

## МЕТОД РАЗРЕЖЕННОГО ФРОНТА ДЛЯ ВЕКТОРИЗАЦИИ ЛИНЕЙЧАТЫХ ИЗОБРАЖЕНИЙ

А.А. КОВТУН

Предложен новый метод векторизации линейчатых изображений. В его основе лежит алгоритм разреженно-пиксельного отслеживания прямых и кривых линий на растровом изображении. Результатом работы данного алгоритма является множество траекторий линий в виде последовательностей точек. Новизна метода заключается в использовании весовых коэффициентов при расчете точек траекторий, что обеспечивает уменьшение зависимости результатов векторизации от зашумленности контуров линий на растровом изображении. Также предложен эффективный алгоритм противодействия повторному отслеживанию линии данным методом. На втором, заключительном этапе векторизации полученные траектории преобразуются во множество векторных примитивов — отрезков и дуг, совокупность которых аппроксимирует прямые и кривые линии и образует векторное изображение. Алгоритм обладает высокой производительностью и может работать без настройки параметров. Приведены сравнительные исследования производительности алгоритма и качества результатов его работы.

### ВВЕДЕНИЕ

Векторизация является преобразованием растровых изображений в векторные. Векторные объекты используются для высокоуровневого анализа в задачах распознавания образов, поэтому векторизация важна как базовый этап решения таких задач. Векторизация как раздел компьютерного зрения на сегодняшний день является областью, актуальной для исследований, поскольку существующие методы не позволяют получить достаточно качественный результат данного преобразования одновременно с высокой производительностью и, следовательно, должны быть усовершенствованы.

Данный метод векторизации был разработан для осуществления базового этапа решения задачи распознавания образов на изображениях рукописных линейчатых объектов, например электронных схем. Основным желаемым свойством получаемых векторных изображений было представление прямых элементов прямыми линиями, а элементов окружностей — соответственно дугами. Также ценными для обеспечения удобства применения метода характеристиками являлись не требующая настройки параметров работа алгоритма на растровых изображениях с различными размерами и свойствами, а также высокая скорость работы, которая позволяет сократить использование вычислительных ресурсов.

Существующие методы можно классифицировать в 4 группы [1], [2].

Первая группа является методами, основанными на преобразовании Хафа. В их основе лежит преобразование исходного изображения по заданному правилу и получению пространства параметров, из которого затем извлекаются объекты, параметры которых находятся в локальных максимумах этого пространства.

Вторая группа основана на процедурах определения серединных линий исходных линий на растровом изображении. Процедуры могут быть реализованы алгоритмами утончения, скелетизации линий (рис. 1, *а*), отслеживания контуров.

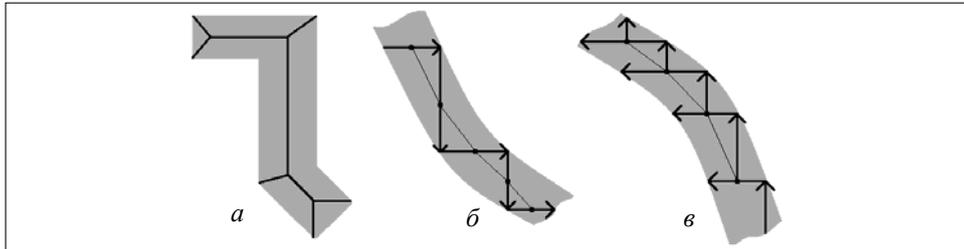


Рис. 1. Иллюстрация работы существующих методов векторизации: *а* — скелетизации; *б* — OZZ; *в* — SPV

Третья группа базируется на предварительной обработке растрового изображения, которая заключается в его представлении в виде количественных характеристик соседних пикселей с последующим анализом этих характеристик.

Четвертая группа — разреженно-пиксельные методы, позволяющие использовать для осуществления векторизации только часть пикселей растрового изображения. Данная особенность может быть использована при разработке разреженно-пиксельного метода. Примеры существующих методов группы — Orthogonal Zig-Zag (OZZ), Sparse Pixel Vectorization (SPV), Line Net Global (LNG), волновой [3].

Метод OZZ (рис. 1, *б*) заключается в отслеживании линий на растровом изображении последовательностями вертикальных и горизонтальных ходов, оканчивающихся при достижении края линии. Недостатком данного метода является малая точность построения траекторий, особенно для кривых линий. Метод SPV (рис. 1, *в*) обеспечивает более точное построение траектории по сравнению с OZZ, отличаясь более сложной, но точной процедурой поиска первой точки линии на растровом изображении и использованием одинаковых циклов ходов для любых линий [4]. Волновой метод и LNG, в отличие от OZZ и SPV, обеспечивают отслеживание линий на растровом изображении с произвольными направлениями, не ограничиваясь вертикальными и горизонтальными ходами [5]. В ходе отслеживания с помощью метода LNG сохраняются данные об угле поворота линии, что обеспечивает в случае достижения соединений линий отслеживание преимущественно в прежнем направлении.

