

4. Тимофеев А. В., Браницкий А. А. Исследование и моделирование нейросетевого метода обнаружения и классификации сетевых атак // Information Technologies & Knowledge. 2012. Vol. 6, № 3. С. 257–265.

5. Технологии обнаружения сетевых атак. URL: <http://www.bstu.by/~opo/ru/uni/bstu/science/ids/>.

6. Дорогов А. Ю., Абатуров В. С. Экспериментальная оценка обобщающей способности нейронных сетей // Нейроинформатика-2013: сб. науч. тр. Ч. 2 / МИФИ. М., 2013. С. 244–251.

7. KDD Cup 1999. Data. URL: <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.

8. The Comprehensive R Archive Network. URL: <http://cran.r-project.org/>.

9. Дорогов А. Ю., Абатуров В. С., Раков И. В. PMML модели быстрых нейронных сетей и спектральных преобразований // Нейроинформатика-2013: сб. науч. тр. Ч. 3 / МИФИ. М., 2013. С. 156–166.

M. A. Protasova

Saint-Petersburg state electrotechnical university «LETI»

NEURAL NETWORK QUALIFIER ANOMALY TELECOMMUNICATION

The application of artificial neural networks to solve the problem of recognition and classification of anomalies in the telecommunication-network is considered. The implementation of the neural network classifier using the programming language R is considered. Selected feature space and formed the training sample of neural network. The implementation of the telecommunications network classifier anomalies is presented.

Training sample, a telecommunications network, the neural network classifier

УДК 621.391

О. Г. Бородина, В. Ю. Цветков

Белорусский государственный университет информатики и радиоэлектроники

Выделение изолированных прямых линий на изображениях с использованием форм-фактора

Предложен метод выделения изолированных прямых контурных линий на основе анализа их длин и размеров. Сущность метода состоит в сопоставлении с единицей значения форм-фактора – отношения размера контурной линии (расстояния между концевыми точками) к ее длине (числу образующих контурных пикселей). Метод позволяет повысить скорость выделения прямых контурных линий по сравнению с методами LSD и на основе преобразования Хафа.

Выделение прямых линий, контурная обработка изображений

Методы обработки перекрывающихся изображений (наложение, сравнение, построение панорам и т. д.) основаны на локализации реперных элементов. Наиболее эффективные методы локализации реперов, такие, как SIFT [1] и SURF [2], используют характерные точки. Это позволяет снизить вычислительную сложность сопоставления изображений по сравнению с корреляционными методами [3], однако приводит к низкой устойчивости результатов локализации реперов, к изменению яркости, контраста и зашумлению изображения. Прямые линии обладают значительно большей устойчивостью по сравнению с

точками. Для их выделения на изображениях широко используются методы, основанные на преобразовании Хафа [4], масочном поиске [5], вычислении градиента [6] и квантовании по ориентации [7]. Однако вычислительная сложность этих методов высока. Устранить данный недостаток можно за счет учета соотношения длин контурных линий и расстояний между их концевыми точками.

Метод выделения изолированных прямых контурных линий на основе форм-фактора. Предлагаемый метод основан на анализе значения форм-фактора – отношения размера контур-

ной линии, вычисляемого по известным координатам ее конечных точек, к длине контурной линии, определяемой суммой образующих ее контурных точек. Для прямых контурных линий данное отношение должно быть близко к единице. По этому критерию метод позволяет выделять только изолированные прямые линии, не имеющие пересечений с другими линиями. Метод позволяет снизить вычислительную сложность выделения прямых контурных линий за счет обработки относительно небольшого числа контурных пикселей.

Алгоритм выделения изолированных прямых контурных линий на основе форм-фактора состоит из следующих шагов:

1. Контурная фильтрация изображения. В результате контурной фильтрации на основе исходного изображения $I = \|i(y, x)\|_{(y=\overline{0, Y-1}, x=\overline{0, X-1})}$

формируется бинарное контурное изображение $I_B = \|i_B(y, x)\|_{(y=\overline{0, Y-1}, x=\overline{0, X-1})}$, единичные

пиксели которого определяют контурные точки, где Y, X – размеры изображения по вертикали и горизонтали.

2. Геометрическая нормализация контуров на изображении. С помощью масок (рис. 1, где a – виды образующих (прямая, косая, ломаная); b – разновидности прямой; v – разновидности косой; z – разновидности ломаной) классифицируются фрагменты контура, далее определяются избыточные контурные пиксели в этих фрагментах, которые удаляются. В результате геометрической нормализации все контуры имеют толщину в один пиксель.

