017

UDC 004.89:004.93

V.M. Kyiko

PhD in Techn. Sciences, senior researcher of pattern recognition department of International Research and Training Centre of Information Technologies and Systems of the National Academy of Sciences and Ministry of Education and Sciences of Ukraine, Glushko ave., 40, Kyiv, 03187, Ukraine, vkiiiko@gmail.com

License Plate Localization and Recognition in Images

Keywords: images, license plates localization and recognition, Levenstein distance, Hough transform.

Introduction. The well known algorithms and systems for license plate localization and recognition are observed. Some ways for increasing robustness of recognition are defined.

Purpose. It is important to provide robust license plates recognition under conditions of long-distance and not frontal location of camera relatively to license plate. To reduce processing time it is desired also preliminary to detect a small number of image parts which may contain license plates and precisely localize these plates.

Methods. Localization of number plates is based on detection of license plate frame contour lines by Hough transform. While recognition special points in symbol contour are detected and modified Levenstein distances are computed between input and etalon chains of these points. Separation of similar in shape symbols is carried out by detection of additional structural and metrical contour features. Used structural and metrical features of symbols are robust in a great measure to scaling and rotations of symbols in image.

Results. The results of experimental testing of the proposed localization and recognition algorithms are presented.

Conclusion. Proposed algorithms can be used as basis for new as well as in developed before systems for license plates recognition in images.

Окончание на стр. 40

1. https://en.wikipedia.org/wiki/Automatic_number_plate_recognition.
2. *Dhawal Wazalwar, Erdal Oruklu, Jafar Saniie*, A Design Flow for Robust License Plate Localization and Recognition in Complex Scenes, *J. of Transportation Technologies*, 2012, 2, P. 13–21.
3. *Martinsky O.* Algorithmic And Mathematical Principles Of Automatic Number Plate Recognition Systems, Bachelor's thesis, Department of Intelligent Systems, Faculty of Information Technology, Brno University of Technology, 2007.
4. *Shih-Hao Yu, Jun-Wei Hsieh and Yung-Sheng Chen*, "Morphology-based License Plate Detection from Complex Scenes," *IEEE Proc. Of Int. Conf. on Pattern Recognition*, Quebec, Canada (August 11–15, 2002), **III**, P. 176–179, 2002.
5. *Amr Badr, Mohamed M. Abdelwahab, Ahmed M. Thabet*, and Ahmed M. Abdelsadek, Automatic Number Plate Recognition System, *Annals of the University of Craiova, Mathematics and Computer Science Series*, **38(1)**, 2011, P. 62–71.
6. *Matas J. and Zimmerman K.*, Unconstrained Licence Plate and Text Localization and Recognition. In: *IEEE Int. Conf. on Intell. Transp. Systems*, New York, 2005, P. 225–230.
7. *Duda R.O. and P.E. Hart*, Use of the Hough Transformation to Detect Lines and Curves in Pictures, *Comm. ACM*, **15**, 1972, P. 11–15.
8. *Kiryati N., Eldar Y. and Bruckstein A.M.*, A Probabilistic Hough transform, *Pattern Recognition*, 1991, **24**, (4), P. 303–316.
9. *Schlezinger M.I.* Fast implementation of one class of linear convolutions. In: *Theoretical and applied questions of image recognition*, Kiev: IK AN USSR, 1991, P. 61–69. (In Russian).
10. *Nobuyuki Otsu*. A threshold selection method from gray-level histograms. *IEEE Trans. Sys., Man., Cyber.* **9** (1), 1979, P. 62–66.
11. *Levenshtejn V.I.* Binary codes with correction of fallouts, insertions and symbol substitutions. *Reports AN SSSR, M.*, 1965, **163** (4), P. 845–848. (In Russian).
12. *Shlezinger M.I., Vodolazskiy E.V., Yakovenko V.M.* Recognition of the similarity of polygons in the strengthened Hausdorff metric. *Cybernetics and Systems Analysis*, 2014, **3**, P. 174–187. (In Russian).
13. *Murygin K.V.* Detection of automobile numbers on the basis of the mixed cascade of classifiers, *Artificial Intelligence*, 2010, **2**, P. 47–152. (In Russian).
14. *Murygin K.V.* Preliminary processing of candidates when car numbers on images are detected, *Artificial Intelligence*, 2013, **2**, P. 32–37. (In Russian).
15. <http://viatec.ua/product/nomerok-4>.
16. <http://isscctv.com.ua/?page=SecurOSAuto>.