## МЕТОД РАЗРЕЖЕННОГО ФРОНТА

Предлагаемый метод обеспечивает в ходе отслеживания линий сканирование разреженных, отдельно расположенных пикселей. Отслеживание осуществляется с произвольными направлениями. Процедура выбора направления отслеживания при определении новой точки траектории является основополагающей для свойств метода, отличается от таковой в существующих методах и позволяет осуществлять векторизацию полноцветных растровых изображений, частично сглаживать получаемые траектории, преодолевать пересечения линий с приоритетом направления «прямо» и выбором линии с большей толщиной и контрастностью относительно фона.

Отдельное внимание при разработке метода было уделено проблеме пометки посещенных участков изображения, которая особенно актуальна из-за сканирования только отдельно расположенных пикселей. Так как толщины линий на растровых изображениях могут варьироваться в широких пределах, расстояния между сканируемыми пикселями линий могут быть достаточно большими. Соответственно, вероятность повторного отслеживания с использованием еще не сканированных пикселей может быть высокой. Для решения этой проблемы был разработан метод решеток покрытий (рис. 2), который заключается в пометке не пикселей растрового изображения, а соответствующих пикселей нескольких решеток с разным шагом. Решетки покрытий являются двумерными массивами, размеры которых соответствуют размерам растра, поделенным на числа от 1 до  $N$  со степенным шагом, и позволяют покрывать траектории с шагом между соседними точками до  $N$  пикселей.

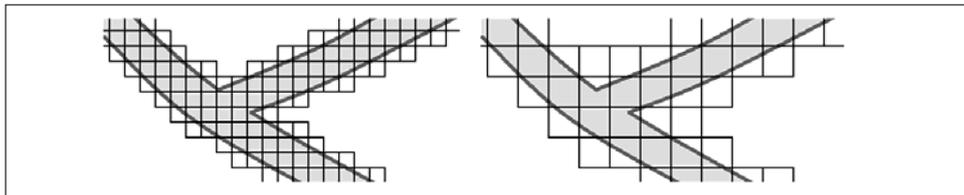


Рис. 2. Покрытие линий решетками покрытий с разными размерами ячеек

### Обнаружение линий на растровом изображении

Для поиска линий сканирование пикселей растрового изображения осуществляется как в LNG — в шахматном порядке, хотя применение другого способа сканирования несущественно влияет на работу метода. Для обеспечения работы с полноцветными растровыми изображениями с разной общей или локальной яркостью обнаружение линии производится по жестко заданному или настраиваемому критерию контрастности по сравнению с несколькими близкими уже отсканированными пикселями. Используется один цветовой канал растрового изображения или усредненные по цветовым каналам значения. Если обнаруженный пиксель линии покрыт решеткой покрытия с ячейкой размером в 1 пиксель, то отслеживание длинной линии уже осуществлялось; тогда сканирование производится дальше.

После обнаружения линии осуществляется шаг отслеживания базовой длины в 1 пиксель в пробном направлении. Если после такого шага зафиксировано покидание линии, то берется следующее пробное направление. Всего имеется заданное количество  $K$  направлений с шагом  $2\pi/K$  радиан. Если все пробные направления привели к покиданию линии (к попаданию текущей точки отслеживания за пределы линии на растровом изображении), то обнаруженная линия является отдельно расположенным пикселем.

Для повышения качества векторизации используется 3 вида отслеживания — пробное, прямое и обратное (рис. 3). Как только в результате пробного шага получена точка траектории, осуществляется пробное отслеживание до достижения заданного количества  $M$  шагов или до покидания линии. Такая операция нужна для перемещения начальной точки траектории от края к середине линии и для установления величины шага соответственно толщине линии. Покрытие точек пробного участка траектории не осуществляется.



Рис. 3. Прямое и обратное отслеживание линии

В случае достижения числа  $M$  шагов последняя точка пробного участка траектории становится стартовой. Начиная с нее, производится прямое (в направлении  $M$ -го шага) и обратное (в противоположном направлении) отслеживание линии с покрытием точек траектории решетками, ячейки которых не крупнее текущего. В случае покидания линии на пробном участке траектории будет производиться только обратное отслеживание, начиная с последней точки пробного участка в качестве стартовой точки. Таким образом, после обнаружения линии в любой ее точке происходит отслеживание сразу всей линии.