3. Сегментация контурных линий. Каждому контурному пикселю $i_B(y, x)$ присваивается номер линии, которой он принадлежит. В результате формируется матрица сегментации $I_S = \|i_S(y, x)\|_{(y=\overline{0, Y-1}, x=\overline{0, X-1})}$, значения элементов которой заключены в интервале $[1, N_S]$, где N_S – число контурных линий.

4. Классификация контурных линий по числу конечных точек. Для каждой сегментированной контурной линии $l(n)$, образованной пикселями $i_B(y, x) = 1$, для которых соответствующие пиксели $i_S(y, x) = n$, где $n \in (1, N_S)$, определяется

число конечных точек. Концевой точкой является единичный пиксель $i_B(y, x)$ бинарного контурного изображения, который имеет только один соседний единичный пиксель $i_B(y + p, x + q)$, где $p = \overline{-1, 1}$, $q = \overline{-1, 1}$ и $p + q \neq 0$.

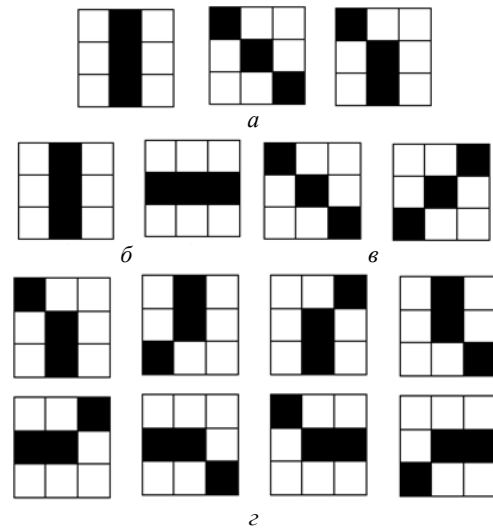


Рис. 1

5. Выделение сегментированных контурных линий с двумя конечными точками. Из множества $L_S = \{l_S(n)\}_{(n=\overline{1, N_S})}$ сегментированных контурных линий, включающего замкнутые, пересекающиеся и разомкнутые изолированные линии, выделяется подмножество $L_2 = \{l_2(n)\}_{(n=\overline{1, N_2})}$ контурных линий, которые имеют две конечные точки, где N_2 – число сегментированных контурных линий с двумя конечными точками. Как правило, пересекающиеся контурные линии имеют более двух конечных точек, поэтому на данном шаге алгоритма выделяются в основном изолированные (не имеющие пересечений) контурные линии. Исключением является пересечение замкнутой и разомкнутой контурных линий. Такое исключение обрабатывается на следующих шагах алгоритма.

6. Определение длин сегментированных контурных линий. Для каждой контурной линии $l_2(n)$, имеющей две конечные точки, определяется число $s_2(n)$ образующих данную линию контурных пикселей $i_B(y, x)$ с помощью выражения

$$s_2(n) = \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} t(m) i_B(y, x) k(n, y, x),$$

где $k(n, y, x) = \begin{cases} 0 & \text{при } i_S(y, x) \neq n, \\ 1 & \text{при } i_S(y, x) = n; \end{cases}$ $t(m)$ – ко-

эффициент, зависящий от маски, к которой принадлежит пиксель. Число $s(n)$ интерпретируется как длина контурной линии.

7. Определение размеров сегментированных контурных линий. Для каждой контурной линии $l_2(n)$, имеющей две конечные точки, вычисляется расстояние $r_2(n)$ между конечными точками (размер контурной линии) с помощью выражения

$$r_2(n) = \sqrt{(y_1(n) - y_2(n))^2 + (x_1(n) - x_2(n))^2},$$

где $(y_1(n), x_1(n)), (y_2(n), x_2(n))$ – координаты конечных точек контурной линии $l_2(n)$.