### Разреженный фронт в итерационной процедуре отслеживания линий

В основе осуществления каждого шага траектории лежит определение ее следующей точки с помощью построения разреженного фронта и анализа его пикселей. Разреженный фронт (рис. 4) — множество из заданного нечетного числа  $S$  точек, равномерно расположенных на дуге с радиусом, равным текущему шагу отслеживания, серединным углом, равным текущему направлению отслеживания, и заданным углом раскрытия, как правило, около  $120^\circ$ . Пиксели, на которых находятся точки фронта, являются пикселями фронта.

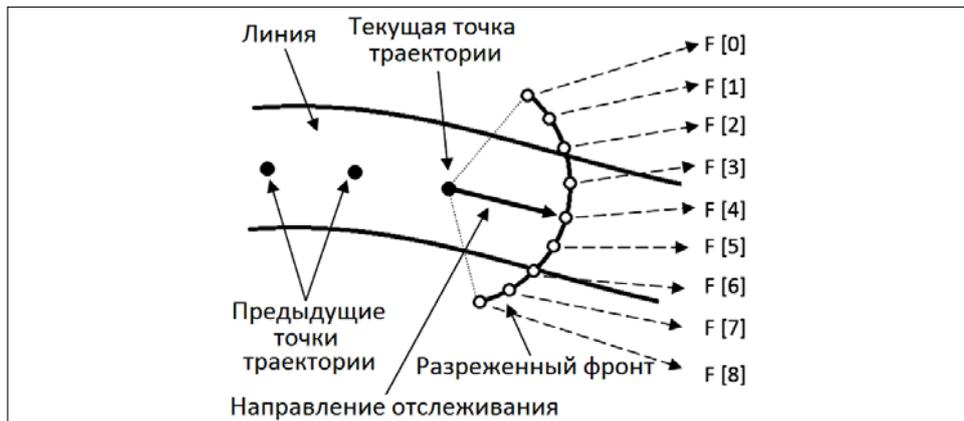


Рис. 4. Разреженный фронт и данные в его точках

После получения пикселей фронта вычисляются веса точек фронта путем умножения значений яркостей пикселей, нормированных в пределах фронта, на весовые коэффициенты. Весовые коэффициенты — это множество значений распределения Гаусса в точках  $\{-L, L\}$  и 0 в остальных точках целочисленного множества  $\left\{-\frac{S-1}{2}, \frac{S-1}{2}\right\}$ . Приравниваются к нулю значе-

ния точек фронта, не лежащих на линии и соответственно меньшие заданной величины (как правило, 1/3). Параметр  $L$  определяет максимальную величину поворота на трех последовательных точках траектории. Таким образом предотвращается покидание линии на растровом изображении из-за его зашумленности. Распределение Гаусса обеспечивает большие веса направлений траектории, близких к направлению на центр фронта, и, следовательно, увеличение приоритета возможных следующих направлений траектории, близких в текущему, в случае пересечения или примыкания линий, а также уменьшение влияния зашумленности краев линии.

Угол следующего направления отслеживания вычисляется как угол, соответствующий математическому ожиданию распределения весов точек фронта, а следующая точка — как точка фронта, соответствующая найденному следующему направлению. Нулевое значение суммы всех весов свидетельствует о резком повороте линии и соответственно об окончании отслеживания данного ее участка (следующий участок, если он есть, будет обнаружен как новая линия). Отслеживание прекращается также в случае покидания линии, что определяется по контрастности пикселей фронта, текущей и следующих точек. Еще одним условием завершения отслеживания является наличие покрытия найденной точки какими-либо решетками покрытий.

Последний этап итерации — определение следующего значения величины шага отслеживания. Так как за пределами линии веса фронта равны нулю и радиус фронта известен, можно определить ширину линии, зная значения весов и приравнять ей величину шага.

### Извлечение векторных примитивов

После получения всех траекторий из растрового изображения имеются такие данные, как последовательности углов и расстояний между точками траекторий. Основываясь на этих данных, производится аппроксимация траекторий векторными примитивами — отрезками и дугами (рис. 5).

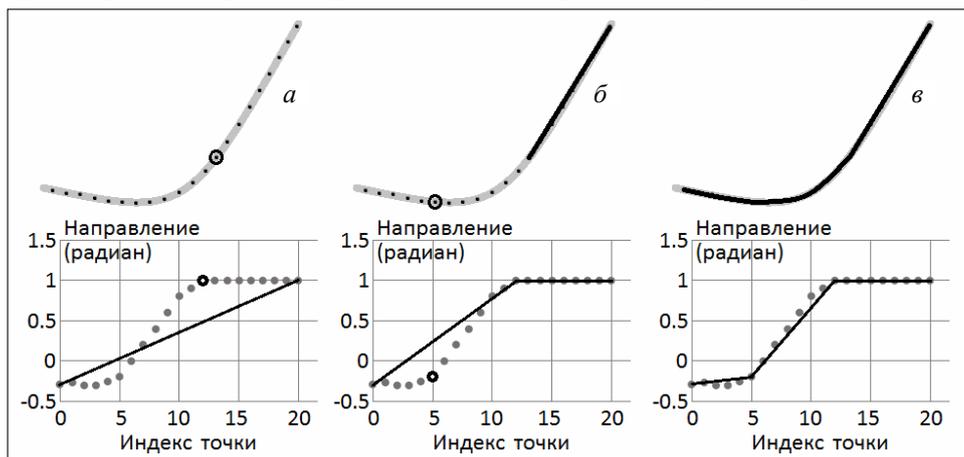


Рис. 5. Деление траектории: текущие участки аппроксимации линии растрового изображения (вверху) и соответствующие участки в пространстве индексов и углов направлений точек траектории (внизу)

Для каждой траектории составляется уравнение прямой, проходящей по ее первой и последней точкам. Область определения прямой как функции

содержит индексы направлений траектории, а область значений — величины углов этих направлений. Определяется индекс точки, для которой величина отклонения значения угла направления от угла наклона построенной прямой наибольшая. Если отклонение превышает заданную пороговую величину, то область определения делится этой точкой на две, для которых процедура повторяется рекурсивно. Если же отклонение не превышает пороговой величины, данный участок траектории аппроксимируется векторным примитивом. Пороговое значение отклонения определяет допустимую погрешность аппроксимации участка траектории. Пример деления траектории на участки с их аппроксимацией показан на рис. 5.

Выбор векторного примитива (отрезка или дуги) определяется заданной пороговой величиной коэффициента наклона прямой, проведенной через крайние точки текущего участка траектории в пространстве индексов и углов направлений.

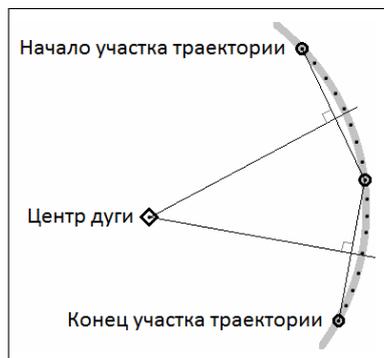


Рис. 6. Нахождение центра дуги для участка траектории

Горизонтальная прямая означает постоянство угла и соответствует отрезку, а наклонная — равномерное изменение угла и соответствует дуге. В качестве координат концов отрезка принимаются координаты концов соответствующего участка траектории. Для дуги (рис. 6) вычисляются координаты центра как точки пересечения серединных перпендикуляров отрезков, образованных первой, средней и последней точками данного участка траектории. Затем вычисляется угол раскрытия дуги как величина разности первого и последнего углов направлений участка траектории. Радиус дуги вычисляется как расстояние между центром и средней точкой участка траектории.

## ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ И КАЧЕСТВА РАБОТЫ МЕТОДА

Для проведения сравнительного исследования получаемых характеристик предлагаемого метода была выбрана распространенная свободная библиотека компьютерного зрения OpenCV (версии 2.4) [6]. Предлагаемый в ней метод векторизации заключается в выделении границ растрового изображения с помощью алгоритма Кэнни с последующим извлечением векторных примитивов (только отрезков) с помощью вероятностного преобразования Хафа [7, 8]. Вероятностное преобразование Хафа, как и предлагаемый метод, относится к группе разреженно-пиксельных методов, поскольку в данном варианте преобразования Хафа сканируется только часть пикселей растрового изображения.

Исследуемыми характеристиками являются время и качество векторизации, количество извлеченных векторных примитивов и распределение их длин, а также зависимость этих параметров от растрового изображения и его размера. В качестве параметра, позволяющего оценить участие наиболее длинных векторных примитивов в формировании векторного изображения, вводится коэффициент длины — относительное количество наиболее длинных векторных примитивов, суммарная длина которых составляет не менее 0,5 общей длины всех примитивов.

### Методика измерения качества векторизации

Качество векторизации может быть численно определено как мера отличия полученного векторного изображения от исходного растрового. Для подготовки этих изображений к сравнению между собой осуществляется их обработка. К исходным растровым изображениям применяется алгоритм адаптивной бинаризации с фиксированными параметрами [9]. Линии и дуги векторных изображений строятся с толщиной в 1 пиксель на растре с размерами, равными размерам исходного растрового изображения. Таким образом, все тестируемые изображения приводятся к монохромным изображениям одинаковых размеров — к обработанным растровым (исходным растровым после применения бинаризации) и обработанным векторным (полученным векторизацией векторным, отображенным на растр).