8. Вычисление форм-фактора линии. Для каждой контурной линии $l_2(n)$, имеющей две конечные точки, вычисляется значение форм-фактора $f_2(n)$ линии с помощью выражения

$$f_2(n) = r_2(n) / s_2(n).$$

9. Анализ форм-фактора линии. Для каждой контурной линии $l_2(n)$, имеющей две конечные точки, значение ее форм-фактора $f_2(n)$ сравнивается с единицей. Если $f_2(n)$ близко к единице, то контурная линия $l_2(n)$ является прямой. В противном случае ($f_2(n) < 1$) линия является кривой или ломаной. Близость значения $f_2(n)$ линии к единице оценивается в результате сопоставления заданного порога T_c , определяющего кривизну линии, с модулем разности, вычисляемым с помощью выражения $|1 - f_2(n)|$. Контурная линия $l_2(n)$ является прямой, если $|1 - f_2(n)| \leq T_c$. В результате выполнения данного шага алгоритма из подмножества L_2 выделяется подмножество $L_D = \{l_D(n)\}_{(n=1, N_D)}$ изолированных прямых контурных линий, где N_D – число изолированных прямых контурных линий.

10. Окончание алгоритма. Фиксируется информация о найденных изолированных прямых контурных линиях (координаты $(y_1(n), x_1(n))$,

$(y_2(n), x_2(n))$ конечных точек, длина $s_2(n)$, значение форм-фактора $f_2(n)$).

Оценка эффективности методов выделения прямых контурных линий. Для оценки эффективности предложенного метода выделения изолированных прямых контурных линий проведено его сравнение с методом LSD (Line Segment Detector) [6] и методом на основе преобразования Хафа [4]. В качестве критериев эффективности использованы время и устойчивость результатов выделения прямых контурных линий на изображении.

В таблице приведено время выделения прямых контурных линий на тестовом изображении 600×600 пикселей, представленном на рис. 1, а, для предложенного метода (FFLD – Form-Factor Line Detection), а также методов LSD и на основе преобразования Хафа (HT). Компьютер, на котором проводились эксперименты, имеет следующие характеристики: число ядер – 2; частота процессора – 2.0 ГГц; частота шины – 250 МГц; размер кеш-памяти – 2.0 Гбайт. Для контурной обработки в предложенном методе и методе на основе преобразования Хафа использован фильтр Канны. Из таблицы следует, что предложенный метод FFLD обеспечивает выигрыш в скорости выделения прямых контурных линий в 4 раза по сравнению с методом LSD и в 14.8 раза по сравнению с методом на основе преобразования Хафа.

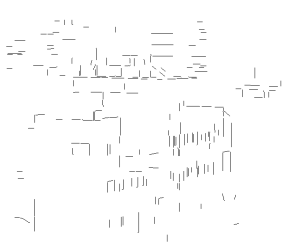
На рис. 2, б–г приведены результаты выделения прямых контурных линий на изображении, представленном на рис. 2, а, с помощью методов FFLD (б), LSD (в) и на основе преобразования Хафа (г). На рис. 3 и 4 представлены характеристики устойчивости (S) результатов выделения прямых контурных линий при изменении яркости (ΔY) и контраста (ΔC) изображения соответственно. Для количественной оценки устойчивости произведено сопоставление числа пикселей на прямых контурных линиях, сохранивших свое местоположение после изменения яркости и контраста изображения, с общим числом пикселей на прямых контурных линиях, выделенных на исходном тестовом изображении.

Установлено, что предложенный метод FFLD выигрывает в устойчивости определения местоположения выделяемых прямых контурных линий по сравнению с методом на основе преобра-

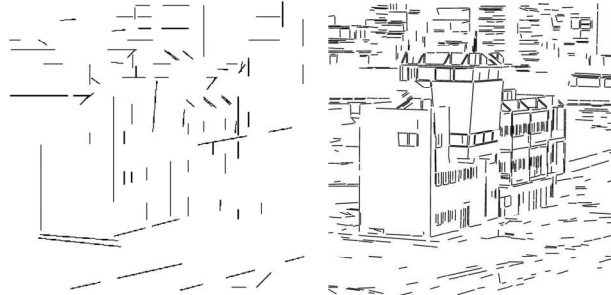
Методы выделения прямых линий	FFLD	LSD	HT
Время выделения прямых линий, с	0.1	0.4	1.48