Высокое качество векторного изображения достигается в случае достаточно полного совпадения его формы (топологии) с формой исходного растрового изображения. Полное совпадение может быть интерпретировано как покрытие линий на обработанном растровом изображении линиями на обработанном векторном (рис. 7) и наоборот, с нулевым относительным количеством непокрытых пикселей. Так как из-за погрешностей векторизации и обработки изображений покрытия будут неполными, они осуществляются не непосредственно пикселями, а множествами кругов варьируемого радиуса  $D$ , центрами которых являются пиксели линий.

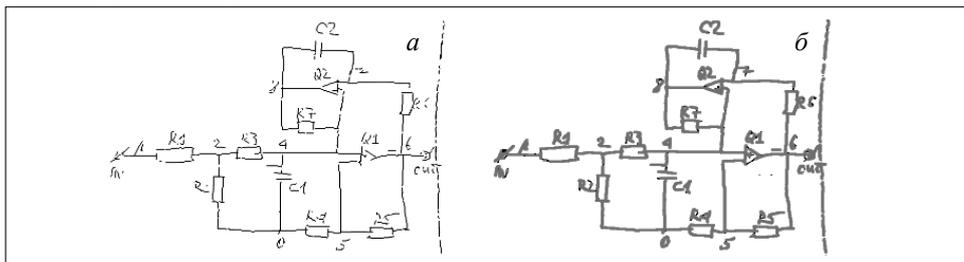


Рис. 7. Обработанное векторное изображение: а — полученное предложенным методом; б — покрытие им обработанного растрового изображения пикселей

Определим количество пикселей покрываемого обработанным растровым изображением обработанного векторного как  $Nv$ , а количество его непокрытых пикселей как  $uNb$ . Определим количество пикселей покрываемого обработанным векторным изображением обработанного растрового как  $Nb$ , а количество его непокрытых пикселей как  $uNv$ . Тогда качество векторизации  $Q$  равно:

$$Q = \left( 1 - 0,5 \frac{uNv}{Nv} - 0,5 \frac{uNb}{Nb} \right) 100\%.$$

### Сравнительные испытания алгоритма и интерпретация результатов

В качестве тестовых исходных растровых изображений взяты фотографии рукописной и печатанной электронных схем, соответственно *Circuit1* (рис. 8,а) и *rca3040* (рис. 9,а), с размерами, приведенными к 0,3, 1 и 3 миллиона пикселей.

Для каждого изображения проводится 4 группы испытаний. Группы различаются по параметрам  $D$  — радиусу кругов покрытий, и  $m$  — минималь-

ной длине учитываемых в испытании векторных примитивов. Для I и II групп  $D_{I,II} = \frac{\sqrt{S}}{200}$ , для III и IV групп  $D_{III,IV} = \frac{\sqrt{S}}{100}$ , где  $S$  — количество пикселей в растре изображения. Для I и III групп  $m = 0$ , для II и IV групп  $m = D$ . Для каждого изображения проводится одно испытание предлагаемого метода с одинаковыми для всех изображений настройками и несколько испытаний преобразования Хафа с помощью OpenCV. Испытания преобразования Хафа для всех изображений начинаются с одинаковыми параметрами, равными параметрам в примере из документации к OpenCV. Параметры меняются по одному, пока качество векторизации в I группе испытаний станет не меньше, чем для предлагаемого метода. Результаты испытаний для изображений на рис. 8,а и на рис. 9,а отображены в табл. 1–6. Параметры преобразования Хафа указаны как значения  $H/T/L/G$  (CannyHigh, threshold, minLength, minGap).

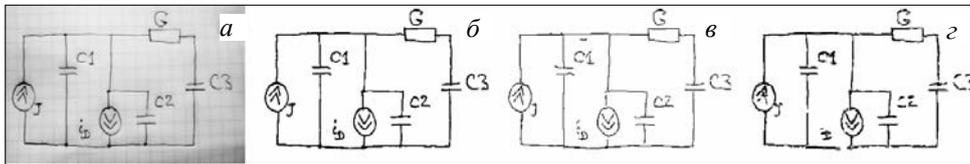


Рис. 8. Изображение *Circuit1*: а — исходное; б — обработанное растровое; в — обработанное векторное, полученное предлагаемым методом; г — полученное преобразованием Хафа с наилучшим для испытаний качеством

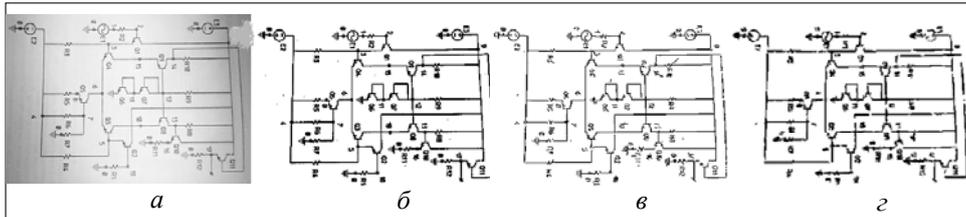


Рис. 9. Изображение *rca3040*: а — исходное; б — обработанное растровое; в — обработанное векторное, полученное предлагаемым методом; г — полученное преобразованием Хафа с наилучшим для испытаний качеством

Для предлагаемого метода и для преобразования Хафа с наборами параметров, обеспечивающими наилучшее качество векторизации, построены графики (рис. 10) отношения времени работы к размеру растрового изобра-

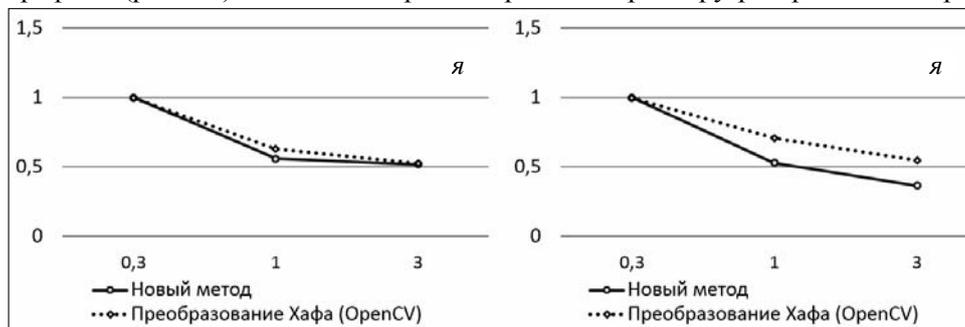


Рис. 10. Графики оценок вычислительной сложности методов (в зависимости от размеров растровых изображений): а — соответствует обработке изображения на рис. 8,а; б — соответствует обработке изображения на рис. 9,а

жения для обоих методов в зависимости от размера. Графики нормированы относительно величин, полученных для изображений с размерами 0,3 млн. пикселей, и позволяют оценить практическую вычислительную сложность сравниваемых методов.

**Таблица 1.** Параметры и результаты сравнительных испытаний методов векторизации для изображения Circuit1 размером в 0,3 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)				
Параметры H/T/L/G		–	200/50/ 50/10	200/50/ 5/10	200/20/ 5/10	200/20/ 5/5	200/15/ 5/5
Время векторизации		15 мс	35 мс	44 мс	47 мс	42 мс	43 мс
Группа I $D=3$ $m=0$	Число линий	329	95	103	332	387	444
	Коэф. длины	0,13	0,27	0,28	0,24	0,26	0,23
	Качество, %	98,6	82,6	82,4	96,9	97,6	98,8
Группа II $D=3$ $m=3$	Число линий	270	95	103	332	387	444
	Коэф. длины	0,14	0,27	0,28	0,24	0,26	0,23
	Качество, %	98,0	82,6	82,4	96,9	97,6	98,8
Группа III $D=6$ $m=0$	Число линий	329	95	103	332	387	444
	Коэф. длины	0,13	0,27	0,28	0,24	0,26	0,23
	Качество, %	99,8	85,8	86,2	99,7	99,2	99,7
Группа IV $D=6$ $m=6$	Число линий	168	95	103	318	356	332
	Коэф. длины	0,19	0,27	0,28	0,24	0,27	0,25
	Качество, %	97,5	85,8	86,2	99,6	99,2	99,7

**Таблица 2.** Параметры и результаты сравнительных испытаний методов векторизации для изображения Circuit1 размером в 1 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)		
Параметры H/T/L/G		–	200/50/50/10	200/50/10/10	200/20/10/10
Время векторизации		28 мс	81 мс	93 мс	90 мс
Группа I $D=5$ $m=0$	Число линий	379	51	236	579
	Коэф. длины	0,09	0,41	0,29	0,28
	Качество, %	99,2	68,5	87,0	99,2
Группа II $D=5$ $m=5$	Число линий	256	51	236	579
	Коэф. длины	0,12	0,41	0,29	0,28
	Качество, %	98,7	68,5	87,0	99,2
Группа III $D=10$ $m=0$	Число линий	379	51	236	579
	Коэф. длины	0,09	0,41	0,29	0,28
	Качество, %	100,0	71,1	91,2	99,8
Группа IV $D=10$ $m=10$	Число линий	164	51	232	567
	Коэф. длины	0,16	0,41	0,29	0,28
	Качество, %	99,1	71,1	91,2	99,8