а



б



в

г

Рис. 2

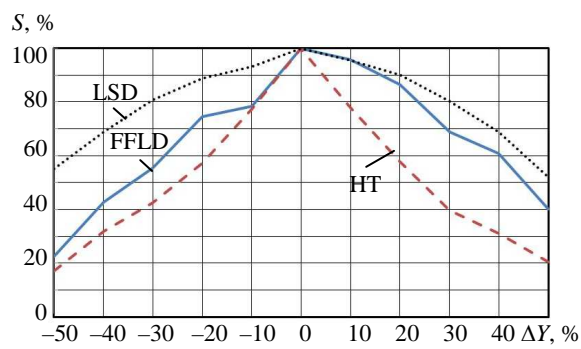


Рис. 3

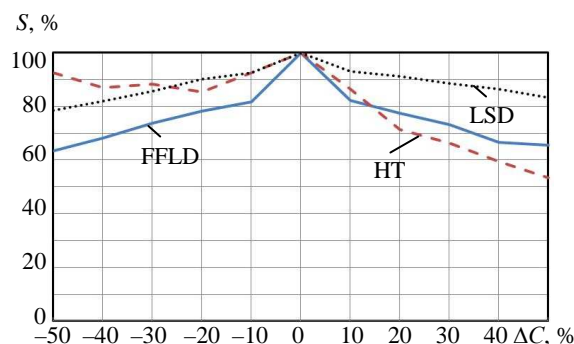


Рис. 4

зования Хафа до 30 и 20 %, но проигрывает методу LSD до 10 и 30 % при увеличении и уменьшении яркости изображения соответственно (рис. 3). При увеличении контраста метод FFLD выигрывает в устойчивости у метода на основе преобразования Хафа до 10 %, но проигрывает методу LSD до 20 % (рис. 4). При уменьшении контраста метод FFLD проигрывает в устойчивости методам на основе преобразования Хафа и LSD до 30 и 15 % соответственно.

Таким образом, предложен метод выделения изолированных прямых контурных линий на основе форм-фактора. Сущность метода состоит в сопоставлении с единицей значения форм-фактора – отношения размера контурной линии (расстояния между концевыми точками) к ее длине (числу образующих контурных пикселей). Для изолированных прямых контурных линий форм-

фактор близок к единице. Установлено, что предложенный метод обеспечивает выигрыш в скорости выделения прямых контурных линий в 4 раза по сравнению с методом LSD и в 14.8 раза по сравнению с методом на основе преобразования Хафа. Показано, что предложенный метод выигрывает в устойчивости определения местоположения выделяемых прямых контурных линий по сравнению с методом на основе преобразования Хафа до 30 %, но проигрывает методу LSD до 30 % при изменении яркости изображения. При увеличении контраста предложенный метод выигрывает в устойчивости у метода на основе преобразования Хафа до 10 %, но проигрывает методу LSD до 20 %. При уменьшении контраста предложенный метод проигрывает в устойчивости методам на основе преобразования Хафа и LSD до 30 и 15 % соответственно.

СПИСОК ЛИТЕРАТУРЫ

1. Lowe D. Distinctive image features from scale invariant keypoints // Intern. J. of Computer Vision. 2004. Vol. 60, № 2. P. 91–110.
2. Bay H. SURF: Speeded up robust features // Proc. of the 9th Europ. Conf. on Computer Vision, Graz, Austria, 2006. Vol. 3951. P. 404–410.
3. Hirschmuller H., Garibaldi J. M. Realtime correlation-based stereo vision with reduced border errors // Intern. J. of Computer Vision. 2002. Vol. 47, № 1–3. P. 229–246.
4. Duda R. O. Use of the Hough Transformation to Detect Lines and Curves in Pictures // Communication of the ACM. 1972. Vol. 15, № 1. P. 229–246.

5. Anver M. M., Stonier R. J. Fuzzy edge detection using competition between multiple masks // Proc. of the 2nd Intern. Conf. on Computational Intelligence, Robotics and Autonomous Systems, CIRAS 2003, Singapore, 2003. P. 344–348.
6. Grompone von Gioi R. LSD: A Fast Line Segment Detector with a False Detection Control // IEEE Transactions on Pattern. Analysis and Machine Intelligence. 2010. Vol. 32, № 4. P. 722–732.
7. Chan T. S., Raymond K. K. Line detection algorithm // Proc. of 13th Intern. Conf. on Pattern Recognition, ICPR 1996, Vienna, 1996. P. 126–130.

O. G. Baradzina, V. Yu. Tsviatkou

Belarusian State University of Informatics and Radioelectronics

LOCALIZATION OF ISOLATED STRAIGHT LINES ON THE IMAGES USING THE FORM-FACTOR

We propose a method for localization of isolated contour lines based on form-factor. The essence of the method is compared with the unit value of form-factor – the ratio of the contour line size (the distance between the end points) to its length (number of pixels forming the contour). The method allows to improve the rate of direct contour lines localization compared with LSD method and method based on Hough transform.

Allocation of direct lines, contour image processing