**Таблица 3.** Параметры и результаты сравнительных испытаний методов векторизации для изображения Circuit1 размером в 3 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)				
Параметры Н/Т/Л/Г			200/50/50/10	100/50/50/10	100/50/40/10	100/35/30/10	100/35/30/20
Время векторизации		78 мс	202 мс	210 мс	222 мс	208 мс	225 мс
Группа I D=9 m=0	Число линий	800	499	645	319	377	586
	Коэф. длины	0,06	0,26	0,23	0,32	0,34	0,30
	Качество, %	99,0	92,1	95,4	88,8	94,4	99,4
Группа II D=9 m=9	Число линий	325	499	645	319	377	586
	Коэф. длины	0,12	0,26	0,23	0,32	0,34	0,30
	Качество, %	98,5	92,1	95,4	88,8	94,4	99,4
Группа III D=17 m=0	Число линий	800	499	645	319	377	586
	Коэф. длины	0,06	0,26	0,23	0,32	0,34	0,30
	Качество, %	99,9	95,1	97,5	92,1	97,0	99,8
Группа IV D=17 m=17	Число линий	189	389	449	319	377	586
	Коэф. длины	0,16	0,29	0,28	0,32	0,34	0,30
	Качество, %	99,1	94,0	96,1	92,1	97,0	99,8

**Таблица 4.** Параметры и результаты сравнительных испытаний методов векторизации для изображения gsa3040 размером в 0,3 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)			
Параметры Н/Т/Л/Г			200/50/50/10	200/50/3/10	200/50/3/3	200/20/3/3
Время векторизации		47 мс	57 мс	62 мс	65 мс	66 мс
Группа I D=3 m=0	Число линий	1142	261	294	432	1086
	Коэф. длины	0,13	0,22	0,19	0,16	0,16
	Качество, %	98,3	87,3	88	87,9	98,7
Группа II D=3 m=3	Число линий	805	261	291	419	1021
	Коэф. длины	0,14	0,22	0,19	0,17	0,17
	Качество, %	97	87,3	88	87,7	98,4
Группа III D=6 m=0	Число линий	1142	261	294	432	1086
	Коэф. длины	0,13	0,22	0,19	0,16	0,16
	Качество, %	99,8	92,7	93,2	93,1	99,8
Группа IV D=6 m=6	Число линий	390	261	274	339	722
	Коэф. длины	0,15	0,22	0,20	0,20	0,19
	Качество, %	95,6	92,7	92,9	91,5	99,2

**Таблица 5.** Параметры и результаты сравнительных испытаний методов векторизации для изображения gsa3040 размером в 1 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)			
Параметры Н/Т/Л/Г			200/50/50/10	200/50/5/10	200/50/5/5	200/25/5/5
Время векторизации		83 мс	135 мс	152 мс	156 мс	156 мс
Группа I D=5 m=0	Число линий	1606	151	722	874	1798
	Коэф. длины	0,09	0,29	0,15	0,15	0,18
	Качество, %	99,1	78,2	95,4	95,2	99,4

Продолжение табл. 5

Группа II $D=5$ $m=5$	Число линий	900	151	714	850	1744
	Коэф. длины	0,10	0,29	0,15	0,15	0,18
	Качество, %	97,6	78,2	95,4	95,0	99,3
Группа III $D=10$ $m=0$	Число линий	1606	151	722	874	1798
	Коэф. длины	0,09	0,29	0,15	0,15	18,00
	Качество, %	99,9	80,6	98,3	98,3	99,9
Группа IV $D=10$ $m=10$	Число линий	353	151	623	617	1091
	Коэф. длины	0,11	0,29	0,16	0,18	0,20
	Качество, %	94,2	80,6	97,9	96,9	99,5

**Таблица 6.** Параметры и результаты сравнительных испытаний методов векторизации для изображения gsa3040 размером в 3 млн. пикселей

Метод векторизации		Новый	Преобразование Хафа (OpenCV)		
Параметры H/T/L/G		–	200/50/50/10	200/50/10/10	200/40/10/10
Время векторизации		172 мс	335 мс	358 мс	361 мс
Группа I $D=9$ $m=0$	Число линий	2723	272	1525	1915
	Коэф. длины	0,08	0,31	0,19	0,19
	Качество, %	99,2	79,5	97,9	99,2
Группа II $D=9$ $m=9$	Число линий	1023	272	1525	1915
	Коэф. длины	0,11	0,31	0,19	0,19
	Качество, %	97,4	79,5	97,9	99,2
Группа III $D=17$ $m=0$	Число линий	2723	272	1525	1915
	Коэф. длины	0,08	0,31	0,19	0,19
	Качество, %	99,6	82,4	99,2	99,8
Группа IV $D=17$ $m=17$	Число линий	422	272	1138	1310
	Коэф. длины	0,10	0,31	0,22	0,22
	Качество, %	94,8	79,5	98,8	99,3

Для I и III групп испытаний длины учитываемых векторных примитивов не ограничиваются снизу. Для II и IV групп испытаний учитываются только векторные примитивы, длины которых больше заданной пороговой величины  $m$ . В связи с большой долей коротких векторных примитивов, получаемых с помощью предлагаемого метода, задание пороговой величины  $m$  уменьшает их количество относительно преобразования Хафа. Это подтверждается данными табл. 1–6. Для II групп предлагаемый метод обеспечил извлечение в 1,2–2 раза меньше векторных примитивов, чем преобразование Хафа, а для IV — в 2–3 раза меньше.

Пары групп испытаний I и II, III и IV различаются радиусом круга покрытия для пикселей линий покрывающего изображения. В последних парах групп этот радиус больше, чем в первых, что приводит к увеличению качества векторизации для обоих методов для III групп (где не применяется отсечение коротких примитивов). Однако для IV групп отсечение коротких векторных примитивов оказывает обратный эффект и приводит к снижению качества, особенно для предлагаемого метода.

Применение в тестах варьируемого радиуса покрытия пикселей показывает характер погрешностей векторизации. Для преобразования Хафа они обусловлены отсутствием векторных примитивов для мелких линий на растровом изображении, а для предлагаемого метода заключаются в неточном соответствии векторных примитивов линиям на растровом изображении.

Для I и III групп испытаний, где не используется отсечение коротких векторных примитивов, предлагаемый метод обеспечивает приблизительно одинаковое качество для обоих тестовых изображений для всех их соответствующих размеров. При этом параметры метода не менялись, тогда как преобразование Хафа требует для каждого растрового изображения и его размера подбора параметров для достижения сравнимого качества векторизации.

Предлагаемый метод во всех группах испытаний для всех размеров всех исходных растровых изображений позволяет получить существенно меньшие коэффициенты половинной длины, что свидетельствует о большем участии крупных векторных примитивов в формировании векторного изображения по сравнению преобразованием Хафа. Также предлагаемый метод показал в 2–3 раза более высокую производительность, чем преобразование Хафа в OpenCV.

Экспериментально полученные оценки вычислительной сложности обоих методов приблизительно равны и являются менее чем линейными относительно числа пикселей в растровых изображениях. Исходя из этих оценок, практически определяемая вычислительная сложность предлагаемого метода больше зависит от изображения, но в общем является меньшей, чем у преобразования Хафа.

## ВЫВОДЫ

Описан разреженно-пиксельный метод векторизации растровых линейчатых изображений. Проведены сравнительные исследования производительности и вычислительной сложности метода, а также качества векторизации по описанной методике. Метод обладает такими преимуществами, как высокая производительность, аппроксимация растрового изображения отрезками и дугами, значительная доля крупных отрезков и дуг в получаемом векторном изображении и автоматическая работа с полноцветными растровыми изображениями.

## ЛИТЕРАТУРА

1. Janssen R.D.T., Vossepoel A.M. Adaptive vectorization of line drawing images // Computer vision and image understanding. — 1997. — 65. — P. 38–56.
2. Wenyin L., Dori D. From Raster to Vectors: Extracting Visual Information from Line Drawings // Pattern Analysis and Applications. — 1999. — 2, N 1. — P.10–21.
3. Помехоустойчивый волновой алгоритм векторизации линейных растровых объектов. — [http://stanislavmoskalenko.narod.ru/articles/article\\_mashin3.htm](http://stanislavmoskalenko.narod.ru/articles/article_mashin3.htm).
4. Dori D., Liu W. Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation // IEEE Pattern Analysis and Machine Intelligence. — 1999. — 21, N 3. — P. 202–215.
5. Song J., Su F., Chen J., Tai C., Cai S. Line Net Global Vectorization: an Algorithm and Its Performance Evaluation // Computer Vision and Pattern Recognition 2000. — 2000. — P. 1383–1388.
6. OpenCV. — <http://opencv.org>.
7. OpenCV Hough Line Transform. — [http://docs.opencv.org/doc/tutorials/imgproc/imgtrans/hough\\_lines/hough\\_lines.html](http://docs.opencv.org/doc/tutorials/imgproc/imgtrans/hough_lines/hough_lines.html).
8. Stephens R.S. A Probabilistic Approach to the Hough Transform // Image and Vision Computing. — 1991. — 9, N 1. — P. 66–71.
9. Bradley D., Roth G. Adaptive Thresholding Using the Integral Image // ACM Journal of Graphics Tools. — 2007. — 12, N 2. — P. 13–21.

Поступила 20.12.2012